Imperial College London

Department of Computing

Hybrid Logic with the Sigma Binder

by

Sebastian Dörner

Submitted in partial fulfilment of the requirements for the MSc Degree in Advanced Computing of Imperial College London

September 2012

Abstract

This thesis investigates the hybrid language with the Σ binder ($\mathrm{ML}+\Sigma$, for "Modal Logic and Σ"). We show that all sentences of the language are essentially equivalent to a boolean combination of modal formulas and first-order sentences. We define a new type of bisimulation ("Σ-bisimulation") to mean bisimilar and elementarily equivalent. We show that any first-order formula $\alpha(x)$ is invariant under Σ-bisimulation if and only if it is equivalent to the standard translation of a formula in $\mathrm{ML}+\Sigma$. We prove a second bisimulation characterisation for $\mathrm{ML}+\Sigma$ using "strong Σ-bisimulation", which means bisimilar and isomorphic. A third characterisation is given using a game. We show that it is undecidable whether a first-order formula is invariant under Σ-bisimulation. Finally, we prove that any sentence in $\mathrm{ML}+\Sigma$ without nominals that is invariant under forward-generated submodels, disjoint unions or p-morphisms is equivalent to a modal formula.

Acknowledgements

I would like to thank Ian Hodkinson for supervising this project with a personal investment of time that is much higher than I could expect. He was always motivated to comment on my results and suggest ways to improve upon them. It is hard to imagine doing this project without him guiding me through the huge body of available knowledge in the area, providing valuable ideas and sharpening my thinking.

Thanks also to Anja Bachmann and Christian Braune for reading drafts of this thesis and for giving some helpful comments and suggestions.

I thank the German National Academic Foundation and my parents for sponsoring my studies at Imperial College. Finally, I would also like to thank my parents and friends for supporting me in the step to go abroad and making this year an unforgettable time.

Contents

Chapter 1

Introduction

1.1 Motivation

With a background in computer science, it is easy to see *relational structures* as ubiquitous. Even the restricted form of graphs is applicable to a variety of real-world problems, from the layout of public infrastructure (water, electricity, public transport) to linguistics, the study of molecules in chemistry or the modelling of dependencies in bug-tracking software.

One of the most widely taught logics reasoning about relational structures is *first-order predicate logic*. It is very expressive, but with atoms, terms and formulas, binding operators, distinctions between free and bound variables and the need for an assignment to evaluate first-order formulas, it is not the most accessible logic. If we restrict to sentences, which can be evaluated independently of assignments, all variables have to be bound. As the binding operators $\exists$ and $\forall$ have global semantics (a formula has to hold for one or all objects of the whole domain), first-order sentences only provide a global perspective on the relational structure. Furthermore, first-order logic becomes undecidable very quickly (depending on what relational and functional symbols we admit, see [BGG97]).

Modal logic is also used to reason about relational structures. But in all the above mentioned aspects, it is quite the opposite from first-order logic: It uses a simple syntax without variables, and so its special operator $\diamond$ is non-binding. As modal formulas are evaluated at a specific point (usually called "world") of the relational structure, modal logic provides an internal view on the structure. The modal operator only allows to get information about worlds that are accessible from the current world of evaluation (via the accessibility relation R). The term "world" comes from the philosophical background of regarding the points of the relational structure as "possible states of affairs". This serves distinguishing absolute truths ("The planet Earth orbits the Sun.") holding at every world from more contingent truths ("You are wearing blue jeans.") that only hold at some of the worlds. In contrast, the global perspective of first-order logic does not provide much support for expressing modality. As a final difference we mention the fact that the basic modal logic is decidable (see [vB10, Chapter 4]).

Hybrid logics combine aspects of both of these approaches. They are mostly regarded as an extension of modal logic adding in first-order concepts. The most central of these is certainly the addition of so-called *world-variables*. They are comparable to first-order variables and name specific worlds. As in first-order logic, they can be bound using operators. However, the binding might be more or less restricted, e. g. by only allowing to bind the current world or requiring to also change the world of evaluation before the binding happens. But hybrid formulas are still evaluated at a world, so the local view of modal logic is retained. This combination is very interesting: For instance, some hybrid logics are able to express that the current world is reflexive, which is not possible in modal logic. Some operators also provide a way to break out of this local perspective and to look at worlds that might not be accessible from the current one. Depending on which hybrid operators are admitted, we obtain hybrid languages with different degrees of expressiveness, some of them as expressive as first-order logic.

Despite their more theoretical and philosophical origins, hybrid logics have already found their way into applications, in particular in the areas of model checking and query and constraint evaluation. In [FdR06], Franceschet and de Rijke look at Lorel [AQM$^+$97], a query language for semi-structured data, e. g. in XML format. They embed different fragments of Lorel into different hybrid logics and show how to use hybrid model checkers to evaluate Lorel queries. Moreover, they do the same for a constraint language, which can be used for "imposing conditions on nodes at arbitrary depths in the data graph" [FdR06, p. 23]. They basically interpret an XML tree as a Kripke model. Similarly, in [HS08], Hoareau and Satoh use a hybrid logic to process location queries. Their location model represents locations as a tree hierarchy where each node represents a location. The more specific the location is, the deeper is the node; for instance the different departments of a hospital would all appear below a common node for the hospital. They then interpret this tree as a Kripke model and use a hybrid language to formulate queries for it. Most recently, Søgaard and Kristiansen presented a very fast querying tool for dependency treebanks, based on hybrid logic [SK12]. "Dependency treebanks are collections of natural language sentences annotated with dependency trees." [SK12] The authors then apply a hybrid query language to these dependency trees.

1.2 Project Objectives

There is a huge difference between formally specifying the semantics of a logic and actually knowing "how it ticks". One way to help us understand the expressive strengths and weaknesses of a language are characterisations specifying how two models must be related to each other in order to be equivalent in terms of the logic. Bisimulations characterise modal logic in such a sense: If two models are bisimilar, modal formulas holding in the one model also hold in the other (i. e. modal formulas are *bisimulation-invariant*). In fact the first-order formulas that are bisimulation-invariant are exactly those that are equivalent to a modal formula. Multiple hybrid languages have also been characterised using variants of bisimulation, but the hybrid logic with the Σ binder has so far not been studied in this regard.

The overall goal of this project was to explore the expressive power of the hybrid logic with the Σ binder. The main way to achieve this is to find a bisimulation-like characterisation (we call it Σ-bisimulation) for the logic. Other ways to investigate the logic have deliberately been left open so as to not restrict the paths on which our initial results may lead us.

1.3 Related Work

The foundations of hybrid logics have been laid by Arthur Prior in the 1950s and 1960s, and his work has been continued by his student Robert Bull. "About fifteen years later in Sofia, Bulgaria, nominals [(one aspect of hybrid logics)] were re-discovered by Gargov, Passy and Tinchev in their investigations on Boolean modal logic and propositional dynamic logic." [BvBW07, p. 828] Finally, the most recent developments were started by Blackburn and Seligman, investigating very expressive hybrid languages towards the end of the 1990s. [BvBW07, Chapter 14].

This work is mostly based on the findings by Blackburn and Seligman [BS95, BS98]. Particularly relevant is [BS95], which investigates the Σ binder as part of a hierarchy of hybrid operators. The authors show that it is strictly less expressive than first-order logic. As a by-product of proving the hierarchy, they also show that Σ-sentences are preserved by a so-called "full internal bisimulation". However, this is not a characterisation, as the full internal bisimulation requires more than what is necessary for Σ-sentences to be preserved. Finally, they give an intuition for the local and global strength of the Σ-language and show that it is undecidable.

Tzakova also explored some very expressive hybrid languages. However, she did not investigate bisimulation characterisations, but focussed on axiomatisation and tableau systems [Tza99a, Tza99b].

The first characterisation of the kind we will present here was the characterisation of modal logic by invariance under bisimulation and is due to van Benthem [vB76, vB85]. Turning to hybrid logics, similar bisimulation-like characterisations (using different notions of bisimulation) have been established for hybrid logics with both the @ and $\downarrow$ operators by Areces, Blackburn and

Marx [ABM01], with only the @ operator by ten Cate [tC05], with only the $\downarrow$ operator by Hodkinson and Tahiri [HT10], and finally for the hybrid logic with the $\forall$ binder by Kouvaros [Kou11].

1.4 A Brief Introduction to Hybrid Logics

This section will give a brief introduction to hybrid logics to facilitate the overview of our results in the next section. A more formal version will be given in Chapter 2. Just as modal logic, hybrid logics operate on so-called Kripke models, which are directed graphs. The nodes are called worlds and are connected by an accessibility relation R. A hybrid formula is always evaluated at a world. It may contain variables naming some of these worlds, and so – like in first-order logic – we need an assignment of these variables to worlds in the Kripke model. We have atomic formulas $p, q, \ldots$, for which an interpretation (part of the model) gives the worlds at which they evaluate to true. These can then be combined with boolean operators. For instance, the following says that the formula $p \wedge x$ is true at a world t of model M under assignment g: $M, g, t \models p \wedge x$. This means that the atom p is true at t and that the assignment g assigns just the world t to the variable x.

To move between worlds, modal logic adds the operator $\Diamond$ with the following semantics:

$$M, g, t \models \Diamond\phi \quad \text{iff} \quad \exists u \in |M| : R(t, u) \wedge M, g, u \models \phi$$

So $\Diamond$ moves along the R-edges. The hybrid logics we are concerned with also add the following operators:

$$\begin{aligned} M, g, t \models E\phi &\quad \text{iff} \quad \exists u \in |M| : M, g, u \models \phi \\ M, g, t \models {\downarrow}x\phi &\quad \text{iff} \quad M, g', t \models \phi, \text{ where } g' \text{ is } g \text{ with the assignment for } x \text{ changed to } t \\ M, g, t \models \Sigma x\phi &\quad \text{iff} \quad M, g, t \models E{\downarrow}x\phi \end{aligned}$$

The operator E moves the evaluation to *any* world in the model (we refer to this as a "jump"), $\downarrow$ binds a variable to the current world and Σ first moves the evaluation to any world in the model and then binds this world to a variable. We name hybrid languages according to which of these operators they are allowed to use, e. g. as $\mathrm{ML} + \Sigma$ (the full modal logic plus the Σ operator).

1.5 Achievements

Here we give an informal overview of our results and the research contributions of this work.

In [BS95], Blackburn and Seligman have shown that $\mathrm{ML} + E + \downarrow$ is equally expressive as first-order logic, but that $\mathrm{ML} + \Sigma$ is strictly less expressive. This is the case, although Σ is just $E{\downarrow}$. So the order of *first* moving to another world and only then saving it is crucial. It is easy to see that Σ can emulate E: Just use a variable y not occurring in the current formula ϕ, then $\Sigma y\phi$ is equivalent to $E\phi$. But this means that in $\mathrm{ML} + \Sigma$ we cannot bind the current world, because otherwise we could also emulate $\downarrow$ and would be as strong as first-order logic. This reveals itself to be the major restriction of the language.

However, once we use the Σ-operator, the new world of evaluation *is* bound to a variable. We will show that once we reach that state, we can always give a name to (or bind) any new worlds we reach through jumps or by travelling along the R-edges. If x is bound to the current world, we can even move backwards along R using formulas like $\Sigma y\Diamond x$. This binds y to an R-predecessor of the world named by x. As we always bind these worlds and can go back to them later on (using Σzx), this gives us the power to analyse the whole graph structure of a finite model. So essentially, we only need an initial jump to become as strong as first-order logic. But this initial jump means we lose track of the world we started with. Therefore the first-order properties we can check are global properties, i. e. first-order sentences. More specifically and in terms of the hybrid logics, any formula of the form $E\phi$ with $\phi \in \mathrm{ML} + E + \downarrow$ is equivalent to a Σ-formula.

Looking at it the other way around, what else can $\mathrm{ML} + \Sigma$ express? If we do not use the Σ-operator, we obviously still have the full power of modal logic. Once we use the Σ-operator, we might jump to *any* world in the model, so we lose all local knowledge. So then we really cannot express any more than first-order sentences. Overall, any sentence in $\mathrm{ML}+\Sigma$ is equivalent to a boolean combination of modal formulas and first-order sentences.

First-order sentences apply globally to the model, i. e. they are not tied to any world. The same applies to formulas of the form $E\phi$ with $\phi \in \mathrm{ML}+E+\downarrow$. We will show that indeed for any first-order sentence there is an equivalent sentence $E\phi \in \mathrm{ML}+E+\downarrow$. As mentioned above, these formulas are also equivalent to Σ-formulas. Hence we know that for every first-order sentence there is an equivalent sentence in $\mathrm{ML}+\Sigma$. This implies undecidability of the Σ-language.

All these preliminary investigations help us to find a bisimulation-like characterisation. If we do not use the Σ operator, we are still as strong as modal logic. So the characterisation will require that the original worlds are indeed bisimilar. Once we use Σ, we lose the specific world we started with but get the strength of first-order sentences. So the models have to be elementarily equivalent, regardless of the initial point of evaluation. So the first bisimulation-like characterisation we prove for $\mathrm{ML}+\Sigma$, the "Σ-bisimulation", means "bisimilar and elementarily equivalent" .

Isomorphisms are in some sense the limit of elementary equivalence. Two isomorphic models are always elementarily equivalent and the converse holds for saturated models of the same cardinality [CK90, Theorem 5.1.13]. So we define "strongly Σ-bisimilar" as "bisimilar and isomorphic". We improve our characterisation by only requiring the weaker assumption of invariance under *strong* Σ-bisimulation, providing the second bisimulation-like characterisation shown in this thesis.

Another type of characterisation is often given through games played on two models. Our game for $\mathrm{ML}+\Sigma$ is a combination of the modal bisimulation game [vB10, Def. 3.5.1] and the Ehrenfeucht-Fraïssé game for first-order logic [Doe96, Def. 3.45]. Every move in the game corresponds to a non-boolean operator of the language. So we have two types of moves, one for $\Diamond$ and one for Σ. The $\Diamond$-type move comes directly from the modal bisimulation game and moves along R-edges. The Σ-type move steers the player to any world in the model. The saving of this world in a variable is simulated using *coloured pebbles*, just as in some definitions of the first-order game. Ultimately, a winning strategy for one of the players implies that the models are Σ-bisimilar.

As $\mathrm{ML}+\Sigma$ is basically a subset of first-order logic, we would like to know for a first-order formula, whether it is equivalent to a sentence in $\mathrm{ML}+\Sigma$. Using a technique called relativisation, which serves to restrict a part of a formula to only apply to a part of a model, we show that a first-order sentence α is valid iff a derived sentence $f(\alpha)$ is invariant under Σ-bisimulation. Therefore, as the validity of first-order sentences is undecidable, so must be invariance under Σ-bisimulation. Given our characterisation this implies that it is undecidable whether a first-order formula is equivalent to a sentence in $\mathrm{ML}+\Sigma$.

Finally, we were quite surprised by finding a modal core of $\mathrm{ML}+\Sigma$: We prove that when a Σ-formula, which is always invariant under Σ-bisimulation, is also invariant under forward-generated submodels, disjoint unions or p-morphisms, then it is equivalent to a modal formula.

We summarise our main results more concisely:

- We show that all sentences of $\mathrm{ML}+\Sigma$ are essentially equivalent to a boolean combination of modal formulas and first-order sentences.

- We define a new type of bisimulation ("Σ-bisimulation") to mean bisimilar and elementarily equivalent. We then show that any first-order formula $\alpha(x)$ is invariant under Σ-bisimulation if and only if it is equivalent to the standard translation of a formula in $\mathrm{ML}+\Sigma$.

- We prove a second bisimulation characterisation for $\mathrm{ML}+\Sigma$ using "strong Σ-bisimulation", which means bisimilar and isomorphic.

- A third characterisation is given using a game combining aspects of the modal and first-order games.

- We show that it is undecidable whether a first-order formula is invariant under Σ-bisimulation.

- We prove that any sentence in $\mathrm{ML}+\Sigma$ without nominals that is invariant under forward-generated submodels, disjoint unions or p-morphisms is equivalent to a modal formula.

1.6 Outline

The rest of this thesis is structured as follows. Chapter 2 formally introduces the modal language and, building up on that, a general notation for hybrid languages. We also present the modal bisimulation and formally state the above mentioned characterisation by van Benthem. Thereby, Chapter 2 consists of previous work we built our results on.

Chapter 3 investigates the expressiveness of the hybrid language with the Σ binder. We first give an intuition for what kind of model properties can be expressed. We then investigate the differences between the two hybrid languages $\mathrm{ML} + \Sigma$ and $\mathrm{ML} + E + \downarrow$. Finally, we relate the hybrid languages to first-order logic.

Building upon these results, Chapter 4 establishes the major contributions of this work. We prove the three characterisations of $\mathrm{ML} + \Sigma$ mentioned above. Chapter 5 establishes that invariance by Σ-bisimulation is undecidable. Chapter 6 shows that those Σ-sentences that are invariant under forward-generated submodels, disjoint unions or p-morphisms are equivalent to modal formulas. In Chapter 7 we conclude this work by summarising our results and giving an outlook to future research opportunities.

Chapter 2

Hybrid Languages

We will present hybrid languages as an extension of the modal language. Hence, before showing the more advanced hybrid concepts, we will lay a foundation by formally introducing modal logic. A lot of the concepts, like Kripke models and the Standard Translation apply to both modal and hybrid logics.

2.1 Modal Language

2.1.1 Syntax of the Modal Language

Definition 1 (Modal signature)**.** Let PROP and NOM be pairwise disjoint, countably infinite sets, where PROP is a set of so-called *propositional symbols* (or *(propositional) atoms*) and NOM a set of *nominals*. Then the pair $(\mathrm{PROP}, \mathrm{NOM})$ is called a *modal signature*.

We define modal languages over these modal signatures and where not specified differently we implicitly use the signature $(\mathrm{PROP}, \mathrm{NOM})$.

Definition 2 (Basic Modal Language ML)**.** Let $(\mathrm{PROP}, \mathrm{NOM})$ be a modal signature. We inductively define the Basic Modal Language ML (the set of modal formulas):

1. $\top \in \mathrm{ML}$
2. if $p \in \mathrm{PROP}$, then $p \in \mathrm{ML}$
3. if $i \in \mathrm{NOM}$, then $i \in \mathrm{ML}$
4. if $\phi \in \mathrm{ML}$, then $\neg\phi \in \mathrm{ML}$ and $\Diamond\phi \in \mathrm{ML}$
5. if $\phi \in \mathrm{ML}$ and $\psi \in \mathrm{ML}$, then $\phi \wedge \psi \in \mathrm{ML}$
6. ML only contains the formulas generated by applying rules 1-5 finitely many times.

All unary operators bind tighter than the binary $\wedge$ operator. As usual, we use parentheses to modify or highlight the operator precedence. Let ϕ and ψ be modal formulas. For convenience, we extend the minimal set of operators above with the following abbreviations:

$$
\begin{array}{rcl}
\bot & := & \neg\top \\
\Box\phi & := & \neg\Diamond\neg\phi \\
\phi \vee \psi & := & \neg(\neg\phi \wedge \neg\psi) \\
\phi \rightarrow \psi & := & \neg(\phi \wedge \neg\psi) \\
\phi \leftrightarrow \psi & := & \neg(\phi \wedge \neg\psi) \wedge \neg(\psi \wedge \neg\phi)
\end{array}
$$

Most of this is known from first-order logic (see [CK90]). The new operators are read as Diamond ($\Diamond$) and Box ($\Box$).

Remark. The inclusion of nominals in the definition of modal formulas is non-standard. Most literature considers nominals to be a hybrid feature already, whereas our definition of hybrid logics only starts when we add extra operators. We do this for convenience, as this means that we can use several definitions for both modal and hybrid logic.

2.1.2 Semantics of the Modal Language

There exist several different semantics of the Modal Language. Throughout this thesis, we will use the most prominent one, which is *Kripke semantics*. Here, modal formulas are evaluated at different points (so-called worlds) of a Kripke frame.

Definition 3 (Kripke frame)**.** Let W be a non-empty set of worlds and $R \subseteq W \times W$ a binary relation on W. Then the pair (W, R) is called a *Kripke frame*.

Definition 4 (Power set)**.** If W is a set, then $\mathcal{P}(W) := \{V \mid V \subseteq W\}$ is called the *power set* of W.

Definition 5 (Kripke model)**.** Let $F = (W, R)$ be a Kripke frame and $h : \mathrm{PROP} \cup \mathrm{NOM} \to \mathcal{P}(W) \cup W$ a map specifying all worlds at which a given atom or nominal is true such that

- for all $p \in \mathrm{PROP}$, $h(p) \in \mathcal{P}(W)$ and
- for all $i \in \mathrm{NOM}$, $h(i) \in W$.

Then $M = (W, R, h)$ is called a *Kripke model*. For convenience, we sometimes simply write $M = (F, h)$ and $|M|$ for W.

Definition 6 (Pointed model)**.** Let $M = (W, R, h)$ be a Kripke model and $t \in W$. Then the pair (M, t) is called a *pointed (Kripke) model*.

Now we can inductively define how to evaluate an arbitrary modal formula at a world t:

Definition 7 (Modal truth)**.** Let $M = (W, R, h)$ be a Kripke model, $t \in W$ a world and ϕ a modal formula. We write $M, t \models \phi$ (read ϕ is *true* at t in M) if and only if one of the following conditions holds:

- $\phi = \top$
- $\phi = p$ for a propositional symbol $p \in \mathrm{PROP}$ and $t \in h(p)$
- $\phi = i$ for a nominal $i \in \mathrm{NOM}$ and $t = h(i)$
- $\phi = \neg\psi$ and $M, t \not\models \psi$
- $\phi = \Diamond\psi$ and $\exists u \in W : R(t, u) \wedge M, u \models \psi$
- $\phi = \psi_1 \wedge \psi_2$ and $M, t \models \psi_1$ and $M, t \models \psi_2$

Definition 8 (Modal equivalence)**.** Let (M, m) and (N, n) be two pointed Kripke models. We say (M, m) and (N, n) are *modally equivalent* (written $(M, m) \equiv_{\mathrm{mod}} (N, n)$) iff they fulfil exactly the same modal formulas, i. e. whenever ϕ is a formula in ML, it holds $M, m \models \phi$ iff $N, n \models \phi$.

We defined truth of modal formulas essentially using statements about the relation R and the truth of atoms and nominals. These definitions are expressible in first-order logic and thus we can define a so-called *Standard Translation* of modal formulas into first-order logic. The signature for this logic is derived from the modal signature:

Definition 9 ($\mathcal{L}$-formula)**.** An *$\mathcal{L}$-formula* is a formula of first-order predicate logic with equality defined over the signature $\mathcal{L} = \{i \mid i \in \mathrm{NOM}\} \cup \{P \mid p \in \mathrm{PROP}\} \cup \{R\}$, where all i's are constant symbols, all P's are unary relation symbols and R is a binary relation symbol (the P's are just the capitalised versions of the propositional symbols). This first-order language is usually called the correspondence language (of modal or hybrid logic).

We use the symbols i and R in both modal (later hybrid) and first-order formulas (or models). It will be clear from the context which one we mean. This is sensible because we also encode the semantics of modal formulas into first-order logic, using the Standard Translation. Since the evaluation of modal formulas depends on a world at which we evaluate, the Standard Translation and thus the corresponding first-order formula has one more free variable to represent that world.

Definition 10 (Modal Standard Translation)**.** The *Standard Translation* $\mathrm{ST}_x(\phi)$ of a modal formula ϕ w. r. t. a variable x into a first-order $\mathcal{L}$-formula is inductively defined as follows:

- $\mathrm{ST}_x(\top) := \top$
- $\mathrm{ST}_x(p) := P(x)$ for all propositional symbols p (associate p with P, q with Q, etc.)
- $\mathrm{ST}_x(i) := x = i$ for all nominals i
- $\mathrm{ST}_x(\neg\phi) := \neg\mathrm{ST}_x(\phi)$
- $\mathrm{ST}_x(\phi \wedge \psi) := \mathrm{ST}_x(\phi) \wedge \mathrm{ST}_x(\psi)$
- $\mathrm{ST}_x(\Diamond\phi) := \exists y(R(x, y) \wedge \mathrm{ST}_y(\phi))$, where y is a variable not occurring in ϕ.

It is easy to see that the Standard Translation is equivalent to the definition of truth in Definition 7, i. e. it reflects the Kripke semantics. So truth of a modal formula in a modal model corresponds to truth of the Standard Translation of that formula in the corresponding first-order structure. More formally, we define:

Definition 11 (Induced first-order structure)**.** Let $M = (W, R, h)$ be a Kripke model. The *first-order structure induced by M* is the first-order structure with domain W over the signature $\mathcal{L}$, where we interpret i (in $\mathcal{L}$) as $h(i)$ for all $i \in \mathrm{NOM}$, R (in $\mathcal{L}$) as R (in the Kripke model) and P as $h(p)$ for all $p \in \mathrm{PROP}$.

When it is clear from the context, we may simply treat a Kripke model as its induced first-order structure without introducing a new name for that. We also denote both modal and first-order truth using "$\models$". As modal truth always has a world on the left of $\models$, these are easily distinguishable. For first-order truth, we either give an assignment on the left, as in $M, g \models \alpha$, or, if α has only one free variable, we write the domain element assigned to this variable in brackets after the formula: $M \models \alpha[t]$. For first-order sentences α, we simply write $M \models \alpha$. The following theorem is a good example, as it relates modal truth to first-order truth.

Theorem 1 (Adequacy of Standard Translation)**.** Let ϕ be a modal formula, (M, t) a pointed Kripke model and x a first-order variable. Then it holds:

$$M, t \models \phi \quad \text{iff} \quad M \models \mathrm{ST}_x(\phi)[t]$$

Proof. Via induction on the structure of ϕ. □

Remember that if M is a Kripke model, $|M|$ denotes its set of worlds. To make this consistent for the induced first-order structures, if N is a first-order structure, we let $|N|$ denote its domain. This means that whether we look at the Kripke model M as an actual Kripke model or as its induced first-order structure, $|M|$ refers to the same set.

Definition 12 (Elementarily equivalent)**.** Let A and B be two first-order structures over $\mathcal{L}$. We say that A and B are elementarily equivalent (written $A \equiv B$) if and only if for all sentences ϕ over $\mathcal{L}$ it holds $A \models \phi$ iff $B \models \phi$.

Let M and N be two Kripke models. We say that M and N are elementarily equivalent iff their induced first-order structures are elementarily equivalent.

2.1.3 Bisimulation

One way to characterise modal logic is by bisimulation. It relates Kripke models to one another, given they fulfil certain conditions. If this is the case for two models, then these two models cannot be distinguished by modal formulas. Formally, we define:

Definition 13 (Bisimulation)**.** A *bisimulation* between two pointed models (M_1, t_1) and (M_2, t_2) with $M_1 = (W_1, R_1, h_1)$ and $M_2 = (W_2, R_2, h_2)$ is a relation $B \subseteq W_1 \times W_2$ such that the following conditions hold:

- $(t_1, t_2) \in B$
- If $(x, y) \in B$ then for all $p \in \mathrm{PROP} : (x \in h_1(p) \leftrightarrow y \in h_2(p))$.
- If $(x, y) \in B$ then for all $i \in \mathrm{NOM} : (x = h_1(i) \leftrightarrow y = h_2(i))$.
- *forth*: If $(x_1, y_1) \in R_1$ and $(x_1, x_2) \in B$, then there is a $y_2 \in W_2$ such that $(x_2, y_2) \in R_2$ and $(y_1, y_2) \in B$.
- *back*: If $(x_2, y_2) \in R_2$ and $(x_1, x_2) \in B$, then there is a $y_1 \in W_1$ such that $(x_1, y_1) \in R_1$ and $(y_1, y_2) \in B$.

If there is a bisimulation between two pointed models (M_1, t_1) and (M_2, t_2), we say that (M_1, t_1) and (M_2, t_2) are *bisimilar* (written $(M_1, t_2) \underline{\leftrightarrow} (M_2, t_2)$).

The following Theorems 2 and 3 constitute seminal work by van Benthem, characterising the relationship between bisimulations and modal logic. For full proofs of see [BdRV02, Theorems 2.20 and 2.68][1].

Theorem 2. Let ϕ be a modal formula and (M_1, t_1) and (M_2, t_2) two bisimilar Kripke models. Then $M_1, t_1 \models \phi$ if and only if $M_2, t_2 \models \phi$.

Proof. Via induction on the structure of ϕ. □

Definition 14 (Bisimulation-invariance)**.** Let $\alpha(x)$ be an $\mathcal{L}$-formula with at most one free variable x. The formula α is said to be *bisimulation-invariant* (or *invariant under bisimulation*) if whenever (M, t) and (M', t') are bisimilar, then $M \models \alpha[t]$ iff $M' \models \alpha[t']$.

Theorem 3 (after [Hod11b], originally van Benthem, 1976)**.** An $\mathcal{L}$-formula $\alpha(x)$ with at most one free variable x is logically equivalent to the Standard Translation of a modal formula if and only if α is bisimulation-invariant.

2.2 Extension to Hybrid Languages

Hybrid languages are based on the basic modal language and usually add some more operators. Some examples of such operators are $\downarrow$, E and Σ. Depending on which and how many of these additional operators are used, we obtain hybrid languages with different properties. We name these languages extending the basic modal language ML in a natural way, e. g. as $\mathrm{ML} + E + \downarrow$ for the hybrid language using the basic modal language extended by the additional operators E and $\downarrow$ (cf. [BS98]).

[1] These theorems are proved for modal logic without nominals, but the proofs work in the same way for modal logic with nominals.

2.2.1 Syntax of Hybrid Languages

We already mentioned that hybrid languages introduce concepts of first-order logic. One such concept are variables, which are used across all hybrid languages we are concerned with in this work. In the context of hybrid languages, they serve to name a specific world in the Kripke frame. Some of the additional hybrid operators bind these world variables ("binding operators") while others do not. In the context of this work, we will use

- the binding operators $\downarrow$ and Σ
- the non-binding operator E.

The non-binding operator is in a sense modal, because it does not use world variables. We add variables to our original modal signature:

Definition 15 (Hybrid signature)**.** Let PROP, NOM and VAR be pairwise disjoint, countably infinite sets, where PROP is a set of propositional symbols (or (propositional) atoms), NOM a set of nominals and VAR a set of *(world) variables.* Then the triple $(\mathrm{PROP}, \mathrm{NOM}, \mathrm{VAR})$ is called a *hybrid signature.* Furthermore, we denote the set of *world symbols* as $\mathrm{WSYM} := \mathrm{NOM} \cup \mathrm{VAR}$.

We define hybrid languages over these hybrid signatures and where not specified differently we implicitly use the signature $(\mathrm{PROP}, \mathrm{NOM}, \mathrm{VAR})$.

Definition 16 (Hybrid Language)**.** We inductively define the Hybrid Language $\mathrm{HL} = \mathrm{ML} + B_1 + B_2 + \ldots + B_n + N_1 + N_2 + \ldots + N_k$ over the signature $(\mathrm{PROP}, \mathrm{NOM}, \mathrm{VAR})$, where $n, k \geq 0$, B_i are binding operators in $\{\downarrow, \Sigma\}$ and N_i are non-binding operators in $\{E\}$ as follows:

1. $\top \in \mathrm{HL}$
2. if $p \in \mathrm{PROP}$, then $p \in \mathrm{HL}$
3. if $i \in \mathrm{NOM}$, then $i \in \mathrm{HL}$
4. if $x \in \mathrm{VAR}$, then $x \in \mathrm{HL}$
5. if $\phi \in \mathrm{HL}$, then $\neg\phi \in \mathrm{HL}$ and $\Diamond\phi \in \mathrm{HL}$
6. if $\phi \in \mathrm{HL}$ and $\psi \in \mathrm{HL}$, then $\phi \wedge \psi \in \mathrm{HL}$
7. if $\phi \in \mathrm{HL}$, then $B_i x\phi \in \mathrm{HL}$, where $1 \leq i \leq n$ and $x \in \mathrm{VAR}$
8. if $\phi \in \mathrm{HL}$, then $N_i\phi \in \mathrm{HL}$, where $1 \leq i \leq k$
9. HL only contains the formulas generated by applying rules 1-8 finitely many times.

The elements of HL are called (hybrid) formulas over $(\mathrm{PROP}, \mathrm{NOM}, \mathrm{VAR})$.

The recursive definition of hybrid formulas gives rise to a formation tree for each formula, with the formula itself as the root of the tree.

Definition 17 (adapted from [Hod11a, Def. 8.3])**.** Let ϕ be a hybrid formula and $x \in \mathrm{VAR}$ a variable occurring in ϕ. We say that an occurrence of x is *bound* if its position in the formation tree of ϕ is below a binding operator $B_i x$. Otherwise x is said to be *free* in ϕ.

Definition 18. A hybrid formula is called a *sentence* if all occurrences of variables in it are bound.

2.2.2 Semantics of Hybrid Languages

As in modal logic, the truth of a hybrid formula depends on the world we evaluate it at. Additionally, it depends on the value of the world variables it contains, i. e. which worlds are named by them. This value is determined by an assignment function, similar to the one known from first-order logic.

Definition 19 (Assignment)**.** Let VAR be a set of world variables and $F = (W, R)$ a Kripke frame. A function $g : \mathrm{VAR} \to W$ is called an *assignment (function)*.

For the semantics of some of our operators, we will need to modify an existing assignment at a specific variable. For that purpose we define an x-variant of an assignment following [Kou11, Definition 1.2.6]:

Definition 20 (x-variant)**.** Let g and g' be assignments and $x \in \mathrm{VAR}$. We call g' an *x-variant* of g, denoted $g' \overset{x}{\sim} g$, if and only if $\forall y \in (\mathrm{VAR} \setminus \{x\}) : g'(y) = g(y)$.

Definition 21 (Value of a world symbol)**.** Let $x \in \mathrm{WSYM}$ be a world symbol, $M = (W, R, h)$ a Kripke model and g an assignment. Then we define the value $[h, g](x) \in W$ of x as

$$[h, g](x) := \begin{cases} g(x) & \text{if } x \in \mathrm{VAR} \\ h(x) & \text{if } x \in \mathrm{NOM} \end{cases}$$

Definition 22 (Hybrid truth)**.** Let $M = (W, R, h)$ be a Kripke model, $t \in W$ a world, g an assignment and ϕ a hybrid formula. We write $M, g, t \models \phi$ (read ϕ is true at t in M under g) if and only if one of the following conditions holds:

- $\phi = \top$
- $\phi = p$ for a propositional symbol $p \in \mathrm{PROP}$ and $t \in h(p)$
- $\phi = x$ for a world symbol $x \in \mathrm{WSYM}$ and $t = [h, g](x)$
- $\phi = \neg\psi$ and $M, g, t \not\models \psi$
- $\phi = \Diamond\psi$ and $\exists u \in W : R(t, u) \wedge M, g, u \models \psi$
- $\phi = \psi_1 \wedge \psi_2$ and $M, g, t \models \psi_1$ and $M, g, t \models \psi_2$
- $\phi = E\psi$ and there is a world $u \in W$ such that $M, g, u \models \psi$
- $\phi = {\downarrow}x\psi$ for some $x \in \mathrm{VAR}$ and there is an assignment g' such that $g' \overset{x}{\sim} g, g'(x) = t$ and $M, g', t \models \psi$
- $\phi = \Sigma x\psi$ for some $x \in \mathrm{VAR}$ and there are a world $u \in W$ and an assignment g' such that $g' \overset{x}{\sim} g, g'(x) = u$ and $M, g', u \models \psi$

Modal formulas always stay in the part of the model that is reachable via the accessibility relation. In contrast, the operators E and Σ are not restricted through R. They cross longer distances within the Kripke model, in particular they can cross between the connected components of the R-graph. Throughout this work, we will refer to this process of applying E or Σ and thus bridging arbitrarily distant parts of the model as *jumping*. As we are mostly concerned with $\mathrm{ML} + \Sigma$, it will usually refer to the Σ operator, but this will be clear from the context.

Binding operators modify the assignment of the variable they bind and the modification does not depend on the previous value of that variable. Therefore the truth of hybrid sentences does not depend on the assignment for which we evaluate it.

Definition 23 (Hybrid truth for sentences)**.** Let ϕ be a hybrid sentence, M a Kripke model and $t \in |M|$. We write $M, t \models \phi$ iff for all assignments g it holds $M, g, t \models \phi$.

Definition 24 (Valid)**.** Let ϕ be a modal formula or a hybrid sentence. We say that ϕ is *valid* iff for all Kripke models M and all worlds $t \in |M|$ it holds $M, t \models \phi$. If α is a first-order sentence, then we say that α is valid iff for all first-order structures M it holds $M \models \alpha$.

We are interested in what a logic – in particular $\mathrm{ML} + \Sigma$ – can express, i. e. which models it can distinguish. If it cannot distinguish two models, they are equivalent with respect to the logic.

Definition 25 (Σ-equivalence)**.** Let (M, m) and (N, n) be two pointed Kripke models. We say (M, m) and (N, n) are Σ-equivalent iff they fulfil exactly the same Σ-sentences, i. e. whenever ϕ is a sentence in $\mathrm{ML} + \Sigma$, it holds $M, m \models \phi$ iff $N, n \models \phi$.

The Standard Translation we gave in Definition 10 can easily be extended to hybrid logic using the definition of truth from above. So both modal and hybrid logics are essentially subsets of first-order logic.

Definition 26 (Hybrid Standard Translation)**.** The Standard Translation $\mathrm{ST}_x(\phi)$ of a hybrid formula ϕ w. r. t. a variable x not occurring in ϕ into a first-order $\mathcal{L}$-formula is inductively defined as follows:

- $\mathrm{ST}_x(\top) := \top$
- $\mathrm{ST}_x(p) := P(x)$ for all propositional symbols p (associate p with P, q with Q, etc.)
- $\mathrm{ST}_x(i) := x = i$ for all nominals i
- $\mathrm{ST}_x(y) := x = y$ for all variables y
- $\mathrm{ST}_x(\neg\phi) := \neg\mathrm{ST}_x(\phi)$
- $\mathrm{ST}_x(\phi \wedge \psi) := \mathrm{ST}_x(\phi) \wedge \mathrm{ST}_x(\psi)$
- $\mathrm{ST}_x(\Diamond\phi) := \exists y(R(x, y) \wedge \mathrm{ST}_y(\phi))$, where y is a variable not occurring in ϕ
- $\mathrm{ST}_x(E\phi) := \exists y\, \mathrm{ST}_y(\phi)$, where y is a variable not occurring in ϕ
- $\mathrm{ST}_x(\downarrow y\phi) := \exists y(x = y \wedge \mathrm{ST}_x(\phi))$
- $\mathrm{ST}_x(\Sigma y\phi) := \exists y\, \mathrm{ST}_y(\phi)$

As in the modal Standard Translation, we identify the symbols for nominals with the first-order constant symbols. Additionally, we identify each hybrid variable with an equally-named first-order variable as visible in the definitions above. It will be clear from the context which one we refer to. As one would expect, there is no difference semantically:

Theorem 4 (Adequacy of Standard Translation)**.** Let ϕ be a hybrid formula, x a variable not occurring in ϕ, (M, t) a pointed Kripke model and g an assignment. Then for $g' \overset{x}{\sim} g$ and $g'(x) = t$ it holds:

$$M, g, t \models \phi \quad \text{iff} \quad M, g' \models \mathrm{ST}_x(\phi)$$

Proof. Via induction on the structure of ϕ. □

Remark. If ϕ in Theorem 4 is a sentence, then its hybrid evaluation does not depend on g. The evaluation of $\mathrm{ST}_x(\phi)$ only depends on the assignment of x, which is the only free variable. So then it holds:

$$M, t \models \phi \quad \text{iff} \quad M \models \mathrm{ST}_x(\phi)[t]$$

Chapter 3

On the Expressiveness of $\mathrm{ML} + \Sigma$

3.1 Introduction

In this section, we will give an intuition for the expressiveness of $\mathrm{ML} + \Sigma$. We look at certain properties of the Kripke model that can be expressed with Σ-formulas. This will guide the reader to an intuition for the sensibility of the definitions occurring in subsequent sections.

From Definition 22, it is easy to see that $\Sigma x\phi$ is equivalent to $E{\downarrow}x\phi$. In [BS95], Blackburn and Seligman have shown that $\mathrm{ML} + E + \downarrow$ is equally expressive as the first-order correspondence language over $\mathcal{L}$, and furthermore that $\mathrm{ML} + \Sigma$ is strictly less expressive. So the restriction to the specific pattern of *first* E and *then (immediately)* ${\downarrow}x$ is crucial.

The fundamental restriction is that Σ-formulas cannot bind the world they are evaluated at (the "current" world), because whenever they bind a variable, they first change the point of evaluation to any world in the model. Circumventing that problem, suppose we already have a world variable x bound to the current world. To illustrate the capabilities of Σ-formulas in this case, suppose we have a model $M = (W, R, h)$, a world t of its frame and an assignment g such that $g(x) = t$. Then Σ-formulas can do the following.

They can force to evaluate a subformula ψ at any world named by any variable y (using a *new* variable z that does not occur in ψ):

$$\begin{array}{lll}
M, g, t \models \Sigma z(y \wedge \psi) & \text{iff} & \exists u \in |M|, g' \overset{z}{\sim} g, g'(z) = u\text{: } M, g', u \models y \text{ and } M, g', u \models \psi \\
& \text{iff} & \exists u \in |M|, g' \overset{z}{\sim} g, g'(z) = g'(y) = u\text{: } M, g', u \models \psi \\
& \text{iff} & \exists g' \overset{z}{\sim} g, g'(z) = g(y)\text{: } M, g', g(y) \models \psi \qquad \text{(as } g'(y) = g(y)\text{)} \\
& \text{iff} & M, g, g(y) \models \psi \qquad \text{(as } z \text{ does not occur in } \psi \text{ and } g' \overset{z}{\sim} g\text{)}
\end{array}$$

Hence for the following changes to the assignment, it is not important that we change the world of evaluation as a side-effect. We can always go back to the original world (named by x) using the method above. Formulas in $\mathrm{ML} + \Sigma$ can also bind a predecessor or a successor of the world named by x to a variable y. The predecessor binding:

$$\begin{array}{lll}
M, g, t \models \Sigma y(\Diamond x \wedge \psi) & \text{iff} & \exists u \in |M|, g' \overset{y}{\sim} g, g'(y) = u\text{: } M, g', u \models \Diamond x \text{ and } M, g', u \models \psi \\
& \text{iff} & \exists u, v \in |M|, g' \overset{y}{\sim} g, g'(y) = u, R(u, v)\text{: } M, g', v \models x \text{ and } M, g', u \models \psi \\
& \text{iff} & \exists u, v \in |M|, g' \overset{y}{\sim} g, g'(y) = u, R(u, v), g'(x) = v\text{: } M, g', u \models \psi \\
& \text{iff} & \exists u \in |M|, g' \overset{y}{\sim} g, g'(y) = u, R(u, g(x))\text{: } M, g', u \models \psi \quad \text{(as } g'(x) = g(x)\text{)}
\end{array}$$

For the successor binding, let z be a variable not occurring in ψ.

$$\begin{aligned}
& M, g, t \models \Sigma y \Sigma z (x \wedge \Diamond y \wedge \psi) \\
\text{iff} \quad & \exists u \in |M|, g' \overset{y}{\sim} g, g'(y) = u\text{: } M, g', u \models \Sigma z (x \wedge \Diamond y \wedge \psi) \\
\text{iff} \quad & \exists u, v \in |M|, g' \overset{y}{\sim} g, g'(y) = u, g'' \overset{z}{\sim} g', g''(z) = v\text{: } M, g'', v \models x \wedge \Diamond y \wedge \psi \\
\text{iff} \quad & \exists u, v \in |M|, g' \overset{y}{\sim} g, g'(y) = u, g'' \overset{z}{\sim} g', g''(z) = g''(x) = v\text{: } M, g'', v \models \Diamond y \wedge \psi \\
\text{iff} \quad & \exists u \in |M|, g' \overset{y}{\sim} g, g'(y) = u\text{: } M, g', g(x) \models \Diamond y \wedge \psi \\
& \qquad \text{(as } g(x) = g''(x) \text{ and } z \text{ does not occur in } \Diamond y \wedge \psi) \\
\text{iff} \quad & \exists u, v \in |M|, g' \overset{y}{\sim} g, g'(y) = u, R(g(x), v)\text{: } M, g', v \models y \text{ and } M, g', g(x) \models \psi \\
\text{iff} \quad & \exists u, v \in |M|, g' \overset{y}{\sim} g, g'(y) = u = v, R(g(x), v)\text{: } M, g', g(x) \models \psi \\
\text{iff} \quad & \exists u \in |M|, g' \overset{y}{\sim} g, g'(y) = u, R(g(x), u)\text{: } M, g', g(x) \models \psi
\end{aligned}$$

A slight variation of this formula is $\Sigma y \Sigma z (x \wedge \Diamond y \wedge \Box y \wedge \psi)$. It is easy to see that this additionally ensures that y is the only successor of x: After binding y, the part Σz jumps back to the world named by x (otherwise x would not hold afterwards). Since y only holds at exactly one world, $\Diamond y \wedge \Box y$ ensures that the world named by y is indeed a successor of the current world, and that there are no other successors.

We can also count the number of predecessors of the world named by x. The formula

$$\phi_n = \Sigma x_1 (\Diamond x \wedge \Sigma x_2 (\neg x_1 \wedge \Diamond x \wedge \ldots \wedge \Sigma x_i (\bigwedge_{j=1}^{i-1} \neg x_j \wedge \Diamond x \wedge \cdots \wedge \Sigma x_n (\bigwedge_{j=1}^{n-1} \neg x_j) \wedge \Diamond x) \ldots)$$

is true iff the world named by x has at least n distinct predecessors. Using $\phi_n \wedge \neg \phi_{n+1}$, we can ensure that the world has exactly n predecessors. The same works for successors, using the formulas

$$\phi_n = \Sigma x_1 \Sigma x_2 (\neg x_1 \wedge \cdots \wedge \Sigma x_i (\bigwedge_{j=1}^{i-1} \neg x_j \wedge \ldots \wedge \Sigma x_n (\bigwedge_{j=1}^{n-1} \neg x_j \wedge \Sigma z (x \wedge \bigwedge_{j=1}^{n} \Diamond x_j) \ldots)$$

instead. Finally, we can globally count the number of distinct worlds in a model using

$$\phi_n = \Sigma x_1 \Sigma x_2 (\neg x_1 \wedge \Sigma x_3 (\neg x_1 \wedge \neg x_2 \wedge \ldots \wedge \Sigma x_i (\bigwedge_{j=1}^{i-1} \neg x_j \wedge \ldots \wedge \Sigma x_n (\bigwedge_{j=1}^{n-1} \neg x_j) \ldots)$$

and the composition $\phi_n \wedge \neg \phi_{n+1}$ as before.

The preceding formula makes it possible to distinguish models with a different number of worlds, as long as one of them is finite. In this case, it even gives names to all of the worlds. Using these names and formulas like $\Sigma z (x_i \wedge \Diamond x_j)$ or $\Sigma z (x_i \wedge \neg \Diamond x_j)$, we can check for the existence of arbitrary edges between those worlds. Checking that an atom holds is also easily possible with $\Sigma z (x_i \wedge p)$.

So if two finite models have the same number of worlds, we can name all of them as explained above and also encode the existence or non-existence of the finitely many possible R-edges between them. If we also encode whether p holds at world w for all worlds w and atoms p (finitely many), we can distinguish finite models up to isomorphism (see Definition 30 defining isomorphisms). So any two finite models fulfilling the same Σ-sentences need to be isomorphic.

Note that all of this relies on the binding of variables. However, once we use the first binding operator Σ, we implicitly have to jump to an arbitrary world in the model. Hence we loose the information of the distinguished point of the pointed model, and so the isomorphism is also independent of the distinguished point.

We know that $\{\mathrm{ST}(\phi) \mid \phi \in \mathrm{ML} + \Sigma\ \}$ is a subset of the first-order language over $\mathcal{L}$. First-order sentences are not able to distinguish models that are elementarily equivalent, so $\mathrm{ML} + \Sigma$ cannot distinguish them either. Finite models are isomorphic if and only if they are elementarily equivalent [Doe96, Theorem 2.3, Proposition 2.5]. But for infinite models, this is not necessarily the case. So in the general case, elementary equivalence might be enough instead of the isomorphism requirement mentioned above.

3.2 $\mathbf{ML} + \Sigma$ and $\mathbf{ML} + E + \downarrow$

In the previous section we gave an intuition for the expressiveness of Σ-formulas. We will now turn this intuition into strict theory. Obviously $\mathrm{ML} + \Sigma \subseteq \mathrm{ML} + E + \downarrow$, as Σ is equivalent to $E\downarrow$. Recall that the expressive capabilities we listed mostly depended on the current world already being bound to a variable. This state could be achieved for a variable x simply by prefixing a formula with $\downarrow x$, which we cannot do in $\mathrm{ML} + \Sigma$. On the other hand, once we make a jump (in formulas: anything after an E), we can always bind the current world after the jump using Σ. And once we have that, we can also bind predecessors and successors.

This section is about this relation between $\mathrm{ML} + \Sigma$ and $\mathrm{ML} + E + \downarrow$. What exactly is the nature of the requirement to first use an E before we can bind any variables? We start with a formalisation of the mentioned properties when we already have a variable bound to the current world. Remember that $\mathrm{ML} + E + \downarrow$ is as strong as first-order logic.

Lemma 1. Let ϕ be a formula in $\mathrm{ML} + E + \downarrow$, and x be a world variable not in ϕ. Then there is a formula ϕ^x in $\mathrm{ML} + \Sigma$ such that for every model $M = (W, R, h)$, every $t \in W$ and every assignment g such that $g(x) = t$, we have $M, g, t \models \phi$ iff $M, g, t \models \phi^x$.

Proof. Suppose ϕ is a formula in $\mathrm{ML} + E + \downarrow$ and x a world variable not in ϕ. We define ϕ^x inductively by

$$
\begin{aligned}
(\top)^x &:= \top \\
(p)^x &:= p && \text{for all atoms } p \in \mathrm{PROP} \\
(i)^x &:= i && \text{for all nominals } i \in \mathrm{NOM} \\
(x_1)^x &:= x_1 && \text{for all world variables } x_1 \in \mathrm{VAR}, x_1 \neq x \\
(\neg\psi)^x &:= \neg\psi^x \\
(\psi_1 \wedge \psi_2)^x &:= \psi_1^x \wedge \psi_2^x \\
(\downarrow x_1 \psi)^x &:= \Sigma x_1(x \wedge \psi^x) \\
(\Diamond\psi)^x &:= \Sigma y \Sigma z(x \wedge \Diamond(y \wedge \psi^y)) \\
(E\psi)^x &:= \Sigma y(\psi^y)
\end{aligned}
$$

where y and z are variables not occurring in the respective subformulas ψ, ψ_1 and ψ_2.

We prove that for every model $M = (W, R, h)$, every $t \in W$ and every assignment g such that $g(x) = t$, we have $M, g, t \models \phi$ iff $M, g, t \models \phi^x$. We prove this via induction on the complexity of ϕ. In the base cases for $\top$, propositional symbols, nominals and world variables, the translation is equal to the original formula and therefore equivalent in all circumstances.

In the inductive step, the negation and conjunction are easy. For both cases, line four follows using the induction hypothesis, because the point of evaluation is still $g(x) = t$:

$$
\begin{array}{llcll}
 & M, g, t \models (\neg\psi)^x & (1) & & M, g, t \models (\psi_1 \wedge \psi_2)^x \\
\text{iff} & M, g, t \models \neg\psi^x & (2) & \text{iff} & M, g, t \models \psi_1^x \wedge \psi_2^x \\
\text{iff} & M, g, t \not\models \psi^x & (3) & \text{iff} & M, g, t \models \psi_1^x \text{ and } M, g, t \models \psi_2^x \\
\text{iff} & M, g, t \not\models \psi & (4) & \text{iff} & M, g, t \models \psi_1 \text{ and } M, g, t \models \psi_2 \\
\text{iff} & M, g, t \models \neg\psi & (5) & \text{iff} & M, g, t \models \psi_1 \wedge \psi_2
\end{array}
$$

The remaining three cases are more interesting:

$$
\begin{array}{lll}
 & M, g, t \models (\downarrow x_1 \psi)^x & \\
\text{iff} & M, g, t \models \Sigma x_1 (x \wedge \psi^x) & \\
\text{iff} & \exists u \in W, g' \overset{x_1}{\sim} g : g'(x_1) = u, M, g', u \models x \wedge \psi^x & \\
\text{iff} & \exists u \in W, g' \overset{x_1}{\sim} g : g'(x_1) = u = g'(x), M, g', u \models \psi^x & \\
\text{iff} & \exists u \in W, g' \overset{x_1}{\sim} g : g'(x_1) = u = g'(x), M, g', u \models \psi & \text{(by induction hypothesis)} \\
\text{iff} & \exists u \in W, g' \overset{x_1}{\sim} g : g'(x_1) = u = g(x), M, g', u \models \psi & \text{(because } g' \overset{x_1}{\sim} g) \\
\text{iff} & \exists u \in W, g' \overset{x_1}{\sim} g : g'(x_1) = u = t, M, g', u \models \psi & \text{(because } g(x) = t) \\
\text{iff} & \exists u \in W, g' \overset{x_1}{\sim} g : g'(x_1) = u = t, M, g', t \models \psi & \\
\text{iff} & \exists g' \overset{x_1}{\sim} g : g'(x_1) = t, M, g', t \models \psi & \\
\text{iff} & M, g, t \models \downarrow x_1 \psi &
\end{array}
$$

$$
\begin{array}{lll}
 & M, g, t \models (\Diamond \psi)^x & \\
\text{iff} & M, g, t \models \Sigma y \Sigma z (x \wedge \Diamond (y \wedge \psi^y)) & \\
\text{iff} & \exists u \in W, g' \overset{y}{\sim} g\text{: } g'(y) = u, M, g', u \models \Sigma z (x \wedge \Diamond (y \wedge \psi^y)) & \\
\text{iff} & \exists u, v \in W, g' \overset{y}{\sim} g, g'' \overset{z}{\sim} g'\text{: } g'(y) = u, g''(z) = v, M, g'', v \models x \wedge \Diamond (y \wedge \psi^y) & \\
\text{iff} & \exists u, v \in W, g' \overset{y}{\sim} g, g'' \overset{z}{\sim} g'\text{: } g'(y) = u, g''(z) = g''(x) = v, M, g'', v \models \Diamond (y \wedge \psi^y) & \\
\text{iff} & \exists u, v \in W, g' \overset{y}{\sim} g, g'' \overset{z}{\sim} g'\text{: } g'(y) = u, g''(z) = g''(x) = v, R(v, u), M, g'', u \models \psi^y & \\
\text{iff} & \exists u, v \in W, g' \overset{y}{\sim} g, g'' \overset{z}{\sim} g'\text{: } g'(y) = u, g''(z) = g''(x) = v, R(v, u), M, g'', u \models \psi & \\
 & & \text{(by induction hypothesis, as } g''(y) = u) \\
\text{iff} & \exists u, v \in W, g' \overset{y}{\sim} g\text{: } g'(y) = u, g'(x) = v, R(v, u), M, g', u \models \psi & \\
 & & \text{(because } g'' \overset{z}{\sim} g'\text{, } z \text{ does not occur in } \psi) \\
\text{iff} & \exists u, v \in W\text{: } g(x) = v, R(v, u), M, g, u \models \psi & \text{(because } g' \overset{y}{\sim} g\text{, } y \text{ does not occur in } \psi) \\
\text{iff} & \exists u \in W\text{: } R(t, u) \wedge M, g, u \models \psi & \text{(because } g(x) = t) \\
\text{iff} & M, g, t \models \Diamond \psi &
\end{array}
$$

$$
\begin{array}{lll}
 & M, g, t \models (E\psi)^x & \\
\text{iff} & M, g, t \models \Sigma y (\psi^y) & \\
\text{iff} & \exists u \in W, g' \overset{y}{\sim} g : g'(y) = u, M, g', u \models \psi^y & \\
\text{iff} & \exists u \in W, g' \overset{y}{\sim} g : g'(y) = u, M, g', u \models \psi & \text{(by induction hypothesis, as } g'(y) = u) \\
\text{iff} & \exists u \in W : M, g, u \models \psi & \text{(because } y \text{ does not occur in } \psi, g' \overset{y}{\sim} g) \\
\text{iff} & M, g, t \models E\psi &
\end{array}
$$

So for all possible formats of ϕ we proved that for every model $M = (W, R, h)$, every $t \in W$ and every assignment g such that $g(x) = t$, we have $M, g, t \models \phi$ iff $M, g, t \models \phi^x$. $\square$

If we start a formula with E, the current world does not matter, so the requirement on g is not necessary anymore:

Theorem 5. Let $E\phi$ be a formula in $\mathrm{ML} + E + \downarrow$, and x be a world variable not in ϕ. Then the formula $\psi := \Sigma x (\phi)^x$ in $\mathrm{ML} + \Sigma$ is such that for every model $M = (W, R, h)$, every world $t \in W$ and every assignment g and every world t: $M, g, t \models E\phi$ iff $M, g, t \models \psi$.

Proof. Let ϕ be a formula in $\mathrm{ML}+E+\downarrow$ and x a world variable not occurring in ϕ, then it holds:

$$\begin{aligned}
& M,g,t \models E\phi && \\
\text{iff} \quad & \exists u \in W : M,g,u \models \phi && \text{(Semantics of } E) \\
\text{iff} \quad & \exists u \in W, g' \overset{x}{\sim} g, g'(x) = u : M,g',u \models \phi && (x \text{ does not occur in } \phi) \\
\text{iff} \quad & \exists u \in W, g' \overset{x}{\sim} g, g'(x) = u : M,g',u \models \phi^x && \text{(using Lemma 1)} \\
\text{iff} \quad & M,g,t \models \Sigma x(\phi^x) && \text{(Semantics of } \Sigma)
\end{aligned}$$

□

Remark. The outlined construction does not always give the shortest equivalent formula in $\mathrm{ML}+\Sigma$ because a leading E may make it possible to simplify the translation of certain ϕ formulas. For example, $E{\downarrow}yp$ would be translated to $\Sigma x({\downarrow}yp)^x = \Sigma x \Sigma x_1(x \wedge p)$, but the shortest equivalent formula in $\mathrm{ML}+\Sigma$ is obviously Σyp.

Looking at it in the opposite way, what is the fragment of $\mathrm{ML}+\Sigma$ within $\mathrm{ML}+E+\downarrow$?

Theorem 6. Any formula ϕ in $\mathrm{ML}+\Sigma$ is equivalent to a formula ϕ' built from atoms, nominals, world variables, $\top$ and formulas of the form $E\chi$, where $\chi \in \mathrm{ML}+E+\downarrow$, using only $\neg, \wedge$ and $\Diamond$.

Proof. Via induction on the complexity of ϕ. In the base case, atoms, nominals, world variables and $\top$ are already of the required form. In the inductive case, if $\phi = \neg\psi$ then ψ fulfils the inductive hypothesis, so there is a formula ψ' of the form described that is equivalent to ψ. But then $\phi' := \neg\psi'$ is also of the required form and equivalent to ϕ. The cases for $\phi = \psi_1 \wedge \psi_2$ and $\phi = \Diamond\psi$ are similar.

The interesting case is $\phi = \Sigma x\psi$. Again, ψ fulfils the inductive hypothesis and so there exists a formula ψ' that is equivalent to ψ and of the required form, so ϕ is equivalent to $\Sigma x\psi'$. By definition of the Σ operator, $\phi' := E{\downarrow}x\psi'$ is then also equivalent to ϕ. As ϕ' is of the form required, this concludes the theorem. □

The proof gives a translation, which is very easy to do. The primary observation is that all $\downarrow$ operators appear after E. This is already useful for our upcoming characterisations, but we can still improve the result. As $E\chi$ does not depend on the world we evaluate it at, there is no need to have it following a Diamond:

Lemma 2. Let $\phi(q_1, \ldots, q_k)$ be a formula built from atoms, nominals, world variables and $\top$ using only $\neg, \wedge$ and $\Diamond$, where $q_1, \ldots, q_k$ are atoms (there might also be other atoms in ϕ besides the q_i). Further let $\psi_1, \ldots, \psi_k$ be formulas in $\mathrm{ML}+E+\downarrow$. Then $\phi(E\psi_1, \ldots, E\psi_k)$ is equivalent to the formula

$$\phi' = \bigwedge_{S \subseteq \{1,\ldots,k\}} \left[\left(\bigwedge_{i \in S} E\psi_i \wedge \bigwedge_{i \in \{1,\ldots,k\} \setminus S} \neg E\psi_i \right) \to \phi(Q_1^S, \ldots, Q_k^S) \right]$$

where $Q_i^S = \begin{cases} \top & \text{if } i \in S \\ \bot & \text{otherwise} \end{cases}$

Proof. For the duration of this proof, let $\equiv$ denote equivalence of hybrid formulas, i. e. $\phi_1 \equiv \phi_2$ if and only if for all Kripke models M, all assignments g and all worlds $t \in |M|$ it holds $M,g,t \models \phi_1$ iff $M,g,t \models \phi_2$. As conventionally done, we define $\bigwedge_{i \in \emptyset} y_i$ as $\top$ for any formula y_i.

We prove the theorem via induction on k. For $k = 0$, we have:

$$\begin{aligned}
\phi_0 &= \phi \qquad \text{(no explicit arguments)} \\
\phi_0' &= \bigwedge_{S \subseteq \emptyset} \left[\left(\bigwedge_{i \in S} E\psi_i \wedge \bigwedge_{i \in \emptyset \setminus S} \neg E\psi_i \right) \to \phi_0 \right] \\
&= \left(\bigwedge_{i \in \emptyset} E\psi_i \wedge \bigwedge_{i \in \emptyset \setminus \emptyset} \neg E\psi_i \right) \to \phi_0 \\
&\equiv \phi_0
\end{aligned}$$

We now show the inductive step from k to $k+1$. Let $\phi_{k+1} = \phi_{k+1}(E\psi_1, \ldots, E\psi_{k+1})$. Then

$$\begin{aligned}
\phi'_{k+1} = & \bigwedge_{S \subseteq \{1,\ldots,k+1\}} \left[\left(\bigwedge_{i \in S} E\psi_i \wedge \bigwedge_{i \in \{1,\ldots,k+1\} \setminus S} \neg E\psi_i \right) \to \phi_{k+1}(Q_1^S, \ldots, Q_{k+1}^S) \right] \\
\equiv & \bigwedge_{S \subseteq \{1,\ldots,k\}} \left[\left(\bigwedge_{i \in S} E\psi_i \wedge \bigwedge_{i \in \{1,\ldots,k+1\} \setminus S} \neg E\psi_i \right) \to \phi_{k+1}(Q_1^S, \ldots, Q_{k+1}^S) \right] \\
& \wedge \bigwedge_{\substack{S \subseteq \{1,\ldots,k\} \\ S' := S \cup \{k+1\}}} \left[\left(\bigwedge_{i \in S'} E\psi_i \wedge \bigwedge_{i \in \{1,\ldots,k+1\} \setminus S'} \neg E\psi_i \right) \to \phi_{k+1}(Q_1^{S'}, \ldots, Q_{k+1}^{S'}) \right] \\
\equiv & \bigwedge_{S \subseteq \{1,\ldots,k\}} \left[\left(\bigwedge_{i \in S} E\psi_i \wedge \neg E\psi_{k+1} \wedge \bigwedge_{i \in \{1,\ldots,k\} \setminus S} \neg E\psi_i \right) \to \phi_{k+1}(Q_1^S, \ldots, Q_k^S, \bot) \right] \\
& \wedge \bigwedge_{S \subseteq \{1,\ldots,k\}} \left[\left(\bigwedge_{i \in S} E\psi_i \wedge E\psi_{k+1} \wedge \bigwedge_{i \in \{1,\ldots,k\} \setminus S} \neg E\psi_i \right) \to \phi_{k+1}(Q_1^S, \ldots, Q_k^S, \top) \right]
\end{aligned}$$

We handle both parts simultaneously, so let $A \in \{E\psi_{k+1}, \neg E\psi_{k+1}\}$ and $B \in \{\top, \bot\}$, then

$$\begin{aligned}
& \bigwedge_{S \subseteq \{1,\ldots,k\}} \left[\left(\bigwedge_{i \in S} E\psi_i \wedge A \wedge \bigwedge_{i \in \{1,\ldots,k\} \setminus S} \neg E\psi_i \right) \to \phi_{k+1}(Q_1^S, \ldots, Q_k^S, B) \right] \\
\equiv & \bigwedge_{S \subseteq \{1,\ldots,k\}} \left[\neg \left(\bigwedge_{i \in S} E\psi_i \wedge A \wedge \bigwedge_{i \in \{1,\ldots,k\} \setminus S} \neg E\psi_i \right) \vee \phi_{k+1}(Q_1^S, \ldots, Q_k^S, B) \right] \\
\equiv & \bigwedge_{S \subseteq \{1,\ldots,k\}} \left[\left(\bigvee_{i \in S} \neg E\psi_i \vee \neg A \vee \bigvee_{i \in \{1,\ldots,k\} \setminus S} E\psi_i \right) \vee \phi_{k+1}(Q_1^S, \ldots, Q_k^S, B) \right] \\
\equiv & \bigwedge_{S \subseteq \{1,\ldots,k\}} \left[\neg A \vee \left(\bigvee_{i \in S} \neg E\psi_i \vee \bigvee_{i \in \{1,\ldots,k\} \setminus S} E\psi_i \right) \vee \phi_{k+1}(Q_1^S, \ldots, Q_k^S, B) \right] \\
\equiv & \neg A \vee \bigwedge_{S \subseteq \{1,\ldots,k\}} \left[\left(\bigvee_{i \in S} \neg E\psi_i \vee \bigvee_{i \in \{1,\ldots,k\} \setminus S} E\psi_i \right) \vee \phi_{k+1}(Q_1^S, \ldots, Q_k^S, B) \right] \\
\equiv & \neg A \vee \bigwedge_{S \subseteq \{1,\ldots,k\}} \left[\left(\bigwedge_{i \in S} E\psi_i \wedge \bigwedge_{i \in \{1,\ldots,k\} \setminus S} \neg E\psi_i \right) \to \phi_{k+1}(Q_1^S, \ldots, Q_k^S, B) \right] \\
\equiv & A \to \bigwedge_{S \subseteq \{1,\ldots,k\}} \left[\left(\bigwedge_{i \in S} E\psi_i \wedge \bigwedge_{i \in \{1,\ldots,k\} \setminus S} \neg E\psi_i \right) \to \phi_{k+1}(Q_1^S, \ldots, Q_k^S, B) \right]
\end{aligned}$$

Applying this equivalence to ϕ'_{k+1} yields:

$$\begin{aligned}
\phi'_{k+1} \equiv & \left(\neg E\psi_{k+1} \to \bigwedge_{S \subseteq \{1,\ldots,k\}} \left[\left(\bigwedge_{i \in S} E\psi_i \wedge \bigwedge_{i \in \{1,\ldots,k\} \setminus S} \neg E\psi_i \right) \to \phi_{k+1}(Q_1^S, \ldots, Q_k^S, \bot) \right] \right) \\
& \wedge \left(E\psi_{k+1} \to \bigwedge_{S \subseteq \{1,\ldots,k\}} \left[\left(\bigwedge_{i \in S} E\psi_i \wedge \bigwedge_{i \in \{1,\ldots,k\} \setminus S} \neg E\psi_i \right) \to \phi_{k+1}(Q_1^S, \ldots, Q_k^S, \top) \right] \right)
\end{aligned}$$

The last parameter of ϕ_{k+1} is constant, so we can in-line it to get two formulas with only k parameters:

$$\equiv \left(\neg E\psi_{k+1} \to \bigwedge_{S\subseteq\{1,\ldots,k\}} \left[\left(\bigwedge_{i\in S} E\psi_i \wedge \bigwedge_{i\in\{1,\ldots,k\}\setminus S} \neg E\psi_i\right) \to \phi_k^1(Q_1^S,\ldots,Q_k^S)\right]\right)$$
$$\wedge \left(E\psi_{k+1} \to \bigwedge_{S\subseteq\{1,\ldots,k\}} \left[\left(\bigwedge_{i\in S} E\psi_i \wedge \bigwedge_{i\in\{1,\ldots,k\}\setminus S} \neg E\psi_i\right) \to \phi_k^2(Q_1^S,\ldots,Q_k^S)\right]\right)$$

Then $\phi_k^1(q_1,\ldots,q_k)$ and $\phi_k^2(q_1,\ldots,q_k)$ are also built from atoms, nominals, world variables and $\top$ using only $\neg, \wedge$ and $\Diamond$. Applying the inductive hypothesis twice gives us

$$\begin{aligned}
&\equiv \big(\neg E\psi_{k+1} \to \phi_k^1(E\psi_1,\ldots,E\psi_k)\big)\\
&\quad\wedge \big(E\psi_{k+1} \to \phi_k^2(E\psi_1,\ldots,E\psi_k)\big)\\
&\equiv (\neg E\psi_{k+1} \to \phi_{k+1}(E\psi_1,\ldots,E\psi_k,\bot))\\
&\quad\wedge (E\psi_{k+1} \to \phi_{k+1}(E\psi_1,\ldots,E\psi_k,\top))
\end{aligned}$$

We now show that for all Kripke models M, for all worlds $t \in |M|$ and all assignments g it holds $M, g, t \models \phi_{k+1}$ iff $M, g, t \models \phi'_{k+1}$. So let M be a Kripke model, $t \in |M|$ and g an assignment. First notice that the Standard Translation of $E\psi_{k+1}$ is $\exists z \mathrm{ST}_z(\psi_{k+1})$ and thus its truth value does not depend on the world we evaluate it at. Therefore we have exactly one of the following two cases:

1. It holds $\forall s \in |M|$: $M, g, s \models E\psi_{k+1}$ $(*)$.
   Hence $\forall s \in |M|$: $M, g, s \models \neg E\psi_{k+1} \to \phi_{k+1}(E\psi_1,\ldots,E\psi_k,\bot)$ $(**)$. So

$$\begin{aligned}
&M,g,t \models \phi'_{k+1}\\
\text{iff}\quad &M,g,t \models (\neg E\psi_{k+1} \to \phi_{k+1}(E\psi_1,\ldots,E\psi_k,\bot))\\
&\qquad\wedge (E\psi_{k+1} \to \phi_{k+1}(E\psi_1,\ldots,E\psi_k,\top)) && \text{(by } \equiv\text{, see above)}\\
\text{iff}\quad &M,g,t \models E\psi_{k+1} \to \phi_{k+1}(E\psi_1,\ldots,E\psi_k,\top) && \text{(because of } (**))\\
\text{iff}\quad &M,g,t \models \phi_{k+1}(E\psi_1,\ldots,E\psi_k,\top) && \text{(because of } (*))\\
\text{iff}\quad &M,g,t \models \phi_{k+1}(E\psi_1,\ldots,E\psi_k,E\psi_{k+1}) && \text{(see below)}
\end{aligned}$$

   The last step is a simple induction on the structure of ϕ_{k+1}. None of the operators handled during the inductive step changes the assignment g. Therefore, in the base case handling the formula $E\psi_{k+1}$, the assignment is still g and we can use $(*)$.

2. It holds $\forall s \in |M|$: $M, g, s \not\models E\psi_{k+1}$ $(*)$.
   Hence $\forall s \in |M|$: $M, g, s \models E\psi_{k+1} \to \phi_{k+1}(E\psi_1,\ldots,E\psi_k,\top)$ $(**)$. So

$$\begin{aligned}
&M,g,t \models \phi'_{k+1}\\
\text{iff}\quad &M,g,t \models (\neg E\psi_{k+1} \to \phi_{k+1}(E\psi_1,\ldots,E\psi_k,\bot))\\
&\qquad\wedge (E\psi_{k+1} \to \phi_{k+1}(E\psi_1,\ldots,E\psi_k,\top)) && \text{(by } \equiv\text{, see above)}\\
\text{iff}\quad &M,g,t \models \neg E\psi_{k+1} \to \phi_{k+1}(E\psi_1,\ldots,E\psi_k,\bot) && \text{(because of } (**))\\
\text{iff}\quad &M,g,t \models \phi_{k+1}(E\psi_1,\ldots,E\psi_k,\bot) && \text{(because of } (*))\\
\text{iff}\quad &M,g,t \models \phi_{k+1}(E\psi_1,\ldots,E\psi_k,E\psi_{k+1}) && \text{(see below)}
\end{aligned}$$

   The last step is a simple induction on the structure of ϕ_{k+1}. None of the operators handled during the inductive step changes the assignment g. Therefore, in the base case handling the formula $E\psi_{k+1}$, the assignment is still g and we can use $(*)$.

This concludes the proof of the inductive claim and thus the lemma. $\square$

Theorem 7. Any formula ϕ in $\mathrm{ML}+\Sigma$ is equivalent to a formula ϕ' built from

- hybrid formulas using only modal operators (i.e. essentially modal formulas, but including world variables) and
- formulas of the form $E\chi$, where $\chi \in \mathrm{ML}+E+\downarrow$,

using only $\neg$ and $\wedge$.

Proof. We use Theorem 6 to get a formula equivalent to ϕ that is built from atoms, nominals, world variables, $\top$ and formulas of the form $E\chi$ where $\chi \in \mathrm{ML}+E+\downarrow$, using only $\neg, \wedge$ and $\Diamond$. The only difference to the formula ϕ' we require here is that formulas of the form $E\chi$ might appear in the scope of a $\Diamond$. We list as explicit parameters $E\chi_1, \ldots, E\chi_k$ all subformulas $E\chi_i$ of ϕ that do not appear below another E in the formation tree of ϕ, so then we have $\phi = \phi(E\chi_1, \ldots, E\chi_k)$. Then the formula $\phi(q_1, \ldots, q_k)$ with additional atoms q_i is built from atoms, nominals, world variables and $\top$ using only $\neg, \wedge$ and $\Diamond$. Now we use Lemma 2 to find the formula ϕ' which is equivalent to $\phi(E\chi_1, \ldots, E\chi_k)$ and built from hybrid formulas using only modal operators and formulas of the form $E\chi$ ($\chi \in \mathrm{ML}+E+\downarrow$), using only $\neg$ and $\wedge$. □

Note that the Standard Translation of formulas of the form $E\chi$ with $\chi \in \mathrm{ML}+E+\downarrow$ are first-order sentences. As Σ-sentences do not contain any free variables, their modal part according to Theorem 7 is an actual modal formula (without world variables). So essentially sentences in $\mathrm{ML}+\Sigma$ are equivalent to a boolean combination of first-order sentences and modal formulas.

3.3 First-Order Sentences and Hybrid Languages

The Standard Translation of hybrid sentences only has one free variable. If moreover the sentence starts with E, its Standard Translation is a first-order sentence. We will see that this mapping is surjective, i.e. that every first-order sentence of the correspondence language is equivalent to a sentence $E\phi \in \mathrm{ML}+E+\downarrow$. For this purpose we define the following translation. A similar translation has been mentioned towards the end of [BS95, Section 5].

Definition 27 (Hybrid Translation)**.** Let α be an $\mathcal{L}$-formula. We define the hybrid translation $\mathrm{HT}(a)$ inductively as follows:

$\mathrm{HT}(\top) := \top$		
$\mathrm{HT}(P(x)) := E(x \wedge p)$	where x is a variable or a constant symbol
$\mathrm{HT}(R(x,y)) := E(x \wedge \Diamond y)$	where x and y are variables or constant symbols
$\mathrm{HT}(x = y) := E(x \wedge y)$	where x and y are variables or constant symbols
$\mathrm{HT}(\psi_1 \wedge \psi_2) := \mathrm{HT}(\psi_1) \wedge \mathrm{HT}(\psi_2)$	where ψ_1 and ψ_2 are $\mathcal{L}$-formulas
$\mathrm{HT}(\neg\psi) := \neg\mathrm{HT}(\psi)$	where ψ is an $\mathcal{L}$-formula
$\mathrm{HT}(\exists x\psi) := E{\downarrow}x\mathrm{HT}(\psi)$	where x is a variable and ψ is an $\mathcal{L}$-formula

Lemma 3. Let α be an $\mathcal{L}$-formula, $M = (W, R, h)$ a Kripke model, g an assignment and M' the first-order structure induced by M. Then for all $t \in W$ it holds: $M', g \models \alpha$ iff $M, g, t \models \mathrm{HT}(\alpha)$.

Proof. Via induction on the complexity of α. Let t be any world in W. The case for $\alpha = \top$ is obvious. The remaining cases are as follows, where x and y are variables or constant symbols (note that these correspond to world symbols in hybrid logic).

$$
\begin{array}{rll}
M', g \models P(x) & \text{iff} & h(p)([h,g](x)) \\
& \text{iff} & M, g, [h,g](x) \models p \\
& \text{iff} & \exists w \in W : [h,g](x) = w \text{ and } M, g, w \models p \\
& \text{iff} & \exists w \in W : M, g, w \models x \text{ and } M, g, w \models p \\
& \text{iff} & \exists w \in W : M, g, w \models x \wedge p \\
& \text{iff} & M, g, t \models E(x \wedge p) \\
& \text{iff} & M, g, t \models \mathrm{HT}(P(x)) \\
M', g \models R(x,y) & \text{iff} & R([h,g](x), [h,g](y)) \\
& \text{iff} & \exists w, v \in W\colon [h,g](x) = w, [h,g](y) = v, R(w,v) \\
& \text{iff} & \exists w \in W\colon [h,g](x) = w, \exists v \in W\colon R(w,v) \wedge [h,g](y) = v \\
& \text{iff} & \exists w \in W\colon [h,g](x) = w, \exists v \in W\colon R(w,v) \wedge M, g, v \models y \\
& \text{iff} & \exists w \in W\colon [h,g](x) = w, M, g, w \models \Diamond y \\
& \text{iff} & \exists w \in W\colon M, g, w \models x \wedge \Diamond y \\
& \text{iff} & M, g, t \models E(x \wedge \Diamond y) \\
& \text{iff} & M, g, t \models \mathrm{HT}(R(x,y)) \\
M', g \models x = y & \text{iff} & [h,g](x) = [h,g](y) \\
& \text{iff} & \exists w \in W\colon w = [h,g](x) \wedge w = [h,g](y) \\
& \text{iff} & \exists w \in W\colon (M, g, w \models x) \wedge (M, g, w \models y) \\
& \text{iff} & \exists w \in W\colon M, g, w \models x \wedge y \\
& \text{iff} & M, g, t \models E(x \wedge y) \\
& \text{iff} & M, g, t \models \mathrm{HT}(x = y) \\
M', g \models \psi_1 \wedge \psi_2 & \text{iff} & (M', g \models \psi_1) \wedge (M', g \models \psi_2) \\
& \text{iff} & (M, g, t \models \mathrm{HT}(\psi_1)) \wedge (M, g, t \models \mathrm{HT}(\psi_2)) \qquad \text{(by induction hypothesis)} \\
& \text{iff} & M, g, t \models \mathrm{HT}(\psi_1) \wedge \mathrm{HT}(\psi_2) \\
& \text{iff} & M, g, t \models \mathrm{HT}(\psi_1 \wedge \psi_2) \\
M', g \models \neg\psi & \text{iff} & M', g \not\models \psi \\
& \text{iff} & M, g, t \not\models \mathrm{HT}(\psi) \qquad \text{(by induction hypothesis)} \\
& \text{iff} & M, g, t \models \neg\mathrm{HT}(\psi) \\
& \text{iff} & M, g, t \models \mathrm{HT}(\neg\psi)
\end{array}
$$

In the last case, we require x to be a variable:

$$
\begin{array}{rll}
M', g \models \exists x\psi & \text{iff} & \exists g' \overset{x}{\sim} g\colon M', g' \models \psi \\
& \text{iff} & \exists g' \overset{x}{\sim} g\colon M, g', g'(x) \models \mathrm{HT}(\psi) \qquad \text{(by induction hypothesis)} \\
& \text{iff} & \exists v \in W, g' \overset{x}{\sim} g\colon g'(x) = v \wedge M, g', v \models \mathrm{HT}(\psi) \\
& \text{iff} & \exists v \in W\colon M, g, v \models {\downarrow} x \mathrm{HT}(\psi) \\
& \text{iff} & M, g, t \models E{\downarrow} x \mathrm{HT}(\psi) \\
& \text{iff} & M, g, t \models \mathrm{HT}(\exists x\psi)
\end{array}
$$

□

Theorem 8. Let α be a first-order sentence over $\mathcal{L}$. Then there is a sentence $E\phi \in \mathrm{ML} + E + {\downarrow}$ such that for all Kripke models M with induced first-order structures M' and all $t \in |M|$ we have: $M, t \models E\phi$ if and only if $M' \models \alpha$.

Proof. Let α be a first-order sentence over $\mathcal{L}$. Define $\phi := \mathrm{HT}(\alpha)$, then $E\phi$ is a sentence in $\mathrm{ML} + E + \downarrow$. Let $M = (W, R, h)$ be a Kripke model, g an assignment, $t \in W$ and M' the first-order structure induced by M. Then it holds:

$$
\begin{aligned}
M, t \models E\phi \quad & \text{iff} \quad M, g, t \models E\phi && (\phi \text{ is a sentence}) \\
& \text{iff} \quad M, g, t \models E\,\mathrm{HT}(\alpha) && \\
& \text{iff} \quad \exists u \in W\colon M, g, u \models \mathrm{HT}(\alpha) && (\text{definition of } E) \\
& \text{iff} \quad \exists u \in W\colon M', g \models \alpha && (\text{Lemma 3}) \\
& \text{iff} \quad M', g \models \alpha && (\text{independent of } u) \\
& \text{iff} \quad M' \models \alpha && (\alpha \text{ is a sentence})
\end{aligned}
$$

□

The next two results follow directly from our previous theorems, but their ideas have already been mentioned by Blackburn and Seligman in [BS95, Proposition 5.3, Corollary 5.1].

Corollary 1. Let α be a first-order sentence over $\mathcal{L}$. Then there is a sentence $\phi \in \mathrm{ML} + \Sigma$ such that for all Kripke models M with induced first-order structures M' and all $t \in |M|$ we have: $M, t \models \phi$ if and only if $M' \models \alpha$.

Proof. Follows directly from Theorem 8, Theorem 5 and the fact that the resulting formula in $\mathrm{ML} + \Sigma$ is a sentence. □

Corollary 2. The validity problem of $\mathrm{ML} + \Sigma$ is undecidable.

Proof. Let α be a first-order sentence over $\mathcal{L}$. Deriving ϕ as in Corollary 1, the result implies that ϕ is valid if and only if α is valid (note that all first-order structures over $\mathcal{L}$ are induced by some Kripke model). Suppose for a contradiction that the problem was decidable for $\mathrm{ML} + \Sigma$, then we could decide whether ϕ and hence also α is valid. But since the validity of first-order $\mathcal{L}$-sentences is undecidable [BGG97, Theorem 4.0.1], this is a contradiction. □

Chapter 4

Semantic Characterisations of $\mathrm{ML} + \Sigma$

The preceding investigations are the basis of the characterisations we show in this chapter. We show two characterisations using a form of bisimulation, namely Σ-*bisimulation* and *strong* Σ-*bisimulation*, and one using a game (Σ-*game*). Sections 4.1 to 4.3 will introduce them one by one and establish implications between them. That the three notions are in fact characterisations will be shown in Section 4.4 using a ring proof that builds on these implications.

4.1 Σ-Bisimilarity

Definition 28 (Σ-bisimulation)**.** A Σ-*bisimulation* between two pointed Kripke models (M_1, t_1) and (M_2, t_2) with $M_1 = (W_1, R_1, h_1)$ and $M_2 = (W_2, R_2, h_2)$ is a binary relation $B \subseteq W_1 \times W_2$ such that the following conditions hold:

- B is a bisimulation
- M_1 and M_2 are elementarily equivalent ($M_1 \equiv M_2$)

If there is a Σ-bisimulation between two pointed models (M_1, t_1) and (M_2, t_2), we say that (M_1, t_1) and (M_2, t_2) are Σ-*bisimilar*.

Lemma 4. Let $E\phi$ be a sentence in $\mathrm{ML} + E + \downarrow$ and $M_1 = (W_1, R_1, h_1)$ and $M_2 = (W_2, R_2, h_2)$ two elementarily equivalent Kripke models. Then for all worlds $t_1 \in W_1$ and $t_2 \in W_2$ it holds $M_1, t_1 \models E\phi$ iff $M_2, t_2 \models E\phi$.

Proof. Let M_1' and M_2' be the first-order structures induced by M_1 and M_2, respectively, and x a variable not in ϕ. For all $t_1 \in W_1$ and $t_2 \in W_2$, it holds:

$$\begin{aligned}
M_1, t_1 \models E\phi \quad & \text{iff} \quad \exists u_1 \in W_1\colon M_1, u_1 \models \phi \\
& \text{iff} \quad \exists u_1 \in W_1\colon M_1' \models \mathrm{ST}_x(\phi)[u_1] \\
& \text{iff} \quad M_1' \models \exists x\, \mathrm{ST}_x(\phi) \\
& \text{iff} \quad M_2' \models \exists x\, \mathrm{ST}_x(\phi) \qquad \text{(by elementary equivalence)} \\
& \text{iff} \quad \exists u_2 \in W_2\colon M_2' \models \mathrm{ST}_x(\phi)[u_2] \\
& \text{iff} \quad \exists u_2 \in W_2\colon M_2, u_2 \models \phi \\
& \text{iff} \quad M_2, t_2 \models E\phi
\end{aligned}$$

$\square$

Theorem 9. Let ϕ be a sentence in $\mathrm{ML} + \Sigma$ and B a Σ-bisimulation between the pointed Kripke models (M_1, t_1) and (M_2, t_2). Then $M_1, t_1 \models \phi$ if and only if $M_2, t_2 \models \phi$.

Proof. Let $M_1 = (W_1, R_1, h_1)$ and $M_2 = (W_2, R_2, h_2)$. By Theorem 6, ϕ is equivalent to a formula ϕ' built from formulas of the form

- atoms
- nominals
- world variables
- $\top$
- $E\chi$, where $\chi \in \mathrm{ML} + E + \downarrow$

using only $\neg, \wedge$ and $\Diamond$. Because ϕ is a sentence, ϕ' is also a sentence. Therefore, all variables in ϕ' are bound, i. e. they only appear after $\downarrow$, hence they must be part of a formula $E\chi$, $\chi \in \mathrm{ML} + E + \downarrow$. So effectively ϕ' is built from atoms, nominals, $\top$ and $E\chi$ ($\chi \in \mathrm{ML} + E + \downarrow$) using only $\neg, \wedge$ and $\Diamond$. We now prove the theorem for ϕ' via induction on the complexity of the formula. We regard $E\chi$ as a base case as we shall see, and therefore none of the cases we handle considers a binding operator. Therefore the substructures to which we apply the induction hypothesis are still sentences.

For $\phi' = p$ where $p \in \mathrm{PROP}$:

$$\begin{aligned} M_1, t_1 \models p \quad &\text{iff} \quad t_1 \in h_1(p) \\ &\text{iff} \quad t_2 \in h_2(p) \qquad \text{(because } t_1 \text{ and } t_2 \text{ agree on atoms, as they are bisimilar)} \\ &\text{iff} \quad M_2, t_2 \models p \end{aligned}$$

For $\phi' = i$ where $i \in \mathrm{NOM}$:

$$\begin{aligned} M_1, t_1 \models i \quad &\text{iff} \quad t_1 = h_1(i) \\ &\text{iff} \quad t_2 = h_2(i) \qquad \text{(because } t_1 \text{ and } t_2 \text{ agree on nominals, as they are bisimilar)} \\ &\text{iff} \quad M_2, t_2 \models i \end{aligned}$$

For $\phi' = \top$:

$$M_1, t_1 \models \top \quad \text{iff} \quad M_2, t_2 \models \top$$

The case $\phi' = E\chi$, $\chi \in \mathrm{ML} + E + \downarrow$ follows from Lemma 4.

For $\phi' = \neg\psi$ and ψ is of the required form, we have:

$$\begin{aligned} M_1, t_1 \models \neg\psi \quad &\text{iff} \quad M_1, t_1 \not\models \psi \\ &\text{iff} \quad M_2, t_2 \not\models \psi \qquad \text{(by induction hypothesis)} \\ &\text{iff} \quad M_2, t_2 \models \neg\psi \end{aligned}$$

For $\phi' = \psi_1 \wedge \psi_2$ and ψ_1 and ψ_2 are of the required form, we have:

$$\begin{aligned} M_1, t_1 \models \psi_1 \wedge \psi_2 \quad &\text{iff} \quad M_1, t_1 \models \psi_1 \text{ and } M_1, t_1 \models \psi_2 \\ &\text{iff} \quad M_2, t_2 \models \psi_1 \text{ and } M_2, t_2 \models \psi_2 \qquad \text{(by induction hypothesis)} \\ &\text{iff} \quad M_2, t_2 \models \psi_1 \wedge \psi_2 \end{aligned}$$

For $\phi' = \Diamond\psi$ and ψ is of the required form, we have:

$$\begin{aligned} M_1, t_1 \models \Diamond\psi \quad &\text{iff} \quad \exists u_1 \in W_1\text{: } R(t_1, u_1) \text{ and } M_1, u_1 \models \psi \\ &\text{iff} \quad \exists u_2 \in W_2\text{: } R(t_2, u_2) \text{ and } M_2, u_2 \models \psi \\ &\qquad \text{(by induction hypothesis, } B(t_1, t_2) \text{ and forth/back with } B(u_1, u_2)\text{)} \\ &\text{iff} \quad M_2, t_2 \models \Diamond\psi \end{aligned}$$

$\square$

4.2 The Σ-Game

Games have been used in the past as an alternative and much more accessible way to characterise logics. Though they have clearly defined rules and are in this sense very strict, their informal nature makes it easy to think about specific cases or to find an idea for a proof. The most classical games of this sort, which are also closely related to the one we will define, are the bisimulation game for modal logic [vB10, Def. 3.5.1] and the Ehrenfeucht game for first-order logic [Doe96, Def. 3.45].

They are usually played on two models of the logic. Two players are in dispute over whether or not the logic can distinguish these models. For modal logic, this question corresponds to whether the models are bisimilar; for first-order logic whether they are equivalent for the infinitary logic. The player *Spoiler* tries to prove that the models can be distinguished, the player *Duplicator* tries to prove the opposite. Their possible moves correspond to the operators of a formula of the logic. If Spoiler wins after a finite number of rounds, his moves can be used to derive a formula that is true in one model, but not in the other.

In this section, we introduce the Σ-game, which will be a second characterisation for $\mathrm{ML} + \Sigma$. Just as the Σ-language combines aspects of first-order sentences and modal formulas, the Σ-game combines aspects of the bisimulation and Ehrenfeucht games.

Definition 29 (Σ-game)**.** Two players *Spoiler* and *Duplicator* play against each other in turns, starting with Spoiler. The playing field are two pointed Kripke models (M, m) and (N, n), for which Spoiler tries to find a Σ-sentence to distinguish them, while Duplicator tries to prove that they fulfil exactly the same Σ-sentences. They also play with pairs of pebbles where each pair has a different colour. The two white pebbles are a distinguished pair, i. e. they will be played in a different way than all other colours. When we speak of *coloured* pebbles, we mean those that are not white. Initially, the white pebbles are placed on the worlds m and n.

The game is played in rounds, each consisting of one move by Spoiler and then one by Duplicator. A round leads the game from one stage to another, and we usually denote such a stage by a pair $(s, t) \in |M| \times |N|$ indicating the two worlds with the white pebbles on them. As there might also be coloured pebbles in play, this does not fully specify the state of affairs in the game, but their positions will be clear from the context. Accordingly, a "Σ-game starting at (s, t)" is one starting with the white pebbles at worlds s and t.

The rounds are played as follows. If we are at a stage (x, y) of the game, then Spoiler starts by choosing between two possible moves:

1. (As in the standard modal bisimulation game, cf. [vB10, Def. 3.5.1]) Choose a world w as either x or y. Then choose an R-successor of w and move the white pebble from w to that successor.

2. Choose one of the two models. Choose any world w in that model and a *new* colour. Put a new pebble of that colour on w and move the white pebble in that model to w as well. We call this a "jump" move.

Duplicator must then answer with one of the following moves, corresponding to the one Spoiler made. That is, if Spoiler chose move 1, then Duplicator has to do a move of type 1 as well, and a move of type 2 otherwise. In both cases, Duplicator acts in just the model Spoiler did not use.

1. Let v be the world in $\{x, y\}$ Spoiler did not choose. Then choose an R-successor of v and move the white pebble from v to that successor.

2. Choose any world v in the model Spoiler did not use. Put a new pebble of the colour Spoiler chose in his move to that world, and also move the white pebble there.

If anyone cannot move, the other player wins. If during the course of the game and directly before Spoiler moves, the worlds with the white pebbles disagree on any atom, any nominal or on any other pebble colour present at that world, Spoiler wins. If Spoiler does not win after any finite number of rounds, Duplicator wins.

The Σ-game has two types of moves. If we only play type 2, we basically simulate an infinite Ehrenfeucht game [Doe96, Def. 3.45]. This makes it possible for Duplicator to transfer a possible winning strategy.

Lemma 5 (Strategy transfer)**.** Let (M, m) and (N, n) be two pointed Kripke models. If Duplicator has a winning strategy WS in the Σ-game starting at (m, n), then WS gives rise to a winning strategy for Duplicator in the infinite Ehrenfeucht game for first-order logic played on the induced first-order structures of (M, m) and (N, n).

Proof. Regard the normal Σ-game as being played on the induced first-order structures of the underlying Kripke models and compare it to the infinite Ehrenfeucht game for first-order logic. All moves Spoiler can make in the latter are also valid moves for Spoiler in an infinite Σ-game, namely "jump" moves. When Spoiler repeatedly does such a move in the Σ-game, Duplicator always has to respond with a type 2 move as well. All such moves of Duplicator are also valid in the Ehrenfeucht game for first-order logic. Of course the Σ-game has the additional white pebbles indicating the current worlds. But as we only make jump moves, these are already marked by another pair of coloured pebbles. Now let Duplicator use her winning strategy from the Σ-game in the first-order game. That is, suppose Spoiler chooses world w in the first-order game. If Spoiler chose w in the Σ-game by moving the white pebble there and putting a new coloured pebble on top, then Duplicator would choose a corresponding world v in the other model, move the white pebble there and put a pebble of the same colour on top. Then, in the first-order game, Duplicator chooses world v in response to Spoiler choosing w. We claim that this is a winning strategy for Duplicator in the first-order game, i. e. that at every finite stage of any game with Duplicator playing the strategy, the moves made so far constitute a local isomorphism between M and N (see [Doe96, Def. 3.3]). Suppose for contradiction that this is not the case, so in some Σ-game with only type 2 moves so far, at a finite stage k with played pebble pairs $(m_1, n_1), \ldots, (m_k, n_k)$ of colours 1 to k, the simple expansions $(M, m_1, \ldots, m_k)$ and $(N, n_1, \ldots, n_k)$, interpreting the constant symbols c_i as $h_1(c_i)$ and $h_2(c_i)$ respectively, disagree on an atomic sentence ϕ. We will distinguish all possible shapes of this sentence, but first we need the following facts. Let $C := \mathrm{NOM} \cup \{c_1, \ldots, c_k\}$.

(∗) For all worlds $w \in |M|$ and $v \in |N|$ it holds: If there is a $c \in C$ such that $w = h_1(c) \leftrightarrow v = h_2(c)$ does not hold, then Duplicator loses the game starting at (w, v).

This is so because

- If $c \in \mathrm{NOM}$, then $w = h_1(c) \not\leftrightarrow v = h_2(c)$ means that w and v disagree on the nominal c, so Duplicator loses immediately.
- If $c = c_i \in \{c_1, \ldots, c_k\}$, then $w = h_1(c) \not\leftrightarrow v = h_2(c)$ means that exactly one of the worlds w and v is marked with a pebble of colour i, so Duplicator loses.

From the converse of (∗) we get

(∗∗) If Duplicator has a winning strategy from (w, v), then $\forall c \in C$: $w = h_1(c) \leftrightarrow v = h_2(c)$.

Now we will look at the possible shapes of the atomic sentence ϕ.

1. The sentence is $c = c'$ for some $c, c' \in C$. W. l. o. g. we have $M \models (c = c')$, but $N \not\models (c = c')$. Then it holds:

$$
\begin{array}{rlcrl}
 & M \models (c = c') & \qquad\qquad & & N \not\models (c = c') \\
\text{iff} & h_1(c) = h_1(c') & & \text{iff} & h_2(c) \neq h_2(c')
\end{array}
$$

Now, in the Σ-game let Spoiler make a jump move to $w = h_1(c)$. If v is the world Duplicator jumps to, then Duplicator must still have a winning strategy from (w, v). Hence by $w = h_1(c)$ and (∗∗), we conclude that the world v is $h_2(c)$. But now there is the constant c' such that $w = h_1(c) = h_1(c')$, but $v = h_2(c) \neq h_2(c')$ and by (∗) Duplicator loses the game starting at (w, v), which is a contradiction to her having a winning strategy from there.

2. The sentence is $P(c)$ for some $c \in C$. Again, w. l. o. g. we have

$$\begin{array}{llcll} & M \models P(c) & & & N \not\models P(c) \\ \text{iff} & h_1(c) \in h_1(p) & & \text{iff} & h_2(c) \notin h_2(p) \end{array}$$

Suppose Spoiler chooses to make a jump move to $w = h_1(c)$ in the Σ-game. If v is the world Duplicator jumps to, then Duplicator must still have a winning strategy from (w, v). Hence by $w = h_1(c)$ and $(**)$, we conclude that the world v is $h_2(c)$. But then the worlds w and v disagree on the atom p, so Duplicator loses which is again a contradiction.

3. Finally, the sentence could be $R(c, c')$ for some $c, c' \in C$. Then w. l. o. g. it holds:

$$\begin{array}{llcll} & M \models R(c, c') & & & N \not\models R(c, c') \\ \text{iff} & (h_1(c), h_1(c')) \in R_M & & \text{iff} & (h_2(c), h_2(c')) \notin R_N \end{array}$$

Now let Spoiler jump to $h_1(c)$, then again by $(**)$ Duplicator has to jump to $h_2(c)$. Then let Spoiler do a type 1 move to $h_1(c')$, which he can do as there is a corresponding edge in R_M. Now Duplicator has to fulfil two conditions in her response:

   (a) go via an R_N edge (because she has to do a type 1 move)

   (b) end up at $h_2(c')$ (because otherwise the worlds disagree on constant c' and by $(*)$ she would lose)

   But since $(h_2(c), h_2(c')) \notin R_N$, this is not possible so Duplicator loses, contradicting her having a winning strategy.

All possible shapes of the atomic sentence on which the expansions disagree lead to a contradiction. Hence our assumption that such a sentence exists was wrong and therefore we conclude that at every finite stage of any game with Duplicator playing the strategy, the moves made so far constitute a local isomorphism between M and N. Thus, by [Doe96, Def. 3.45], the strategy is a winning strategy for Duplicator in the Ehrenfeucht game for first-order logic. □

Lemma 6. Let (M, m) and (N, n) be two pointed Kripke models. If Duplicator has a winning strategy in the infinite Σ-game with white pebbles at worlds m and n, then (M, m) and (N, n) are bisimilar.

Proof. Let $M = (W_M, R_M, h_M)$ and $N = (W_N, R_N, h_N)$. We show that (M, m) and (N, n) are bisimilar by proving that $B := \{(w, v) \in W_M \times W_N \mid$ Duplicator has a winning strategy in the infinite Σ-game starting at $(w, v)\}$ is a bisimulation between (M, m) and (N, n). By our premise, Duplicator has a winning strategy from (m, n) and thus $(m, n) \in B$. Now, let (x, y) be an arbitrary element of B. By definition of B, Duplicator has a winning strategy from (x, y). This implies that Duplicator has not already lost after 0 rounds, and so x and y must agree on atoms and on nominals. We prove the remaining conditions *back* and *forth*, starting with the latter. Hence suppose there is a world $x^* \in W_M$ such that $R_M(x, x^*)$. We have to prove the existence of an R_N-successor of y that is bisimilar to x^*. In a Σ-game starting from (x, y), let Spoiler choose model M and move the white pebble from x to x^* along the edge connecting them (move of type 1). Using her winning strategy, Duplicator can find an R_N-successor y^* of y to which she can move the white pebble in model N, such that she still has a winning strategy from (x^*, y^*) (otherwise her original winning strategy would be invalid). This means $(x^*, y^*) \in B$. So y^* is a world in W_N that fulfils both $R_N(y, y^*)$ and $B(x^*, y^*)$, and hence is the world required by the *forth* condition. The proof for the *back* property is similar. □

Theorem 10. Let (M, m) and (N, n) be two pointed Kripke models such that Duplicator has a winning strategy in the infinite Σ-game starting at (m, n). Then (M, m) and (N, n) are Σ-bisimilar.

Proof. Suppose that (M, m) and (N, n) are two pointed Kripke models such that Duplicator has a winning strategy in the infinite Σ-game starting at (m, n). By Lemma 6, (M, m) and (N, n) are bisimilar. By Lemma 5, Duplicator also has a winning strategy in the infinite first-order Ehrenfeucht game on the induced first-order structures. By [Doe96, Proposition 3.52], this implies $M \equiv N$. Combined with the bisimilarity of the pointed models, we know that (M, m) and (N, n) are Σ-bisimilar. □

4.3 Strong Σ-Bisimilarity

Now we come to a strengthened version of the Σ-bisimulation, which will be used by our third characterisation. It is based on bisimilarity and isomorphisms.

Definition 30. Let A and B be two first-order structures over $\mathcal{L}$. A bijective function $f\colon |A| \to |B|$ is called an isomorphism between A and B iff

1. For each k-ary relation symbol R of $\mathcal{L}$ with interpretations R^A and R^B in A and B, and for all $a_1, \ldots, a_k \in |A|$ it holds that $R^A(a_1, \ldots, a_k)$ iff $R^B(f(a_1), \ldots, f(a_k))$.

2. For each constant symbol i of $\mathcal{L}$ with interpretations i_A and i_B in A and B, it holds $f(i_A) = i_B$.

If there is an isomorphism between A and B, they are called isomorphic.

Let M and N be two Kripke models and M' and N' their respective induced first-order structures. If f is an isomorphism between M' and N', we also say it is an isomorphism between M and N. If there is an isomorphism between M and N, we call M and N isomorphic.

Theorem 11. Let f be an isomorphism between two Kripke models $M = (W_M, R_M, h_M)$ and $N = (W_N, R_N, h_N)$. Then for all worlds $t \in W_M$, it holds:

1. For each propositional atom p: $t \in h_M(p)$ iff $f(t) \in h_N(p)$

2. For each nominal i: $t = h_M(i)$ iff $f(t) = h_N(i)$

3. For all worlds $s \in W_M$: $R_M(s, t)$ iff $R_N(f(s), f(t))$

Proof. Let M' and N' be the induced first-order structures of M and N, respectively. Note that $|M| = |M'|$ and $|N| = |N'|$, and so $f : |M| \to |N|$ is still bijective. The above properties follow directly from the way we interpret the relational symbols and constant symbols in the induced first-order structures and the fact that these are isomorphic. □

Definition 31 (Strong Σ-bisimulation)**.** A *strong Σ-bisimulation* between two pointed Kripke models (M_1, t_1) and (M_2, t_2) with $M_1 = (W_1, R_1, h_1)$ and $M_2 = (W_2, R_2, h_2)$ is a binary relation $B \subseteq W_1 \times W_2$ such that the following conditions hold:

- B is a bisimulation

- M_1 and M_2 are isomorphic

If there is a strong Σ-bisimulation between two pointed models (M_1, t_1) and (M_2, t_2), we say that (M_1, t_1) and (M_2, t_2) are *strongly Σ-bisimilar.*

Theorem 12. Let (M, m) and (N, n) be two strongly Σ-bisimilar pointed Kripke models. Then Duplicator has a winning strategy in the infinite Σ-game starting from (m, n).

Proof. Suppose that (M, m) and (N, n) are two strongly bisimilar pointed Kripke models, i. e. they are bisimilar and M and N are isomorphic. Let $f\colon |M| \to |N|$ be an isomorphism. We need to show that Duplicator has a winning strategy in the infinite Σ-game starting from (m, n). We give a strategy for Duplicator and prove it is a winning strategy by proving inductively for any stage (s, t) of the game:

(a) If there have not been any jump moves, then (M, s) and (N, t) are bisimilar.

(b) If there has been at least one jump move, then $f(s) = t$.

(c) For all $s \in |M|$, s and $f(s)$ agree on all pebble colours (of coloured pebbles).

(d) Duplicator does not lose the game until this stage.

At the beginning of the game, there have not been any jump moves, so (b) is trivially fulfilled. Since $(s, t) = (m, n)$ is the initial stage, we also know that (M, s) and (N, t) are bisimilar. Therefore, s and t agree on all atoms and nominals. As furthermore no coloured pebbles have been played, we get points (c) and (d).

For the inductive case, first suppose there have not been any jump moves so far and Spoiler's move is also of type 1. If the game is at (s, t) before Spoiler's move, then by inductive hypothesis s and t are bisimilar. W. l. o. g. let Spoiler act in M and let s' be the R_M-successor of s he chooses. By bisimilarity, there is a world $t' \in N$ that is an R_N-successor of t and bisimilar to s'. This is the world Duplicator moves to in N, so point (a) is fulfilled. As no coloured pebbles are in the game yet, point (c) is fulfilled. Since moreover the bisimilarity of s' and t' implies agreement on atoms and nominals, Duplicator does not lose in this round, fulfilling point (d) given the inductive hypothesis. Point (b) is trivially true.

Now suppose there has been at least one jump move or the current move by Spoiler is a jump move. Suppose the game is at (s, t) before Spoiler's move and w. l. o. g. Spoiler acts in M and moves the white pebble to s' (using any move type). Then let Duplicator move to $t' = f(s')$. Why is this a valid move? If Spoiler's move was a jump move, then Duplicator can move to any world, so the move is valid. If Spoiler's move was of type 1, but there has been at least one jump move before, then by inductive hypothesis $t = f(s)$. As the move was of type 1, $R_M(s, s')$. As f is an isomorphism, $R_N(f(s), f(s'))$, so there is also an edge from $t = f(s)$ to $t' = f(s')$ and Duplicator's move is valid. Why does (s', t') fulfil the inductive claim? Point (a) is trivially fulfilled as we just did a jump move. Point (b) is fulfilled as we defined t' as $f(s')$. For point (c), notice that the possibly newly placed coloured pebbles are on two worlds connected by f, namely s' and $f(s')$, so all worlds connected by f agree on this pebble colour. By the inductive hypothesis, all worlds connected by f also agree on all previous pebble colours, fulfilling point (c). In particular, s' and $t' = f(s')$ agree on all pebble colours. Since moreover s' and t' agree on propositional atoms and nominals as the isomorphism f assigns s' to t', Duplicator does not lose the game at this stage. Together with the inductive hypothesis, this fulfils point (d).

Overall we proved that using the strategy given, Duplicator can always move and does not lose at any finite stage of the game. This means that the strategy is a winning strategy for Duplicator in the infinite Σ-game starting from (m, n). $\square$

Theorem 13. Let (M, m) and (N, n) be two *countable* pointed Kripke models. Duplicator has a winning strategy in the infinite Σ-game with white pebbles at worlds m and n if and only if (M, m) and (N, n) are strongly Σ-bisimilar.

Proof. For the left-to-right direction assume Duplicator has a winning strategy in the infinite Σ-game starting at (m, n). Then by Lemma 6, (M, m) and (N, n) are bisimilar. By Lemma 5, Duplicator's winning strategy yields a winning strategy in the infinite Ehrenfeucht game on the induced first-order structures. By [Doe96, Theorem 3.48] and the fact that M and N are countable we conclude that M and N are isomorphic. Therefore (M, m) and (N, n) are strongly Σ-bisimilar.

The right-to-left direction follows from Theorem 12. $\square$

So far, we related strong Σ-bisimulation to the Σ-game, the Σ-game to Σ-bisimulation and Σ-bisimulation to Σ-formulas, always going from a stronger notion to a weaker one. But if we restrict to what first-order formulas can say about the models, the notions are all equivalent as we shall see. To complement the ring proof, we need some additional concepts coming mostly from first-order logic.

Definition 32 (Invariance for (strong) Σ-bisimulation)**.** Let $\alpha(x)$ be an $\mathcal{L}$-formula with at most one free variable x. We say α is *invariant for (strong) Σ-bisimulation* if whenever two pointed Kripke models (M, m) and (N, n) are (strongly) Σ-bisimilar, then $M \models \alpha[m]$ iff $N \models \alpha[n]$.

Definition 33 (Simple expansion, cf. [Doe96, 3.2])**.** Let $\mathcal{L}$ be a first-order signature and M an $\mathcal{L}$-structure. Further, let n be a natural number, $a_1, \ldots, a_n \in |M|$ and $\mathcal{L}^+ = \mathcal{L} \uplus \{c_1, \ldots, c_n\}$, where the c_i are constant symbols. Then the *simple expansion* $(M, a_1, \ldots, a_n)$ of M to $\mathcal{L}^+$ is the $\mathcal{L}^+$-structure with domain $|M|$ interpreting all elements of $\mathcal{L}$ just as M does, and c_i as a_i for all i.

Definition 34. Let M be an $\mathcal{L}$-model and $\Sigma(x)$ a set of $\mathcal{L}$-formulas with at most one free variable x. M is said to *realise* $\Sigma(x)$ iff there is $a \in |M|$ such that for all $\phi \in \Sigma(x)$ it holds $M \models \phi[a]$. M is said to *finitely realise* $\Sigma(x)$ iff for every finite subset $X \subseteq \Sigma(x)$ there is $a \in |M|$ such that for all $\phi \in X$ it holds $M \models \phi[a]$. Equivalently, we might say that $\Sigma(x)$ *is (finitely) realised* in M.

Definition 35 (ω-saturated, adapted from [BdRV02, Def. 2.63])**.** A first-order structure M over the signature $\mathcal{L}$ is ω-saturated if for every finite subset $\{b_1, \ldots, b_n\} \subseteq |M|$, the simple expansion $M^+ = (M, b_1, \cdots, b_n)$ to $\mathcal{L}^+ = \mathcal{L} \uplus \{c_1, \ldots, c_n\}$ realises every set $\Sigma(x)$ of formulas in $\mathcal{L}^+$ (with only x occurring free) that is finitely realised in M^+.

The following is the main theorem complementing the ring proof. It corresponds to the "backwards direction" of van Benthem's theorem (Theorem 3).

Theorem 14. Let $\alpha(x)$ be a first-order formula in $\mathcal{L}$ with at most one free variable x. If $\alpha(x)$ is invariant for strong Σ-bisimulation, then α is equivalent to the Standard Translation of a sentence in $\mathrm{ML} + \Sigma$.

Proof. The general idea and outer structure of the proof comes from the proof of van Benthem's modal version of the theorem as presented in [BdRV02, Theorem 2.68]. We assume that $\alpha(x)$ is invariant for Σ-bisimulation and look at the set of its Σ-consequences:

$$C_\Sigma(\alpha) = \{\mathrm{ST}_x(\phi) \mid \phi \text{ is a sentence in } \mathrm{ML} + \Sigma \text{ and } \alpha(x) \models \mathrm{ST}_x(\phi)\}$$

We first prove that if $C_\Sigma(\alpha) \models \alpha(x)$, then $\alpha(x)$ is equivalent to the Standard Translation of a sentence in $\mathrm{ML} + \Sigma$. Assume $C_\Sigma(\alpha) \models \alpha(x)$. We show that for some finite subset $X \subseteq C_\Sigma(\alpha)$ it holds $X \models \alpha(x)$. Suppose for a contradiction that for every finite subset X of $C_\Sigma(\alpha)$, $X \cup \{\neg\alpha\}$ has a model. Then, by compactness of first-order logic, $C_\Sigma(\alpha) \cup \{\neg\alpha(x)\}$ has a model. But this is a contradiction to $C_\Sigma(\alpha) \models \alpha(x)$. So for some finite subset $X \subseteq C_\Sigma(\alpha)$ we have $X \models \alpha(x)$. This means that $\models \bigwedge X \to \alpha(x)$. With $\models \alpha(x) \to \bigwedge X$, we get $\models \alpha(x) \leftrightarrow \bigwedge X$. As every element in X is the Standard Translation of a Σ-sentence, so is $\bigwedge X$. This proves the claimed implication.

Using this result, it is enough to show that $C_\Sigma(\alpha) \models \alpha(x)$, which we do by assuming that $M \models C_\Sigma(\alpha)[w]$ and showing $M \models \alpha(x)[w]$. Let

$$T(x) := \{\mathrm{ST}_x(\phi) \mid \phi \text{ is a sentence in } \mathrm{ML} + \Sigma \text{ and } M \models \mathrm{ST}_x(\phi)[w]\}$$

We show that $T(x) \cup \{\alpha\}$ is consistent. To find a contradiction, assume that it is inconsistent. In this case, compactness of first-order logic requires that some finite subset of $T(x) \cup \{\alpha\}$ is also inconsistent. As $T(x)$ is consistent, this subset must include $\alpha(x)$, so denote it by $T_0(x) \cup \{\alpha\}$ with $T_0(x) \subseteq T(x)$. Its inconsistency implies $\models \alpha(x) \to \neg \bigwedge T_0(x)$ and thus $\neg \bigwedge T_0(x) \in C_\Sigma(\alpha)$. It follows $M \models \neg \bigwedge T_0(x)[w]$, but that contradicts $T_0(x) \subseteq T(x)$ and $M \models T(x)[w]$.

So there are a first-order model N and a domain element v of N such that $N \models T(x) \cup \{\alpha(x)\}[v]$. We observe that w and v fulfil the same Σ-sentences: With $M, w \models \phi$ we have $\mathrm{ST}_x(\phi) \in T(x)$ and thus $N, v \models \phi$. Conversely, $M, w \not\models \phi$ implies $M, w \models \neg\phi$, therefore $N, v \models \neg\phi$ and $N, v \not\models \phi$.

We now show that $M \equiv N$. Suppose for contradiction that they are not elementarily equivalent, so w. l. o. g. there is a first-order sentence β such that $M \models \beta$, but $N \not\models \beta$. By Corollary 1, this implies the existence of a sentence $\psi \in \mathrm{ML} + \Sigma$ such that for all $s \in |M|$ and for all $t \in |N|$ it holds $M, s \models \psi$ but $N, t \not\models \psi$. In particular it holds $M, w \models \psi$, but $N, v \not\models \psi$. This is a contradiction to the Σ-equivalence of (M, w) and (N, v). Hence $M \equiv N$.

As M and N are elementarily equivalent, by [CK90, Proposition 3.1.4] there is a model B such that both M and N are elementarily embedded in B, i. e. there are isomorphisms f_M and f_N from M and N onto elementary submodels B_M and B_N of B, respectively. Furthermore, by [Hod97, Corollary 8.2.2], there is an ω^+-saturated and thus ω-saturated elementary extension B^* of B. By [CK90, Proposition 3.1.1. (iii)], B_M and B_N are also elementary submodels of B^* and so both M and N are also elementarily embedded in B^*. Looking at the pointed models, this means that there are $w^*, v^* \in |B^*|$ such that (M, w) is elementarily embedded in (B^*, w^*) and (N, v) is elementarily embedded in (B^*, v^*), where w^* and v^* are just those elements of $|B^*|$ assigned to w and v by f_M and f_N, respectively.

As (M, w) and (N, v) fulfil the same Σ-sentences and elementary embeddings preserve all first-order formulas both ways, (B^*, w^*) and (B^*, v^*) are also Σ-equivalent. As $\mathrm{ML} \subseteq \mathrm{ML} + \Sigma$, they are also modally equivalent. Moreover, B^* is ω-saturated, and so we conclude by [BdRV02, Theorem 2.65] that (B^*, w^*) and (B^*, v^*) are bisimilar[1]. Obviously B^* is isomorphic to itself, so (B^*, w^*) and (B^*, v^*) are strongly Σ-bisimilar. Overall we have established:

$$\begin{aligned} N \models \alpha(x)[v] \quad &\text{iff} \quad B^* \models \alpha(x)[v^*] && \text{(elementary embedding preserves first-order formulas)} \\ &\text{iff} \quad B^* \models \alpha(x)[w^*] && \text{(by strong } \Sigma\text{-bisimilarity and the premise of the theorem)} \\ &\text{iff} \quad M \models \alpha(x)[w] && \text{(elementary embedding preserves first-order formulas)} \end{aligned}$$

Since $N \models \alpha(x)[v]$, we therefore have $M \models \alpha(x)[w]$. This concludes the theorem. □

4.4 Characterising $\mathrm{ML} + \Sigma$

In this section, all of the results of this chapter come together. We prove the equivalence and validity of the three characterisations for $\mathrm{ML} + \Sigma$ by means of a ring proof using the previous theorems.

Theorem 15. Let $\alpha(x)$ be a first-order formula over $\mathcal{L}$ with at most one free variable x. Then the following are equivalent:

(1) α is equivalent to the Standard Translation of a sentence in $\mathrm{ML} + \Sigma$.

(2) α is invariant for Σ-bisimulation.

(3) For any two pointed Kripke models (M, m) and (N, n) it holds: If Duplicator has a winning strategy in the infinite Σ-game starting with the white pebbles at m and n, then $M \models \alpha[m]$ iff $N \models \alpha[n]$.

(4) α is invariant for strong Σ-bisimulation.

Proof. The proof from (1) to (2) follows directly from Theorem 9 using the adequacy of the Standard Translation.

We prove that (2) implies (3). Suppose that (M, m) and (N, n) are two pointed Kripke models such that Duplicator has a winning strategy in the Σ-game starting with the white pebbles at m and n. We need to show that $M \models \alpha[m]$ iff $N \models \alpha[n]$. By Theorem 10, (M, m) and (N, n) are Σ-bisimilar. Assuming (2), we conclude $M \models \alpha[m]$ iff $N \models \alpha[n]$, so point (2) implies point (3).

We prove that (3) implies (4). Suppose that (M, m) and (N, n) are two strongly bisimilar pointed Kripke models. We need to show that $M \models \alpha[m]$ iff $N \models \alpha[n]$. By Theorem 12, Duplicator has a winning strategy in the infinite Σ-game starting from (m, n). Assuming point (3), it then holds $M \models \alpha[m]$ iff $N \models \alpha[n]$, proving the implication (3)→(4).

Finally, the proof from (4) to (1) is given by Theorem 14. □

[1]The quoted theorem is proved for modal logic without nominals, but the proof works in the same way for modal logic with nominals.

Chapter 5

Undecidability of Invariance by Σ-Bisimulation

For a first-order $\mathcal{L}$-formula α, we would like to know if it is equivalent to a sentence in $\mathrm{ML}+\Sigma$. By Theorem 15, this is equivalent to deciding if α is invariant for Σ-bisimulation. In this chapter we show that the latter is undecidable.

Van Benthem proved in [vB96, Remark 4.19] that normal bisimulation-invariance is undecidable for first-order formulas. This proof was adapted for quasi-injective bisimulations by Hodkinson and Tahiri in [HT10, Theorem 4.1] and for proper generated submodel^N isomorphisms by Kouvaros in [Kou11, Theorem 4.0.7]. Our proof for Σ-bisimulation is mostly based on the version in [HT10].

One key idea is to create a disjunction whose two alternatives only apply to a part of the model. In order to enforce this, we use a technique called relativisation:

Definition 36 (Relativisation)**.** Let L be a first-order signature, α an L-formula and P a unary relation symbol not occurring in L. We inductively define the relativisation of α to P (denoted α^P) as follows:

$$\begin{aligned}
\alpha^P &= \alpha \text{ if } \alpha \text{ is atomic}\\
(\alpha_1 \wedge \alpha_2)^P &= \alpha_1^P \wedge \alpha_2^P\\
(\neg\alpha)^P &= \neg(\alpha^P)\\
(\exists x\alpha)^P &= \exists x(P(x) \wedge \alpha^P)
\end{aligned}$$

Lemma 7 (Relativisation lemma)**.** Let L be a first-order signature and P a unary relation symbol in L. Let M be a first-order structure over L with interpretation h, let $P' = h(P)$ and g an assignment such that $g(x) \in P'$ for all variables x. Further let $M{\restriction}P$ be the model obtained by restricting M to the domain P' and let h' be its interpretation. That is, $|M{\restriction}P| = P'$ and all relation symbols are defined by the intersection of their M-correspondents with P' and $P' \times P'$, respectively. Then for all L-formulas α not containing P nor any constant symbols i with $h(i) \notin P'$, it holds

$$M, g \models \alpha^P \quad \text{iff} \quad M{\restriction}P, g \models \alpha$$

Proof. Via induction on the structure of α. In the base case, α is atomic, so let $\alpha = Q(x_1, \ldots, x_n)$ where Q is equality or any of the relation symbols in $L \setminus \{P\}$, and $x_1, \ldots, x_n$ variables or constant symbols. For any interpretation h let $h(=)$ denote equality. Then

$$\begin{aligned}
&M, g \models \alpha^P\\
\text{iff}\quad &M, g \models Q(x_1, \ldots, x_n)\\
\text{iff}\quad &h(Q)([h,g](x_1), \ldots, [h,g](x_n))\\
\text{iff}\quad &h'(Q)([h',g](x_1), \ldots, [h',g](x_n)) &&\text{(as } [h,g](x_i) \in |M{\restriction}P| \text{ for all } i)\\
\text{iff}\quad &M{\restriction}P, g \models Q(x_1, \ldots, x_n)
\end{aligned}$$

If $\alpha = \neg\beta$ or $\alpha = \beta_1 \wedge \beta_2$, we conclude as follows, where in both cases the step from (3) to (4) follows from the inductive hypothesis:

$$\begin{array}{llr}
 & M, g \models (\neg\beta)^P & (1) \\
\text{iff} & M, g \models \neg(\beta^P) & (2) \\
\text{iff} & M, g \not\models (\beta)^P & (3) \\
\text{iff} & M{\restriction}P, g \not\models \beta & (4) \\
\text{iff} & M{\restriction}P, g \models \neg\beta & (5)
\end{array}
\qquad
\begin{array}{ll}
 & M, g \models (\beta_1 \wedge \beta_2)^P \\
\text{iff} & M, g \models \beta_1^P \wedge \beta_2^P \\
\text{iff} & M, g \models \beta_1^P \text{ and } M, g \models \beta_2^P \\
\text{iff} & M{\restriction}P, g \models \beta_1 \text{ and } M{\restriction}P, g \models \beta_2 \\
\text{iff} & M{\restriction}P, g \models \beta_1 \wedge \beta_2
\end{array}$$

The most interesting case follows. Let $\alpha = \exists x\beta$.

$$\begin{array}{ll}
 & M, g \models (\exists x\beta)^P \\
\text{iff} & M, g \models \exists x(P(x) \wedge \beta^P) \\
\text{iff} & \exists a \in |M|, g' \overset{x}{\sim} g\text{: } g'(x) = a \text{ and } M, g' \models P(x) \wedge \beta^P \\
\text{iff} & \exists a \in |M|, g' \overset{x}{\sim} g\text{: } g'(x) = a, a \in P' \text{ and } M, g' \models \beta^P \\
\text{iff} & \exists a \in |M|, g' \overset{x}{\sim} g\text{: } g'(x) = a, a \in P' \text{ and } M{\restriction}P, g' \models \beta \\
 & \hfill \text{(by inductive hypothesis, as } g'(x) \in P'\text{)} \\
\text{iff} & \exists a \in |M{\restriction}P|, g' \overset{x}{\sim} g\text{: } g'(x) = a \text{ and } M{\restriction}P, g' \models \beta \qquad \text{(as } |M{\restriction}P| = |M| \cap P'\text{)} \\
\text{iff} & M{\restriction}P, g \models \exists x\beta
\end{array}$$

□

Remark. If α in Lemma 7 is a sentence, then its value in either model does not depend on g. So then it holds

$$M \models \alpha^P \quad \text{iff} \quad M{\restriction}P \models \alpha$$

Theorem 16. It is not decidable whether a given $\mathcal{L}$-formula is invariant under Σ-bisimulation.

Proof. We first show that $\phi(x) = R(x, x)$ is not invariant under Σ-bisimulation. Let M be a first-order structure with domain $\{w, v\}$ interpreting R as $\{(v, w), (w, w)\}$. The relation $B = \{(w, w), (v, w), (w, v)\}$ is a Σ-bisimulation between (M, w) and (M, v) (for PROP = NOM = $\emptyset$). However, $M \models \phi[w]$ and $M \not\models \phi[v]$, so ϕ is not invariant under Σ-bisimulation.

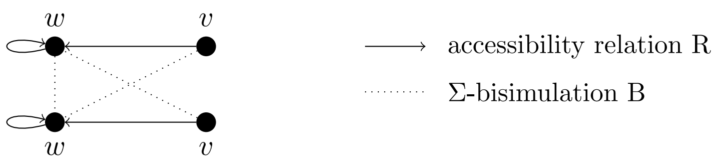

Figure 5.1: Two copies of structure M and the Σ-bisimulation between them.

Now, define $\mathcal{L}' := \mathcal{L} \setminus \text{NOM}$, and let α be a first-order sentence over $\mathcal{L}'$ and P a unary relation symbol not in $\mathcal{L}'$. Define $f(\alpha)$ over $\mathcal{L}' \cup \{P\}$ as

$$f(\alpha) := \phi(x) \vee (\exists x P(x) \rightarrow \alpha^P)$$

We prove that α is valid iff $f(\alpha)$ is invariant under Σ-bisimulation.

- $\rightarrow$: Assume α is valid. We first show that $(\exists x P(x) \rightarrow \alpha^P)$ is valid, i. e. that for any model N with $N \models \exists x P(x)$, it holds $N \models \alpha^P$. As $N \models \exists x P(x)$, $|N{\restriction}P| \neq \emptyset$ and $N{\restriction}P$ is a model. Hence it holds $N{\restriction}P \models \alpha$ and thus, by Lemma 7, $N \models \alpha^P$ as required. So $(\exists x P(x) \rightarrow \alpha^P)$ is valid, and so is $f(\alpha)$. This means that $f(\alpha)$ is equivalent to $\top$ and hence invariant under Σ-bisimulation.

$\leftarrow$: We show the contraposition, so assume α is not valid. That means there is a first-order $\mathcal{L}'$-model N such that $N \not\models \alpha$. Let U be the disjoint union of M and N and let P be a new unary predicate symbol made true in $|N|$.

Remember that $\phi(x) = R(x, x)$. With $M \models \phi[w]$ we get $U \models \phi[w]$ and hence $U \models f(\alpha)[w]$. On the other hand, $U \models \exists x P(x)$ as $|N| \neq \emptyset$. As $N = U{\upharpoonright}P$, we have $U{\upharpoonright}P \not\models \alpha$. We can now apply Lemma 7 and conclude $U \not\models \alpha^P$. Therefore $U \not\models \exists x P(x) \to \alpha^P$. Since furthermore, $M \not\models \phi[v]$ implies $U \not\models \phi[v]$, we get $U \not\models f(\alpha)[v]$. Overall, as (U, w) and (U, v) are Σ-bisimilar, $U \models f(\alpha)[w]$ but $U \not\models f(\alpha)[v]$, we know that $f(\alpha)$ is not invariant under Σ-bisimulation.

In order to conclude the theorem, suppose for a contradiction that it was decidable whether a given $\mathcal{L}$-formula is invariant under Σ-bisimulation. Then, for any first-order sentence α over $\mathcal{L}'$, we could construct $f(\alpha)$ and decide whether it is invariant under Σ-bisimulation. By the equivalence above, this would also decide whether α is valid. However, as $\mathcal{L}'$ contains the binary relation symbol R, the validity of first-order $\mathcal{L}'$-sentences is undecidable [BGG97, Theorem 4.0.1], so this is a contradiction. Hence it is undecidable whether a given $\mathcal{L}$-formula is invariant under Σ-bisimulation. $\square$

Chapter 6

The Modal Core of $\mathbf{ML} + \Sigma$

In this chapter, we will look at the behaviour of Σ-sentences under three operations: forward-generated submodels, disjoint unions and p-morphisms. Modal formulas are all invariant under these operations, but for first-order formulas this does not hold in general. Looking at the results of Theorem 7, we suspect that Σ-sentences that are invariant under such an operation are equivalent to a boolean combination of first-order sentences that are invariant under the operation and modal formulas. However, the actual results were much more interesting.

Definition 37 (P-morphism)**.** Let M and N be Kripke models with $M = (W_M, R_M, h_M)$ and $N = (W_N, R_N, h_N)$. A function f: $W_M \to W_N$ is called *p-morphism* from M to N iff

1. For all $w \in W_M$ and $p \in \text{PROP}$ it holds $w \in h_M(p)$ iff $f(w) \in h_N(p)$.
2. For all $w \in W_M$ and $i \in \text{NOM}$ it holds $w = h_M(i)$ iff $f(w) = h_N(i)$.
3. *forth*: For all $w, w' \in W_M$ with $R_M(w, w')$ it holds $R_N(f(w), f(w'))$.
4. *back*: For all $w \in W_M$ and $v \in W_N$ with $R_N(f(w), v)$ there is a world $w' \in W_M$ such that $f(w') = v$ and $R_M(w, w')$.

Definition 38. Let $\alpha(x)$ be a first-order formula with at most one free variable x. We say that α is invariant under p-morphisms iff for all pointed Kripke models (M, m) and (N, n) over the same signature and all p-morphisms f from M to N such that $f(m) = n$, it holds $M \models \alpha[m]$ iff $N \models \alpha[n]$.

Lemma 8 (Adapted from [Hod11c, Exercise 4.10])**.** Let $N = (W, R, h)$, $N' = (W', R', h')$ be Kripke models and $t \in W, t' \in W'$ be worlds. Then (N, t) and (N', t') are bisimilar iff there is a model M, a world u of its frame and p-morphisms $f : M \to N, f' : M \to N'$ with $f(u) = t$ and $f'(u) = t'$.

Proof. First assume there is such a model $M = (W_M, R_M, h_M)$, a world $u \in W_M$ and p-morphisms f and f'. Let $B = \{(f(m), f'(m)) \mid m \in W_M\}$. B is a bisimulation between (N, t) and (N', t'):

- $(t, t') = (f(u), f'(u)) \in B$ as $u \in W_M$
- For $p \in \text{PROP}$ and $(f(m), f'(m)) \in B$ it holds $f(m) \in h(p)$ iff $m \in h_M(p)$ iff $f'(m) \in h'(p)$.
- For $i \in \text{NOM}$ and $(f(m), f'(m)) \in B$ it holds $f(m) = h(i)$ iff $m = h_M(i)$ iff $f'(m) = h'(i)$.
- If $(f(m), f'(m)) \in B$ and $(f(m), v) \in R$ for some $v \in W_M$, then – by the (back) property of p-morphism f – there is a world $m' \in W_M$ with $f(m') = v$ and $R_M(m, m')$. With $m' \in W_M$ we know $(f(m'), f'(m')) \in B$ and – by the (forth) property of p-morphism f' – $R'(f'(m), f'(m'))$. The proof of the other direction is analogue.

For the converse assume that B is a bisimulation between (N, t) and (N', t'). We construct a model $M = (W_M, R_M, h_M)$ with p-morphisms to N and N' as described by defining:

- $W_M = B$

- $R_M((w,w'),(v,v'))$ iff $R(w,v)$ and $R'(w',v')$
- $(w,w') \in h_M(p)$ iff $w \in h(p)$ for all $p \in \mathrm{PROP}$
- $(w,w') = h_M(i)$ iff $w = h(i)$ for all $i \in \mathrm{NOM}$

Further define the p-morphisms $f : M \to N$ as $f(w,w') = w$ and $f' : M \to N'$ as $f'(w,w') = w'$. We show that f is a p-morphism: The preservation of atoms and nominals follows directly from the definitions of h_M and f. The (forth) condition follows directly from the definitions of R_M and f. The (back) condition: Let $(w,w') \in W_M = B$ and $v \in W$ such that $R(f(w,w'),v)$, i. e. $R(w,v)$. As B is a bisimulation, there is $v' \in W'$ such that $(v,v') \in B$ and $R'(w',v')$. But now we have the world $(v,v') \in W_M$ such that $f(v,v') = v$ and $R_M((w,w'),(v,v'))$.

Note that because B is a bisimulation, the last two definitions above are equivalent to $(w,w') \in h_M(p)$ iff $w' \in h'(p)$ for all $p \in \mathrm{PROP}$ and $(w,w') = h_M(i)$ iff $w' = h'(i)$ for all $i \in \mathrm{NOM}$. These are just the symmetric definitions for f' and thus the proof that f' is a p-morphism is the same as for f.

To conclude the proof, we define $u := (t,t')$, so it holds $f(u) = t$ and $f'(u) = t'$. $\square$

For the rest of this chapter we only look at Σ-formulas over signatures with $\mathrm{NOM} = \emptyset$. This implies that the first-order formulas do not contain any constant symbols.

Definition 39 (Disjoint union)**.** Let $M = (W_M, R_M, h_M)$ and $N = (W_N, R_N, h_N)$ be Kripke models over the same signature such that $W_M \cap W_N = \emptyset$ ("disjoint Kripke models"). The model $M \uplus N = (W_M \cup W_N, R_M \cup R_N, h)$, where $h(p) = h_M(p) \cup h_N(p)$ for all propositional symbols p, is called the *disjoint union* of M and N. If M and N are not disjoint, their disjoint union is defined as the disjoint union of disjoint isomorphic copies of M and N.

Definition 40. Let $\alpha(x)$ be a first-order formula with at most one free variable x. We say that α is invariant under disjoint unions iff for all disjoint Kripke models M and N over the same signature and all $w \in |M|$ it holds $M \models \alpha[w]$ iff $(M \uplus N) \models \alpha[w]$.

Definition 41 (Forward-generated submodel)**.** Let $M = (W, R, h)$ be a Kripke model and $m \in W$. The *forward-generated* (short *f-generated* or *generated*) *submodel* of (M,m) is the smallest Kripke model $N = (W', R', h')$ such that

- $m \in W'$
- $W' \subseteq W$
- If $w \in W', v \in W$ and $R(w,v)$, then $v \in W'$.
- $R' = R \cap (W' \times W')$
- $h'(p) = h(p) \cap W'$ for all $p \in \mathrm{PROP}$

Definition 42. Let $\alpha(x)$ be a first-order formula with at most one free variable x. We say that α is invariant under forward-generated submodels iff for each pointed Kripke model (M,w) with forward-generated submodel (N,w) it holds $M \models \alpha[w]$ iff $N \models \alpha[w]$.

Theorem 17. Let $\alpha(x)$ be the Standard Translation of a Σ-sentence without nominals. Then the following are equivalent:

(1) α is equivalent to the Standard Translation of a modal formula.

(2) α is invariant under bisimulation.

(3) α is invariant under forward-generated submodels.

(4) α is invariant under disjoint unions.

(5) α is invariant under p-morphisms.

Proof. The bisimulation characterisation of modal logic (Theorem 3) directly entails the back and forth between (1) and (2). The direction from (1) to (3), (4) and (5) follows from the adequacy of the Standard Translation and the fact that modal formulas are invariant under f-generated submodels [BdRV02, Proposition 2.6], disjoint unions [BdRV02, Proposition 2.3] and p-morphisms [BdRV02, Proposition 2.14].

We now show that if either of (3), (4) or (5) holds, so does (1). The beginning of the proof is very similar to the proof of Theorem 14. We assume that $\alpha(x)$ is invariant under one of the transformations listed above and look at the set of its modal consequences:

$$C_{\mathrm{mod}}(\alpha) = \{\mathrm{ST}_x(\phi) \,|\, \phi \text{ is a modal formula and } \alpha(x) \models \mathrm{ST}_x(\phi)\}$$

We first prove that if $C_{\mathrm{mod}}(\alpha) \models \alpha(x)$, then $\alpha(x)$ is equivalent to the Standard Translation of a modal formula. Assume $C_{\mathrm{mod}}(\alpha) \models \alpha(x)$. We show that for some finite subset $X \subseteq C_{\mathrm{mod}}(\alpha)$ it holds $X \models \alpha(x)$. Suppose for a contradiction that for every finite subset X of $C_{\mathrm{mod}}(\alpha)$, $X \cup \{\neg\alpha\}$ has a model. Then, by compactness of first-order logic, $C_{\mathrm{mod}}(\alpha) \cup \{\neg\alpha(x)\}$ has a model. But this is a contradiction to $C_{\mathrm{mod}}(\alpha) \models \alpha(x)$. So for some finite subset $X \subseteq C_{\mathrm{mod}}(\alpha)$ we have $X \models \alpha(x)$. This means that $\models \bigwedge X \to \alpha(x)$. With $\models \alpha(x) \to \bigwedge X$, we get $\models \alpha(x) \leftrightarrow \bigwedge X$. As every element in X is the Standard Translation of a modal formula, so is $\bigwedge X$. This proves the claimed implication.

Using this result, it is enough to show that $C_{\mathrm{mod}}(\alpha) \models \alpha(x)$, which we do by assuming that $M \models C_{\mathrm{mod}}(\alpha)[w]$ and showing $M \models \alpha(x)[w]$. Let

$$T(x) := \{\mathrm{ST}_x(\phi) \,|\, \phi \text{ is a modal formula and } M \models \mathrm{ST}_x(\phi)[w]\}$$

We show that $T(x) \cup \{\alpha\}$ is consistent. To find a contradiction, assume that it is inconsistent. In this case, compactness of first-order logic requires that some finite subset of $T(x) \cup \{\alpha\}$ is also inconsistent. As $T(x)$ is consistent, this subset must include $\alpha(x)$, so denote it by $T_0(x) \cup \{\alpha\}$ with $T_0(x) \subseteq T(x)$. Its inconsistency implies $\models \alpha(x) \to \neg \bigwedge T_0(x)$ and thus $\neg \bigwedge T_0(x) \in C_{\mathrm{mod}}(\alpha)$. It follows $M \models \neg \bigwedge T_0(x)[w]$, but that contradicts $T_0(x) \subseteq T(x)$ and $M \models T(x)[w]$.

So there are a first-order model N and a domain element v of N such that $N \models T(x) \cup \{\alpha(x)\}[v]$. We observe that (M, w) and (N, v) are modally equivalent: With $M, w \models \phi$ we have $\mathrm{ST}_x(\phi) \in T(x)$ and thus $N, v \models \phi$. Conversely, $M, w \not\models \phi$ implies $M, w \models \neg\phi$, therefore $N, v \models \neg\phi$ and $N, v \not\models \phi$.

Now, we need to distinguish our three cases, depending on the transformation α is invariant under. The models involved are illustrated in Figure 6.1 to Figure 6.3.

(3) α is invariant under forward-generated submodels.
Let (M', w) and (N', v) be the f-generated submodels of (M, w) and (N, v). Let B be the disjoint union of M' and N'. Then (M', w) is also the forward-generated submodel of (B, w) and (N', v) is the forward-generated submodel of (B, v). By [Hod97, Corollary 8.2.2], there is an ω^+-saturated and thus ω-saturated elementary extension B^* of B.

Every modal formula is invariant under f-generated submodels [BdRV02, Proposition 2.6]. As (M, w) and (N, v) are modally equivalent, so are (M', w) and (N', v), and subsequently (B, w) and (B, v). Elementary extensions preserve all first-order formulas in both ways, in particular the Standard Translations of all modal formulas. Hence (B^*, w) and (B^*, v) are also modally equivalent and – by ω-saturation – bisimilar [BdRV02, Theorem 2.65]. Further $B^* \equiv B^*$, so (B^*, w) and (B^*, v) are Σ-bisimilar.

Now we can conclude that α holds subsequently in the models (N, v) (by definition of N, v), (N', v) (α invariant under f-generated submodels), (B, v) (α invariant under f-generated submodels), (B^*, v) (elementary extension preserves all first-order formulas), (B^*, w) (by Σ-bisimilarity), (B, w) (elementary extension), (M', w) (f-generated submodel) and finally (M, w) (f-generated submodel).

(4) α is invariant under disjoint unions.
Let B be the disjoint union of M and N. Again, by [Hod97, Corollary 8.2.2], there is an ω-saturated elementary extension B^* of B. As every modal formula is invariant under disjoint unions [BdRV02, Proposition 2.3] and (M, w) and (N, v) are modally equivalent, so are (B, w)

and (B, v). As above we conclude that (B^*, w) and (B^*, v) are also modally equivalent and by ω-saturation bisimilar. With $B^* \equiv B^*$, we know that (B^*, w) and (B^*, v) are Σ-bisimilar.

Now α holds subsequently in (N, v) (by definition of N and v), (B, v) (α invariant under disjoint unions), (B^*, v) (elementary extension preserves all first-order formulas), (B^*, w) (by Σ-bisimilarity), (B, w) (elementary extension) and thus (M, w) (disjoint unions).

(5) α is invariant under p-morphisms.
By [Hod97, Corollary 8.2.2], there are ω-saturated models M' and N' elementarily extending M and N, respectively. As all first-order formulas are invariant under elementary extensions and (M, w) and (N, v) are modally equivalent, (M', w) and (N', v) are also modally equivalent and by ω-saturation bisimilar. Then, by Lemma 8, there is a model (B, u) with p-morphisms from u to w and from u to v.

Now α holds subsequently in (N, v) (by definition of N, v), (N', v) (elementary extension preserves all first-order formulas), (B, u) (α invariant under p-morphisms), (M', w) (α invariant under p-morphisms) and finally (M, w) (elementary extension).

In each case we have shown that $M, w \models \alpha$ and therefore $C_{\text{mod}}(\alpha) \models \alpha(x)$. This concludes the theorem. □

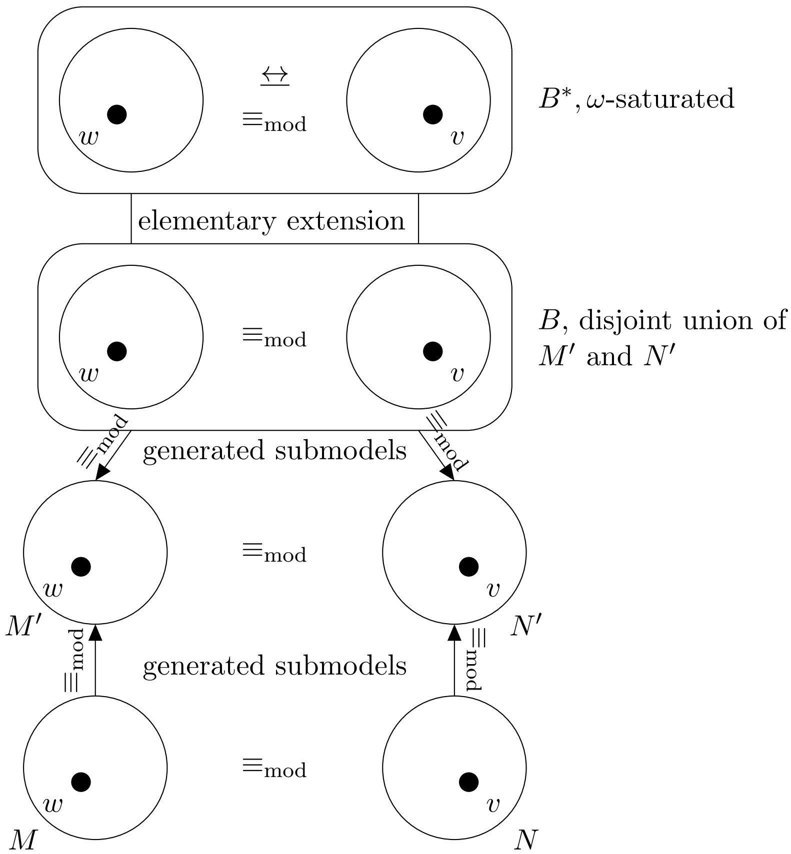

Figure 6.1: Illustration of the models involved for the forward-generated submodel case of the proof of Theorem 17.

Given that we were surprised by these results, we will try to provide an intuitive explanation after the fact: If a Σ-sentence was *not* equivalent to a modal formula, it would contain an essential "first-order-sentence part". But first-order sentences have all their variables bound and thus have a global perspective.

As the theorem uses invariance, our operations can be looked at in both ways (e. g. either a model that *is* a generated submodel or a model that *has* a generated submodel). In one way or another, all of of the operations can be used to extend a model with a new connection component

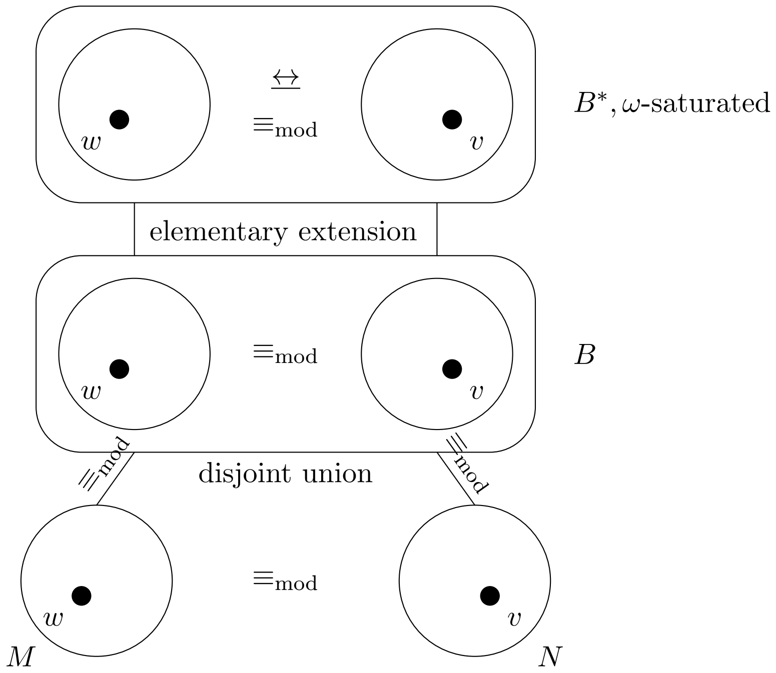

Figure 6.2: Illustration of the models involved for the disjoint union case of the proof of Theorem 17.

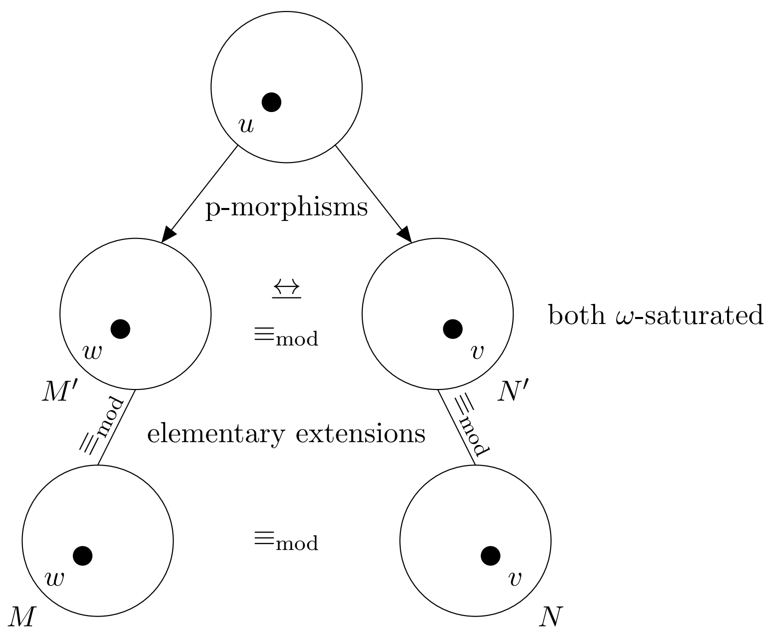

Figure 6.3: Illustration of the models involved for the p-morphism case of the proof of Theorem 17.

or to remove a distant connection component. This is very natural for the disjoint union, and both p-morphisms and f-generated submodels have basically “local semantics”.

If the first binder in our first-order part is $\exists$, we can take a model fulfilling the Σ-sentence which fulfills the first-order part only in a distant part of the model. Our operations can then remove that distant part from the model and the resulting model will violate the sentence. So the Σ-sentence is not invariant under the operation. If the first binder in our first-order part is $\forall$, we can extend a model fulfilling the Σ-sentence with a new connection component that violates the first-order statement and hence also the Σ-sentence. So the sentence is not invariant under the operation.

Chapter 7

Conclusions

7.1 Summary of Results

This thesis investigated the hybrid logic $\mathrm{ML}+\Sigma$. We proved that any formula $E\phi \in \mathrm{ML}+E+\downarrow$ is equivalent to a formula in $\mathrm{ML}+\Sigma$. Conversely, we showed that any sentence in $\mathrm{ML}+\Sigma$ is equivalent to a boolean combination of modal formulas and formulas of the form $E\phi$ with $\phi \in \mathrm{ML}+E+\downarrow$.

As the Standard Translations of formulas of the form $E\phi$ ($\phi \in \mathrm{ML}+E+\downarrow$) are first-order sentences, this means that all sentences of $\mathrm{ML}+\Sigma$ are essentially equivalent to a boolean combination of modal formulas and first-order sentences. Moreover, for every first-order sentence of the correspondence language there is an equivalent sentence in $\mathrm{ML}+\Sigma$. Hence $\mathrm{ML}+\Sigma$ is undecidable.

We then derived three different characterisations for the hybrid language with the Σ binder, two of them using a form of bisimulation and one using a game. We defined Σ-*bisimilar* to mean "bisimilar and elementarily equivalent" and *strongly* Σ-*bisimilar* as "bisimilar and isomorphic". The Σ-game used two different kinds of moves, one representing $\Diamond$ and one representing Σ. It is a hybrid game using aspects from the modal bisimulation game and the Ehrenfeucht-Fraïssé game for first-order logic. We proved that if $\alpha(x)$ is a first-order $\mathcal{L}$-formula, then the following are equivalent:

- α is equivalent to the Standard Translation of a sentence in $\mathrm{ML}+\Sigma$.
- α is invariant for Σ-bisimulation.
- α is invariant for infinite Σ-games in which Duplicator has a winning strategy.
- α is invariant for strong Σ-bisimulation.

Given our characterisation of $\mathrm{ML}+\Sigma$, we would like to use it to decide if a first-order formula is equivalent to a sentence in $\mathrm{ML}+\Sigma$. However, we showed that this is not possible, because it is undecidable whether a first-order formula is invariant under Σ-bisimulation.

Finally, we found a modal core of the Σ-language: We proved that if a Σ-formula without nominals is invariant under forward-generated submodels, disjoint unions *or* p-morphisms, then it is equivalent to a modal formula.

7.2 Further Research

We propose the following directions for further research.

In Chapter 6, we proved that invariance under forward-generated submodels, disjoint unions or p-morphisms are sufficient criteria for a Σ-sentence (without nominals) to be equivalent to a modal formula. There might be more operations such that Σ-formulas that are invariant under those operations are equivalent to modal formulas.

Interpolation is a property possessed by both modal logic (without nominals) and first-order logic. It reads: "Let ϕ and ψ be modal formulas (first-order sentences) such that $\phi \to \psi$ is valid. Then there is a modal formula (first-order sentence) χ using only those propositional symbols (relation symbols) that occur in both ϕ and ψ such that $\phi \to \chi$ and $\chi \to \psi$ are valid." As $\mathrm{ML} + \Sigma$ is essentially a combination of modal logic and first-order sentences, we suspect that it also has the interpolation property. It would be interesting to see if it really does.

The logics we were concerned with in this work are defined very concretely: We know how they are syntactically constructed and what their exact semantics are. In contrast, abstract model theory defines an abstract logic in terms of some essential properties it has to fulfil. Lindström's theorem is one of the best known results in abstract model theory. It states that an abstract logic that is countably compact and fulfils the downward Löwenheim-Skolem property is equivalent to first-order logic [CK90, Theorem 2.5.4]. There are also Lindström theorems for modal logic, differing in the properties they require [BdRV02, Ben07]. One of these properties is bisimulation-invariance, so the characterising property for the logic as a fragment of first-order logic is also used in the context of abstract logics. Kouvaros proved a Lindström theorem for hybrid logic with the $\forall$ binder [Kou11, Section 6]. By finding a Lindström theorem for $\mathrm{ML} + \Sigma$ (possibly using Σ-bisimilarity), we could generalise the results we presented here.

Bibliography

[ABM01] Carlos Areces, Patrick Blackburn, and Maarten Marx. Hybrid logics: Characterization, interpolation and complexity. *Journal of Symbolic Logic*, 66(3):977–1010, 2001.

[AQM+97] Serge Abiteboul, Dallan Quass, Jason Mchugh, Jennifer Widom, and Janet Wiener. The lorel query language for semistructured data. *International Journal on Digital Libraries*, 1:68–88, 1997.

[BdRV02] Patrick Blackburn, Maarten de Rijke, and Yde Venema. *Modal Logic*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 2002.

[Ben07] Johan Benthem. A new modal lindström theorem. *Logica Universalis*, 1:125–138, 2007.

[BGG97] Egon Börger, Erich Grädel, and Yuri Gurevich. *The classical decision problem*. Perspectives in mathematical logic. Springer, 1997.

[BS95] Patrick Blackburn and Jerry Seligman. Hybrid languages. *Journal of Logic, Language and Information*, 4:251–272, 1995.

[BS98] Patrick Blackburn and Jerry Seligman. What are hybrid languages? In M. Kracht, M. de Rijke, H. Wansing, and M. Zakharyaschev, editors, *Advances in Modal Logic*, volume 1, pages 41–62. CSLI Publications, Stanford University, 1998.

[BvBW07] Patrick Blackburn, Johan van Benthem, and Frank Wolter, editors. *Handbook of Modal Logic*. Studies in Logic and Practical Reasoning. Elsevier, 2007.

[CK90] C.C. Chang and H.J. Keisler. *Model Theory*. Studies in Logic and the Foundations of Mathematics. North-Holland, 1990.

[Doe96] Kees Doets. *Basic Model Theory*. Studies in Logic, Language, and Information. CLSI Publications, 1996.

[FdR06] Massimo Franceschet and Maarten de Rijke. Model checking hybrid logics (with an application to semistructured data). *Journal of Applied Logic*, 4(3):279–304, 2006.

[Hod97] Wilfrid Hodges. *A Shorter Model Theory*. Cambridge University Press, 1997.

[Hod11a] Ian Hodkinson. 140 Logic. Lecture slide deck, `http://www.doc.ic.ac.uk/~imh/teaching/140_logic/140.pdf`, October 2011.

[Hod11b] Ian Hodkinson. 499 Modal and temporal logic. Lecture slide deck, `http://www.doc.ic.ac.uk/~imh/teaching/499_mtl/499.html`, October 2011.

[Hod11c] Ian Hodkinson. MTL solutions. Solutions to exercises accompanying lecture 499 – Modal and temporal logic, `http://www.doc.ic.ac.uk/~imh/teaching/499_mtl/499.html`, October 2011.

[HS08] Christian Hoareau and Ichiro Satoh. Hybrid logics and model checking: A recipe for query processing in location-aware environments. In *22nd International Conference on Advanced Information Networking and Applications*, pages 130–137, March 2008.

[HT10] Ian Hodkinson and Hicham Tahiri. A bisimulation characterization theorem for hybrid logic with the current-state binder. *The Review of Symbolic Logic*, 3(2):247–261, June 2010.

[Kou11] Panagiotis Kouvaros. A characterization theorem for hybrid logic with the for-all binder. Master's thesis, Imperial College London, September 2011.

[SK12] Anders Søgaard and Søren Lind Kristiansen. Using hybrid logic for querying dependency treebanks. *Linguistic Issues in Language Technology*, 7(5), January 2012.

[tC05] Balder David ten Cate. *Model theory for extended modal languages.* PhD thesis, Institute for Logic, Language and Computation, University of Amsterdam, 2005.

[Tza99a] Miroslava Tzakova. *Hybrid Languages.* PhD thesis, Universität des Saarlandes, Saarbrücken, 1999.

[Tza99b] Miroslava Tzakova. Tableau calculi for hybrid logics. In *Proceedings of the International Conference on Automated Reasoning with Analytic Tableaux and Related Methods*, TABLEAUX '99, pages 278–292, London, UK, 1999. Springer-Verlag.

[vB76] Johan van Benthem. *Modal Correspondence Theory.* PhD thesis, Mathematisch Instituut & Instituut voor Grondslagenonderzoek, University of Amsterdam, 1976.

[vB85] Johan van Benthem. *Modal Logic and Classical Logic.* Bibliopolis, Napoli, Italy, 1985.

[vB96] Johan van Benthem. *Exploring Logical Dynamics.* Studies in Logic, Language and Information. CSLI Publications, 1996.

[vB10] Johan van Benthem. *Modal Logic for Open Minds.* CSLI lecture notes. Center for the Study of Language and Information, 2010.