Expressiveness modulo bisimilarity: a coalgebraic perspective

Yde Venema

Abstract One of van Benthem's seminal results is the Bisimulation Theorem characterizing modal logic as the bisimulation-invariant fragment of first-order logic. Janin and Walukiewicz extended this theorem to include fixpoint operators, showing that the modal μ-calculus μML is the bisimulation-invariant fragment of monadic second-order logic MSO. Their proof uses parity automata that operate on Kripke models, and feature a transition map defined in terms of certain fragments of monadic first-order logic. In this paper we decompose their proof in three parts: (1) two automata-theoretic characterizations, of MSO and μML respectively, (2) a simple model-theoretic characterization of the identity-free fragment of monadic first-order logic, and (3) an automata-theoretic result, stating that (a strong version of) the second result somehow propagates to the level of full fixpoint logics.
Our main contribution shows that the third result is an instance of a more general phenomenon that is essentially coalgebraic in nature. We prove that if one set Λ of predicate liftings (or modalities) for a certain set functor T uniformly corresponds to the T-natural fragment of another such set Λ', then the fixpoint logic associated with Λ is the bisimulation-invariant logic of the fixpoint logic associated with Λ'.

Keywords modal fixpoint logic, bisimulation invariance, coalgebra, coalgebraic modal logic, predicate liftings, monadic second-order logic.

1 Introduction

Johan van Benthem is one of the founders of *correspondence theory* [3] as a branch of modal logic where the expressiveness of modal logic as a language for describing

Yde Venema
Institute for Logic, Language and Computation, Universiteit van Amsterdam, Science Park 107, 1098 XG Amsterdam, The Netherlands
e-mail: Y.Venema@uva.nl

Kripke structures is compared to that of more classical languages such as first-order logic. Perhaps his most important contribution to this area is the Bisimulation Theorem stating that modal logic is the bisimulation-invariant fragment of first-order logic [2], in a slogan:

$$ML = FO/\leftrightarroweq. \tag{1}$$

More precisely, van Benthem showed that a formula $\varphi(x)$, in the language of first-order logic for Kripke models, is invariant under bisimulations iff it is equivalent to (the standard translation of) a modal formula. This observation fits the model-theoretic tradition of *preservation results*, characterizing fragments of (first-order) logic through a certain semantic property. What makes the result important is that in many applications of modal logic, it is natural to identify bisimilar states, and so properties that are not bisimulation invariant are irrelevant. From this perspective, the Bisimulation Theorem states an *expressive completeness* result: when it comes to relevant properties, modal logic has the same expressive power as first-order logic.

Van Benthem's result has inspired many modal logicians, and over the years a wealth of variants of the Bisimulation Theorem have been obtained. Roughly, these can be classified as follows:

- Results showing that van Benthem's result still holds on *restricted classes* of models. In particular, Rosen proved that van Benthem's result is one of the few preservation results that transfers to the setting of finite models [18]; for a recent, rich source of van Benthem-style characterization results, see Dawar & Otto [7].
- Results characterizing *extensions* of basic modal logic as the bisimulation-invariant fragment of some extension of first-order logic. Here a key example is the theorem by Janin & Walukiewicz [13], characterizing the modal μ-calculus as the bisimulation-invariant fragment of monadic second-order logic.
- Results characterizing *variants* of modal logic as fragments of first-order logic that are invariant under some appropriate variant of the standard notion of bisimulation. Here we mention the result by Andréka, van Benthem and Németi, who characterized the guarded fragment [1] as the fragment of first-order logic that is invariant under guarded bisimulations; Otto [16] provides a overview of the results in this area, and of the (game-theoretic) methods used to prove these.
- Results on variants of modal logic where the modalities find their interpretation in different structures than the standard Kripke models. For instance, ten Cate, Gabelaia & Sustretov proved a van Benthem-style characterisation result for *topological* structures [20]. Recently, coalgebraic variations and generalizations of van Benthem's result have been obtained by Litak et alii [14].
- Clearly, researchers have been considering *combinations* of the above variations and generalizations; for example, Grädel, Hirsch & Otto [10] proved a characterization result for guarded fixpoint logic. An outstanding open problem is whether the Janin-Walukiewicz theorem also holds for finite models, or equivalently, whether Rosen's result can be extended to the modal μ-calculus.

In this paper we will look in some detail at the result by Janin and Walukiewicz, which we can formulate as:

$$\mu ML = MSO/\leftrightarroweq. \tag{2}$$

Taking a *coalgebraic* perspective, we will show how the proof of (2) can be decomposed into three more or less independent parts:

1. a (non-trivial) result showing that both the modal μ-calculus and (on the class of tree models) monadic second-order logic can be characterized by certain *automata*,
2. a fairly simple model-theoretic characterisation result in monadic first-order logic, and
3. a general result on coalgebra automata.

Towards the end of this section we briefly discuss the relation of this chapter to Johan's work, and to the theme of this volume, viz., Logical Dynamics. First we turn to a fairly detailed explanation of the above decomposition, motivating our coalgebraic perspective. For this purpose we need to introduce automata. We fix a set Q of proposition letters. Elements of $P\mathsf{Q}$ will be called *colors*, and given a valuation $V : \mathsf{Q} \to PS$, we define its associated *coloring* as the *transposed* map $V^\flat : S \to P\mathsf{Q}$ given by $V^\flat : s \to \{p \in \mathsf{Q} \mid p \in V(s)\}$.

The automata that we will consider here will be of the shape $\mathbb{A} = (A, \delta, \Omega)$, where A is a finite set of states and Ω is a parity map, $\Omega : A \to \mathbb{N}$. We will see a state $a \in A$ as a propositional variable, or, very much in the spirit of modal correspondence theory, as a monadic predicate. In this way, A provides a (monadic) first-order signature, which we will also denote as A. We define $\Phi^{=}(A)$ and $\Phi(A)$ as the sets of sentences (with and without equality, respectively) in this signature. We may *initialize* the automaton $\mathbb{A}$ by selecting an initial state $a \in A$. What shall interest us most is the *transition map* δ associating, with each state $a \in A$ and each color $c \in P\mathsf{Q}$, a first-order sentence $\delta(a,c) \in \Phi(A)$.

Acceptance of a pointed Kripke model $(\mathbb{S}, s)$ by such an initialized automaton $(\mathbb{A}, a)$ is defined in terms of an infinite two-player *acceptance game* $\mathscr{A}(\mathbb{A}, \mathbb{S})$. A match of this game consists of the two players, $\forall$ and $\exists$, moving a token from one position to another. In a so-called *basic position*, which is of the form $(a,s) \in A \times S$, $\exists$ needs to define an A-valuation $M : A \to PS$ on S, with the proviso that M turns the set of successors of s into a *structure* for the signature A where the formula $\delta(a, V^\flat(s))$ is *true*. That, is we require that

$$(\sigma_R(s), M\restriction_{\sigma_R(s)}) \models \delta(a,c),$$

where $c = V^\flat(s) \in P\mathsf{Q}$ is the color of s in $\mathbb{S}$, $\sigma_R(s)$ is the set of successors of s, and $M\restriction_{\sigma_R(s)}$ is the A-valuation M restricted to $\sigma_R(s)$. In other words, the pair $(\sigma_R(s), M\restriction_{\sigma_R(s)})$ is an A-structure (in the sense of first-order model theory), and it is the aim of $\exists$ to make the sentence $\delta(a, V^\flat(s))$ true in this structure by chosing an appropriate A-valuation M. Given such a choice $M : A \to PS$, the game moves on with $\forall$ picking a next basic position from the set $\{(b,t) \mid t \in M(b)\}$. In this way, a match proceeds from one basic position (a_i, s_i) to the next (a_{i+1}, s_{i+1}). An infinite match of this game is won by $\exists$ if the highest parity $\Omega(a_i)$ occurring infinitely often during this match is even.

If $\exists$ has a winning strategy in the instantiation of the game that starts at the basic position (a,s), we say that the initialized automaton $(\mathbb{A},a)$ *accepts* the pointed model $(\mathbb{S},s)$. Initialized automata thus determine classes of pointed Kripke models, and we may compare the expressive power of such automata to that of a logic such as the modal μ-calculus or monadic second-order logic.

Definition 1.1 Given a fragment Θ of monadic first-order logic (in the sense that Θ assigns to each A a set of sentences $\Theta(A) \subseteq \Phi^{=}(A)$), we obtain an associated class Aut_{Θ} of initialized automata $(\mathbb{A},a)$ by requiring that $\Theta(A)$ is the co-domain of the transition function of $\mathbb{A}$, that is, we have $\delta : A \times PQ \to \Theta(A)$. $\lhd$

Definition 1.2 We let $\Pi^{=}(A)$ and $\Pi(A)$ denote the sets of sentences in $\Phi^{=}(A)$, with and without equality, respectively, where each occurrence of a monadic predicate is positive. $\lhd$

In particular, we can now substantiate the claim that both monadic second-order logic (on the class of tree models) and the modal μ-calculus can be captured by automata-theoretic means. This link between logic and automata theory essentially goes back to the work of Rabin and Büchi on stream and tree automata. The two statements in Fact 1.3 below can be found in Walukiewicz [23] and Janin & Walukiewicz [12], respectively.

Fact 1.3 *1. On tree models, monadic second-order logic corresponds to* $\mathrm{Aut}_{\Pi^{=}}$.
2. The modal μ*-calculus corresponds to* Aut_{Π}.

The main point of this paper is that, for any fragment Θ of $\Phi^{=}$, properties of Θ-automata are determined by properties of Θ. In particular, given two distinct fragments Θ and Θ', we will see how the question whether $\mathrm{Aut}(\Theta)$ is the bisimulation invariant fragment of $\mathrm{Aut}(\Theta')$, may already be determined at the level of Θ and Θ'.

For this purpose we introduce the notion of *P-invariance*. We say that two A-structures (D,V) and (D',V') are P-equivalent, notation: $(D,V) \equiv_P (D',V')$, if for all $d \in D$ there is a $d' \in D'$ with the same A-color, and vice versa. A first-order sentence $\alpha \in \Phi^{=}(A)$ is *P-invariant* if $(D,V) \models \alpha \iff (D',V') \models \alpha$, for all pairs of P-equivalent A-structures (D,V) and (D',V'). (As we will see, this property is equivalent to being preserved under surjective homomorphisms.) Given two fragments Θ,Θ' of $\Phi^{=}$, we say that Θ *corresponds to the P-invariant fragment of* Θ' if any sentence $\alpha \in \Theta'(A)$ is P-invariant iff it is equivalent to a formula $\alpha^* \in \Theta(A)$. The 'fairly simple result in monadic first-order logic' mentioned as the second item above, can now be stated as follows.

Proposition 1.4 Π *corresponds to the P-invariant fragment of* $\Pi^{=}$.

Our observation is that the Janin-Walukiewicz Theorem is a direct *corollary* of Fact 1.3 and Proposition 1.4. To be more precise, we will define a translation $(\cdot)^*$ mapping a $\Pi^{=}(A)$-sentence α to $\Pi(A)$-sentence α^* satisfying

$$\alpha \equiv \alpha^* \text{ iff } \alpha \text{ is } P\text{-invariant.} \tag{3}$$

On the basis of this, we can present the proof of Janin & Walukiewicz as follows.

First, given a *MSO*-formula φ, consider the equivalent initialized $\Pi^=$-automaton $(\mathbb{A}_\varphi, a_\varphi)$ given by Fact 1.3(1). Where $\mathbb{A}_\varphi = (A, \delta, \Omega)$, with $\delta : A \times P\mathrm{Q} \to \Pi^=(A)$, define the Π-automaton $\mathbb{A}^*_\varphi := (A, \delta^*, \Omega)$, with $\delta^* : A \times P\mathrm{Q} \to \Pi(A)$ by putting $\delta^*(a,c) := (\delta(a,c))^*$. Let φ^* be the μML-formula that is equivalent to $(\mathbb{A}^*_\varphi, a_\varphi)$, given by Fact 1.3(2). Using (3) one may then show that for any pointed Kripke model $(\mathbb{S}, s)$ there is a pointed Kripke model $(\mathbb{S}', s')$, and a bounded morphism $f : \mathbb{S}' \to \mathbb{S}$ such that $fs' = s$, while for any *MSO*-formula φ we have

$$\mathbb{S}, s \Vdash \varphi^* \text{ iff } \mathbb{S}', s' \Vdash \varphi. \tag{4}$$

Now suppose that φ is a bisimulation-invariant *MSO*-formula. Then for any pointed Kripke model $(\mathbb{S}, s)$ we have that

$$\begin{array}{lll} \mathbb{S}, s \Vdash \varphi & \text{iff} \quad \mathbb{S}', s' \Vdash \varphi & \qquad\qquad \text{(assumption on } \varphi) \\ & \text{iff} \quad \mathbb{S}, s \Vdash \varphi^* & \qquad\qquad (4) \end{array}$$

Clearly this shows that φ is equivalent to φ^*, and since φ^* is a formula in the modal μ-calculus, this suffices to prove the Janin-Walukiewicz theorem.

In fact, the argument just given can be generalized to prove the following result.

Theorem 1.5 *Let Θ and Θ' be fragments of monadic first-order logic. If Θ corresponds to the P-invariant fragment of Θ', then* Aut_Θ *corresponds to the bisimulation-invariant fragment of* $\mathrm{Aut}_{\Theta'}$.

The second and main contribution of this paper is the observation that Theorem 1.5 is itself an instance of a more general phenomenon that is essentially *coalgebraic* in nature. Universal Coalgebra [19] provides the notion of a *coalgebra* as the natural mathematical generalization of state-based evolving systems such as streams, (infinite) trees, finite state automata, Kripke frames and models, (probabilistic) transition systems, and many others. Formally, a coalgebra is a pair $\mathbb{S} = (S, \sigma)$, where S is the carrier or state space of the coalgebra, and $\sigma : S \to TS$ is its unfolding or transition map. This approach combines simplicity with generality and wide applicability: many features, including input, output, nondeterminism, probability, and interaction, can easily be encoded in the coalgebra type T (formally an endofunctor on the category Set of sets as objects with functions as arrows).

Logic enters the picture if one wants to specify and reason about *behavior*, one of the most fundamental notions admitting a coalgebraic formalization. With Kripke structures constituting key examples of coalgebras, it should come as no surprise that most coalgebraic logics are some kind of modification or generalization of *modal logic* [5]. Moss [15] introduced a modality ∇_T generalizing the so-called 'cover modality' from Kripke structures to coalgebras of arbitrary type. This approach is uniform in the functor T, but as a drawback only works properly if T satisfies a certain category-theoretic property (viz., it should preserve weak pullbacks);

also the nabla modality is syntactically rather nonstandard. As an alternative, Pattinson [17] and others developed coalgebraic modal formalisms, based on a completely standard syntax, that work for coalgebras of arbitrary type. In this approach, the semantics of each modality is determined by a so-called *predicate lifting* (see Definition 2.15 below). Many well-known variations of modal logic in fact arise as the coalgebraic logic L_Λ associated with a set Λ of such predicate liftings; examples include both standard and (monotone) neighborhood modal logic, graded and probabilistic modal logic, coalition logic, and conditional logic.

In order to reason about *ongoing* coalgebraic behavior, modal logicians have introduced fixpoint extensions of coalgebraic logics [22, 4] and developed the corresponding automata theory [8]. For instance, each set Λ of predicate liftings comes with a modal logic L_Λ, a coalgebraic μ-calculus μL_Λ, and an equivalent class of automata Aut_Λ.

Kripke frames are coalgebras for the power set functor P, and each sentence α in monadic first-order logic induces a predicate lifting $\widehat{\alpha}$ for the power set functor. However, corresponding to the fact that we are looking at logics that are not bisimulation invariant, not all of these predicate liftings will be *natural* (in some technical sense to be defined below). In fact, we will introduce a coalgebraic novelty in this paper, in that we will consider non-natural predicate liftings for an arbitrary functor T. Generalizing the notion of P-invariance discussed above, we will define what it means for one set of predicate lifting to be the *natural fragment* of another set. Our coalgebraic generalization of Theorem 1.5 then roughly states the following:

> if Λ provides the T-natural fragment of Λ', then μL_Λ is the bisimulation-invariant fragment of $\mu L_{\Lambda'}$.

For a more precise formulation, we refer to Theorem 5.1 below.

To conclude this introduction, we briefly discuss the relation of this chapter with van Benthem's work. First of all, it may have struck the reader's attention that while van Benthem's Bisimulation Theorem concerns the bisimulation-invariant fragment of *first-order* logic, our focus is on monadic *second-order* logic. We certainly believe that our coalgebraic perspective has some bearing on first-order logic as well, but we will leave this topic for later work. The main reason for this is that we wanted to give a detailed account of the coalgebraic perspective on fixpoint logics and automata theory. Note that in a general coalgebraic context, it is always clear how to define modal fixpoint logics and their associated automata. This is not necessarily the case with first-order logic, although recently some interesting proposals have been made, see for instance Litak et alii [14].

Another matter concerns the link between this chapter and the volume's theme, viz., *Logical Dynamics*. Here, again, *coalgebra* is the key word: as mentioned, universal coalgebra is a very natural mathematical framework for the kind of state-based evolving systems that play a fundamental role in the study of dynamics. In particular, many of the game-like processes that van Benthem is interested in, allow for a coalgebraic presentation. From this perspective coalgebraic modal logics, and in particular their fixed-point variants, provide natural logics for representing dynamic phenomena. The question of bisimulation invariance then makes us focus on the power of logical languages to express those properties that are *relevant* from the perspective of modelling dynamics. As such, our chapter not only connects with van Benthem's earliest technical work, but also with his foundational studies to the nature of the dynamics of information-related processes.

2 Coalgebra and modal logic

This section contains an introducton to coalgebra and coalgebraic modal logic.

We assume familiarity with basic notions from category theory, but not going beyond categories, functors, and natural transformations. We let Set denote the category with sets as objects and functions as arrows. Functors that feature prominently in this paper are the *co-* and the *contravariant power set functor*, P and $\breve{P}$, respectively. Both act on objects by mapping a set S to its power set $PS = \breve{P}S$; a function $f : S' \to S$ is mapped by P to the direct image function $Pf : PS' \to PS$ given by $(Pf)X' := \{fs' \in S \mid s' \in X'\}$, and by $\breve{P}$ to the inverse image function $\breve{P}f : PS \to PS'$ given by $(\breve{P}f)X := \{s' \in S' \mid fs' \in X\}$.

2.1 Coalgebra

We start with introducing coalgebras and their morphisms.

Definition 2.1 Let $T : \mathsf{Set} \to \mathsf{Set}$ be a (covariant) set functor. A *T-coalgebra* is a pair $\mathbb{S} = \langle S, \sigma \rangle$ where S is a set and σ is a function $\sigma : S \to TS$. Elements of S are called *states* of the coalgebra and σ is called the *transition map* of *coalgebra map* of $\mathbb{S}$. We may refer to T as the *type* of $\mathbb{S}$. A *pointed T-coalgebra* is a pair $(\mathbb{S}, s)$ consisting of a T-coalgebra $\mathbb{S}$ and a state $s \in S$.

If, for a function $f : S' \to S$, the following diagram commutes:

$$\begin{array}{ccc} S' & \xrightarrow{f} & S \\ \downarrow{\scriptstyle \sigma'} & & \downarrow{\scriptstyle \sigma} \\ TS' & \xrightarrow{Tf} & TS \end{array} \tag{5}$$

we call f a *(coalgebra) morphism* from $\mathbb{S}' = \langle S', \sigma \rangle$ to $\mathbb{S} = \langle S, \sigma \rangle$, and write $f : \mathbb{S}' \to \mathbb{S}$. ◁

Convention 2.2 Throughout this paper we will discuss an arbitrary but fixed (covariant) set functor that we denote as $T : \mathsf{Set} \to \mathsf{Set}$.

Many structures that are well-known from theoretical computer science or from modal logic admit a natural presentation as coalgebras.

Example 2.3 (•) Kripke frames are coalgebras for the power set functor P: a Kripke frame $\langle S,R\rangle$ with $R \subseteq S \times S$ can be represented as the coalgebra $\langle S,\rho_R\rangle$, where ρ_R maps a state in S to the collection of its successors: $\rho_R : s \mapsto \{t \in S \mid Rst\}$. It is straightforward to verify that the notion of a coalgebra morphism for P-coalgebras coincides with that of a bounded morphism between Kripke frames. In other words, the category of P-coalgebras is isomorphic to that of Kripke frames (with bounded morphisms).

(•) Kripke models are coalgebras as well. Fix a set Q of proposition letters, and observe that the information given by a valuation $V : \mathsf{Q} \to PS$ can just as well be provided by its *transpose* $V^\flat : S \to P\mathsf{Q}$ given by $V^\flat : s \to \{p \in \mathsf{Q} \mid p \in V(s)\}$. On the basis of this, we may represent a Kripke model $\langle S,R,V\rangle$ as a a coalgebra $\langle S,\sigma_{V,R}\rangle$, where $\sigma_{R,V} : S \to P\mathsf{Q} \times PS$ is given by $\sigma_{R,V} : s \mapsto (V^\flat(s),\rho_R(s)$. In other words, Kripke models (over Q) are coalgebras for the functor $P_\mathsf{Q} := P\mathsf{Q} \times P-$.

(•) Recall that a deterministic finite state automaton (DFA) over a finite alphabet (or color set) C is a triple $\langle S,\delta,F\rangle$ with $\delta : S \times C \to S$ and $F \subseteq S$. Representing the transition map δ, through currying, by a function $\delta' : S \to S^C$, and the set F of accepting states by its characteristic function $\chi_F : S \to \{0,1\}$, we may think of this DFA as a coalgebra $\langle S,(\chi_F,\delta')\rangle$ for the functor $D_C := \{0,1\} \times (-)^C$. Here (as in subsequent examples) we omit to check that the *morphisms* induced by the coalgebra framework are the natural, standard ones.

(•) Given a set A of atomic actions, we can represent a transition system $(S,(R_a)_{a\in A})$, where each atomic action a is interpreted as a binary relation $R_a \subseteq S \times S$, as a coalgebra for the functor $(P-)^A$.

(•) Define the covariant set functor $N : \mathsf{Set} \to \mathsf{Set}$ as the composition of the contravariant power set with itself, $N := \breve{P} \circ \breve{P}$. Coalgebras for this functor correspond to the well-known *neighborhood* models in modal logic.

Restricting this example somewhat, we may obtain various interesting classes of structures. For instance, take the functor M given by $MS := \{\mathscr{U} \in NS \mid \mathscr{U}$ is upward closed with respect to $\subseteq\}$ and $Mf = Nf$. M-coalgebras are known in modal logic as monotone neighborhood frames.

(•) For a slightly more involved example, consider the finitary *multiset* or *bag* functor B_ω. This functor takes a set S to the collection $B_\omega S$ of maps $\mu : S \to \mathbb{N}$ of finite support (that is, for which the set $Supp(\mu) := \{s \in S \mid \mu(s) > 0\}$ is finite), while its action on arrows is defined as follows. Given an arrow $f : S \to S'$ and a map $\mu \in B_\omega S$, we define $(B_\omega f)(\mu) : S' \to \mathbb{N}$ by putting $(B_\omega f)(\mu)(s') := \sum\{\mu(s) \mid f(s) = s'\}$. Coalgebras for this functor are *weighted* transition systems, where each transition from one state to another carries a weight given by a natural number. Observe that a Kripke frame $\langle S,R\rangle$ can be seen as a B_ω-coalgebra $\langle S,\rho'_R\rangle$ by putting $\rho'_R(s)(t) = 1$ if Rst, and $\rho_R(s)(t) = 0$ otherwise.

(•) As a variant of B_ω, consider the finitary probability functor D_ω, where $D_\omega S = \{\delta : S \to [0,1] \mid Supp(\delta)$ is finite and $\sum_{s\in S} \delta(s) = 1\}$, while the action of D_ω on arrows is just like that of B_ω. Coalgebras for this functor are known as *Markov chains*.

The connection between Kripke frames and Kripke models can be generalized to coalgebras of arbitrary type.

Definition 2.4 Let T be a set functor and let Q be a set of proposition letters. We define the set functor $T_{\mathsf{Q}} := P\mathsf{Q} \times T$. A *$T$-model over* Q is a pair $(\mathbb{S}, V)$ consisting of a T-coalgebra $\mathbb{S} = \langle S, \sigma \rangle$ and a Q-*valuation* V on S, that is, a function $V : \mathsf{Q} \to PS$. The *coloring* associated with V is the *transpose* map $V^{\flat} : S \to P\mathsf{Q}$ given by

$$V^{\flat}(s) := \{p \in \mathsf{Q} \mid s \in V(p)\}.$$

Hence the pair $(\mathbb{S}, V)$ induces a T_{Q}-coalgebra $\langle S, (V^{\flat}, \sigma) \rangle$. ◁

Convention 2.5 In the remainder of this paper we will identify T-models over Q with the T_{Q}-coalgebras they induce. For instance, morphisms between T-models are implicitly defined as coalgebra morphisms between the induced T_{Q}-coalgebras. That is, a map $f : S' \to S$ is a morphism from $(\mathbb{S}', V')$ to $(\mathbb{S}, V)$ if (1) $f : \mathbb{S}' \to \mathbb{S}$ and (2) $s' \in V'(p)$ iff $fs' \in V(p)$, for all $s' \in S'$ and all $p \in \mathsf{Q}$.

The key coalgebraic notion of equivalence is that of two pointed coalgebras being *behaviorally equivalent*. In case the functor T admits a coalgebra $\mathbb{Z} = \langle Z, \zeta \rangle$ which is *final* (in the sense that for every T-coalgebra $\mathbb{S}$ there is a unique coalgebra morphism $!_{\mathbb{S}} : \mathbb{S} \to \mathbb{Z}$), the elements of Z often provide an intuitive encoding of the notion of *behaviour*, and the unique coalgebra morphism $!_{\mathbb{S}}$ can be seen as a map that assigns to a state x in $\mathbb{S}$ its behaviour. In this case we call two pointed coalgebras, $(\mathbb{S}, s)$ and $(\mathbb{S}', s')$, *behaviorally equivalent* if $!_{\mathbb{S}}s = !_{\mathbb{S}'}s'$. In the general case, when we may not assume the existence of a set-based final coalgebra, we define the notion as follows.

Definition 2.6 Let $(\mathbb{S}, s)$ and $(\mathbb{S}', s')$ be two pointed coalgebras. If there are coalgebra morphisms f, f' with a common codomain such that $f(s) = f'(s')$, we call the two pointed coalgebras *behaviorally equivalent*, notation: $\mathbb{S}, s \simeq \mathbb{S}', s'$. We will often apply this notion to the states s and s'. ◁

Remark 2.7 In many cases, including those of Kripke frames and models, behavioral equivalence is the same as *bisimilarity*, but in cases where the two notions diverge, behavioral equivalence is the more natural notion. For the purpose of this paper it suffices to work with behavorial equivalence, and we do not need to discuss generalisations of the notion of a bisimulation to coalgebras of arbitrary type, referring the reader to [21] for more information.

2.2 Coalgebraic Logics

It will be convenient to have a rather abstract notion of coalgebraic languages and logics for T (so that for instance, we can think of automata as proper formulas).

Definition 2.8 An abstract coalgebraic logic is a pair $(L, \Vdash^L)$ such that L is a set and $\Vdash$ is a collection of relations associating with each T-coalgebra $\mathbb{S} = \langle S, \sigma, V \rangle$ a binary relation $\Vdash^L_{\mathbb{S}} \subseteq S \times L$. The set L is called the *language* of the logic, and its elements will be called *formulas*. If $s \Vdash^L_{\mathbb{S}} \varphi$ we say that the formula φ is *true* or *satisfied* at s in $\mathbb{S}$, and we will often write $\mathbb{S}, s \Vdash \varphi$.

The satisfaction relation $\Vdash^L_{\mathbb{S}}$ induces a *meaning function* $[\![\cdot]\!]^{\mathbb{S}} : L \to PS$ given by

$$s \in [\![\varphi]\!]^{\mathbb{S}} \quad \text{iff} \quad s \Vdash^L_{\mathbb{S}} \varphi. \tag{6}$$

◁

Example 2.9 Let us see how monadic second-order logic fits as a coalgebraic logic for P_{Q}-coalgebras (Kripke models over some fixed set Q of proposition letters). Clearly we may also see elements of Q as monadic predicate symbols.

To define the syntax of this logic, let $\mathsf{IVar} = \{u, v, \ldots\}$ be a set of individual (first-order) variables, and let $\mathsf{Var} = \{x, y, \ldots\}$ be a set of objects that one may think of alternatively as propositional variables or monadic predicate (that is, second-order) variables. Define the set of *MSO*(Q)*-formulas* by the following grammar:

$$\varphi \ ::= \ p(v) \mid x(v) \mid Ruv \mid \bot \mid \neg\varphi \mid \varphi_0 \vee \varphi_1 \mid \exists v.\varphi \mid \exists x.\varphi,$$

where $p \in \mathsf{Q}$, $v \in \mathsf{IVar}$, and $x \in \mathsf{Var}$. The interpretation of this language on a Kripke model $\langle S, R, V \rangle$ is standard.

Finally, we define $MSO_v(\mathsf{Q})$ as the set of $MSO(\mathsf{Q})$-formulas $\varphi(v)$ that contain a single free individual variable v, and no free variables in Var. (We need the free variable v in order to interpret formulas in *pointed* models.) Thus we obtain a coalgebraic logic $(MSO_v(\mathsf{Q}), \Vdash^{MSO})$ by putting $\mathbb{S}, s \Vdash \varphi$ iff $\mathbb{S} \models \varphi(s)$.

Coalgebraic logics naturally induce equivalence relations between formulas, and between pointed coalgebras.

Definition 2.10 Let $(L, \Vdash)$ be a coalgebraic logic. Two formulas φ and ψ are called *equivalent*, notation: $\varphi \equiv_{L,\Vdash} \psi$, if for all pointed coalgebras $(\mathbb{S}, s)$ we have $\mathbb{S}, s \Vdash \varphi \iff \mathbb{S}, s \Vdash \psi$.

Similarly, two pointed coalgebras $(\mathbb{S}, s)$ and $(\mathbb{S}', s')$ are called *equivalent*, notation: $\mathbb{S}, s \equiv^{L,\Vdash} \mathbb{S}', s'$, if $\mathbb{S}, s \Vdash \varphi \iff \mathbb{S}', s' \Vdash \varphi$, for all $\varphi \in L$. ◁

Since the satisfaction relation is usually determined by the language, in practice we will often blur the distinction between logics and their languages. For instance, we will write $\equiv^L$ rather than $\equiv^{(L,\Vdash)}$, etc.

Generally, in abstract model theory, an abstract logic is required not to distinguish isomorphic structures. Clearly such a condition would make sense here as well, but it is not relevant to our story. On the other hand, a condition that features crucially in our story is the requirement that coalgebraic logics cannot distinguish behaviorally equivalent states. Since we are generally much more interested in the behavior of a system then in its precise representation, this so-called *adequacy* property is a very natural one.

Definition 2.11 A formula φ is *behaviorally invariant* if for all pairs of behaviorally equivalent pointed coalgebras $\mathbb{S}, s \simeq \mathbb{S}', s'$ it holds that $\mathbb{S}, s \Vdash \varphi \iff \mathbb{S}', s' \Vdash \varphi$. A coalgebraic language is *behaviorally invariant* or *adequate* if all its formulas are behaviorally invariant, or equivalently, if $\simeq_T \subseteq \equiv^L$. $\triangleleft$

We have now arrived at the central notion in this paper, namely, that of one logic corresponding to the behaviorally invariant fragment of another.

For its definition, observe that if $(L, \Vdash)$ is a coalgebraic logic, then any set $L' \subseteq L$ induces a logic $(L', \Vdash\restriction_{L'})$, where $\Vdash\restriction_{L'}$ is the obviously defined restriction of the relation $\Vdash$ to L'. In the sequel we will simply write $\Vdash$ rather than $\Vdash\restriction_{L'}$.

Definition 2.12 Let $(L, \Vdash)$ be a coalgebraic logic, and let $L' \subseteq L$. We say that L' *corresponds to the behaviorally invariant fragment of* L, notation: $L' \equiv L/T$, if (1) $(L', \Vdash)$ is behaviorally invariant, and (2) every behaviorally invariant formula $\varphi \in L$ is equivalent to some formula $\varphi' \in L'$. $\triangleleft$

The technical work in this paper will be based on a stronger, somewhat more 'constructive' version of this notion.

Definition 2.13 Let $(L, \Vdash)$ be a coalgebraic logic, and let $L' \subseteq L$. We say that L' *strongly corresponds to the behaviorally invariant fragment of* L, notation: $L' \equiv^s L/T$, if (1) $(L', \Vdash)$ is behaviorally invariant, and (2$'$) there is a translation $(\cdot)^* : L \to L'$ and a map associating with each pointed coalgebra $(\mathbb{S}, s)$ a pointed coalgebra $(\mathbb{S}', s')$, together with a morphism $f : (\mathbb{S}', s') \to (\mathbb{S}, s)$ such that

$$\mathbb{S}, s \Vdash \varphi^* \quad \text{iff} \quad \mathbb{S}', s' \Vdash \varphi. \tag{7}$$

for all formulas $\varphi \in L$. $\triangleleft$

The following proposition justifies our terminology.

Proposition 2.14 *If $L' \equiv^s L/T$ then $L' \equiv L/T$.*

Proof. Assume that L' strongly corresponds to the behaviorally invariant fragment of L, via the translation $(\cdot)^* : L \to L'$, and let φ be an arbitrary behaviorally invariant formula in L. In order to prove the Proposition, it suffices to show that $\varphi \equiv \varphi^*$. For this purpose, take an arbitrary pointed coalgebra $(\mathbb{S}, s)$. By our assumption there is a morphism $f : \mathbb{S}' \to \mathbb{S}$ and a state s' in $\mathbb{S}'$ with $fs' = s$, and satisfying

$$\mathbb{S}, s \Vdash \varphi \quad \text{iff} \quad \mathbb{S}', s' \Vdash \varphi \quad \text{iff} \quad \mathbb{S}, s \Vdash \varphi^*. \tag{8}$$

Here the first equivalence is by our assumption on φ, and the second equivalence is by (7). The equivalence of φ and φ^* is immediate from (8). QED

2.3 Predicate Liftings

Many coalgebraic logics are induced by a set of so-called predicate liftings. In this subsection we will be interested in T-models; we fix a set Q of proposition letters.

Definition 2.15 An n-ary *predicate lifting for T* is a collection λ of maps, associating a function

$$\lambda_S : (PS)^n \to PTS$$

with each set S. ◁

In other words, an n-ary predicate lifting λ associates, with each set S, a map that yields a subset $\lambda_S(X_1,\ldots,X_n) \subseteq TS$ for each n-tuple $X_1,\ldots,X_n$ of subsets of S.

Note that our definition deviates from the usual one in that we do not require predicate liftings to be *natural* (see Definition 2.23).

Example 2.16 Here are some predicate liftings for the functors discussed in Example 2.3.

(•) Given a set S, a unary predicate lifting λ for the power set functor yields a map $\lambda_S : PS \to PPS$. Here are three examples, $\Diamond$, $\Box$ and ∞:

$$\begin{aligned}
&\Diamond_S : X \mapsto \{D \in PS \mid D \cap X \neq \varnothing\},\\
&\Box_S : X \mapsto \{D \in PS \mid D \subseteq X\},\\
&\infty_S : X \mapsto \{D \in PS \mid |D \cap X| \geq \omega\}.
\end{aligned}$$

For an example of a binary predicate lifting, consider the following definition, for a set S:

$$\star_S : \quad (X,Y) \;\mapsto\; \{D \in S \mid X \subseteq D \subseteq Y\}.$$

(•) A *nullary* predicate lifting λ assigns to each set S, a function λ_S from $(PS)^0$ to PTS; such a function can be identified with a subset of TS that we will also denote as λ_S. As a particularly interesting example, consider the functor T_{Q}. With each proposition letter $p \in \mathsf{Q}$ we may associate a nullary predicate lifting $\underline{p}$ by defining, for each set S, the following subset of $T_{\mathsf{Q}}S$:

$$\underline{p}_S := \{(\Pi,\tau) \in P\mathsf{Q} \times TS \mid p \in \Pi\}.$$

(•) Regarding the functor D_C corresponding to finite state automata over C, consider the nullary predicate lifting $\surd$ and the unary ⓒ (for any $c \in C$), defined, for a set S, by

$$\begin{aligned}
&\surd_S \;\;:=\;\; \{(i,f) \in 2 \times S^C \mid i = 1\},\\
&ⓒ_S : X \mapsto \{(i,f) \in 2 \times S^C \mid f(c) \in X\}.
\end{aligned}$$

(•) For the functor P^A of A-labelled transition systems, consider the unary lifting $\langle a \rangle$ given by

$$\langle a \rangle_S : X \mapsto \{D \in (PS)^A \mid D(a) \cap X \neq \varnothing\},$$

(•) With respect to the neighborhood functor N, we define a unary predicate lifting $\Diamond$ by putting, for a set S:

$$\Diamond_S : X \mapsto \{\mathscr{A} \in NS \mid X \in \mathscr{A}\}.$$

(•) Finally, consider the functor B_ω. Given a natural number $k \in \omega$, we define the predicate lifting $\underline{k}$ by putting

$$\underline{k}_S : X \mapsto \{\mu \in B_\omega S \mid \sum_{x \in X} \mu(x) \geq k\}.$$

Definition 2.17 With each predicate lifting λ we associate a modality $\heartsuit_\lambda$ with the same arity as λ. Given a set Λ of predicate liftings, we obtain the modal language $L_\Lambda(\mathsf{Q})$ by defining its set of formulas by the following grammar:

$$\varphi ::= p \mid \bot \mid \neg\varphi \mid \varphi_0 \vee \varphi_1 \mid \heartsuit_\lambda(\varphi_1, \ldots, \varphi_n)$$

where $p \in \mathsf{Q}$, and $\lambda \in \Lambda$ is n-ary. ◁

Definition 2.18 Let Λ be a set of predicate liftings. For any T-model $\mathbb{S} = \langle S, \sigma, V \rangle$, by induction on the complexity of L_Λ-formulas, we define the *meaning* function $[\![\cdot]\!]^{\mathbb{S}} : L_\Lambda \to PS$:

$$\begin{array}{ll}
[\![\bot]\!]^{\mathbb{S}} & := \varnothing \\
[\![p]\!]^{\mathbb{S}} & := V(p) \\
[\![\neg\varphi]\!]^{\mathbb{S}} & := S \setminus [\![\varphi]\!]^{\mathbb{S}} \\
[\![\varphi_0 \vee \varphi_1]\!]^{\mathbb{S}} & := [\![\varphi_0]\!]^{\mathbb{S}} \cup [\![\varphi_1]\!]^{\mathbb{S}} \\
[\![\heartsuit_\lambda(\varphi_1, \ldots, \varphi_n)]\!]^{\mathbb{S}} & := (\breve{P}\sigma)(\lambda_S([\![\varphi_1]\!]^{\mathbb{S}}, \ldots, [\![\varphi_n]\!]^{\mathbb{S}})).
\end{array}$$

The meaning function $[\![\cdot]\!]^{\mathbb{S}}$ induces a satisfaction relation $\Vdash_{\mathbb{S}}$ given by (6). ◁

In terms of the satisfaction relation $\Vdash$, the meaning of the modality $\heartsuit_\lambda$ is given by

$$\mathbb{S}, s \Vdash \heartsuit_\lambda(\varphi_1, \ldots, \varphi_n) \quad \text{iff} \quad \sigma(s) \in \lambda_S([\![\varphi_1]\!], \ldots, [\![\varphi_n]\!]).$$

Example 2.19 (•) It is easy to see that, for a Kripke model $\mathbb{S} = \langle S, R, V \rangle$, we have

$$\begin{array}{lcl}
\mathbb{S}, s \Vdash \heartsuit_\Diamond \varphi & \iff & \mathbb{S}, t \Vdash \varphi \text{ for some } t \in R(s) \\
\mathbb{S}, s \Vdash \heartsuit_\Box \varphi & \iff & \mathbb{S}, t \Vdash \varphi \text{ for all } t \in R(s) \\
\mathbb{S}, s \Vdash \heartsuit_\infty \varphi & \iff & \mathbb{S}, t \Vdash \varphi \text{ for infinitely many } t \in R(s) \\
\mathbb{S}, s \Vdash \heartsuit_\star(\varphi, \psi) & \iff & \mathbb{S}, t \Vdash \varphi \text{ for all } t \in R(s), \text{ and } Rsu \text{ for all } u \text{ with } \mathbb{S}, u \Vdash \psi.
\end{array}$$

The first two examples shows in particular that the well-known diamond and box operator from modal logic are coalgebraic modalities indeed.

(•) The definition of the predicate lifting $\underline{p}$, for a proposition letter $p \in \mathsf{Q}$, ensures that for any T-model $\mathbb{S} = (S, \sigma, V)$ we have

$$\begin{array}{llll}
\mathbb{S}, s \Vdash \underline{p} & \text{iff} & p \in V^\flat(p) & \text{(semantics } \underline{p}) \\
& \text{iff} & s \in V(p) & \text{(definition } V^\flat) \\
& \text{iff} & \mathbb{S}, s \Vdash p & \text{(semantics } p)
\end{array}$$

(•) For a model $\mathbb{S}$ based on a finite state automaton $\langle S, F, \delta \rangle$, we have

$$\begin{array}{lcl}
\mathbb{S}, s \Vdash \heartsuit_{\surd} \varphi & \iff & s \in F, \\
\mathbb{S}, s \Vdash \heartsuit_{©} \varphi & \iff & \mathbb{S}, \delta(c, s) \Vdash \varphi.
\end{array}$$

(•) If $\mathbb{S}$ is a model based on an A-labelled transition system, we find

$$\mathbb{S}, s \Vdash \heartsuit_{\langle a \rangle} \varphi \iff \mathbb{S}, t \Vdash \varphi \text{ for some } t \in R_a(s)$$

(•) For a neighborhood model $\mathbb{S}$ we obtain

$$\mathbb{S}, s \Vdash \heartsuit_{\diamond} \varphi \iff [\![\varphi]\!]^{\mathbb{S}} \in \sigma(s),$$

showing that classical modal logic is a coalgebraic logic indeed.

(•) Finally, suppose that we consider a Kripke frame as a coalgebra $\mathbb{S}$ for the functor B_ω. Then for any natural number $k \in \omega$ we obtain

$$\mathbb{S}, s \Vdash \heartsuit_{\underline{k}} \varphi \iff s \text{ has at least } k \text{ successors } t \text{ such that } \mathbb{S}, t \Vdash \varphi.$$

In other words, *graded modal logic* can be presented as a coalgebraic logic too.

We now turn to *coalgebraic μ-calculi*, that is, extensions of coalgebraic logics with fixpoint operators. In order to guarantee well-definedness of the semantics, we need to restrict attention to monotone predicate liftings.

Definition 2.20 An n-ary predicate lfiting λ is *monotone* if for every set S, the map $\lambda_S : (PS)^n \to PTS$ is order-preserving in each coordinate (with respect to the subset order). The predicate lifting $\overline{\lambda} : (P-)^n \to PT-$, given by

$$\overline{\lambda}_S(X_1, \ldots, X_n) := TS \setminus \lambda_S(S \setminus X_1, \ldots, S \setminus X_1),$$

is called the *(Boolean) dual* of Λ. ◁

Since we are working with a fixed set Q of proposition letters, we need to introduce a set $\mathsf{Var} = \{x, y, z, x_0, \ldots\}$ of propositional *variables* in our formal set-up.

Definition 2.21 Let Λ be a set of monotone predicate liftings. The modal language $\mu L_\Lambda(\mathsf{Q})$ is defined by the following grammar:

$$\varphi ::= p \mid x \mid \bot \mid \neg\varphi \mid \varphi_0 \vee \varphi_1 \mid \heartsuit_\lambda(\varphi_1, \ldots, \varphi_n) \mid \mu x.\varphi$$

where $p \in \mathsf{Q}$, $x \in \mathsf{Var}$, and the application of the fixpoint operator μx is subject to the proviso that all occurrences of x in φ are positive (that is, under an even number of negations).

The sets of free and bound variables in a formula is defined as usual, and we define a $\mu L_\Lambda(\mathsf{Q})$-*sentence* as a formula with no bound variables. ◁

The semantics of this language contains no surprises.

Definition 2.22 Let Λ be a set of monotone predicate liftings, and let $\mathbb{S}$ be a T-model. An *assignment* is a map $h : \mathsf{Var} \to PS$ assigning a meaning to each variable in Var. By induction on the complexity of $\mu L_\Lambda(\mathsf{Q})$-formulas we define, for each assignment h, a meaning function $[\![\cdot]\!]^{\mathbb{S},h} : \mu L_\Lambda(\mathsf{Q}) \to PS$. Here we only give the following two clauses:

$$\begin{array}{ll} [\![x]\!]^{\mathbb{S},h} & := h(x) \\ [\![\mu x.\varphi)]\!]^{\mathbb{S},h} & := \bigcap_{X \subseteq S} [\![\varphi]\!]^{\mathbb{S},h[x\mapsto X]}, \end{array}$$

where $h[x \mapsto X]$ is the assignment sending every $y \in \mathsf{Var}$ to $V(y)$, except for x which is sent to X. ◁

It is a routine exercise to verify that with this definition, the formula $\mu x.\varphi$ is interpreted as the least fixed point of the formula $\varphi(x)$.

Returning to the notion of adequacy, for logics generated by predicate liftings this property follows from naturality of the predicate liftings.

Definition 2.23 A predicate lifting λ is *natural for T* if it is a natural transformation $\lambda : (\breve{P}-)^n \dot{\to} (\breve{P}T-)$, i.e. if for each function $f : S' \to S$, the following diagram commutes:

$$\begin{array}{ccc} S & \quad & (PS)^n \xrightarrow{\lambda_S} PTS \\ \uparrow f & & (\breve{P}f)^n \downarrow \qquad\qquad \downarrow \breve{P}Tf \\ S' & & (PS')^n \xrightarrow{\lambda_{S'}} PTS' \end{array} \tag{9}$$

A set of predicate liftings is natural for T if this property applies to each of its members. ◁

Proposition 2.24 *If Λ is natural for T, then $L_\Lambda(\mathsf{Q})$ and $\mu L_\Lambda(\mathsf{Q})$ are behaviorally invariant for each set Q.*

Proof. By definition of behavioral equivalence it suffices to prove that for every morphism $f : \mathbb{S}' \to \mathbb{S}$, every formula φ, and every state $s' \in S'$, it holds that

$$\mathbb{S}', s' \Vdash \varphi \quad \text{iff} \quad \mathbb{S}, fs' \Vdash \varphi. \tag{10}$$

In terms of the meaning function $[\![\cdot]\!]$, we may prove equivalently that for all formulas φ

$$[\![\varphi]\!]^{\mathbb{S}'} = (\breve{P}f)[\![\varphi]\!]^{\mathbb{S}}. \tag{11}$$

We prove (11) by induction on φ. For the key inductive clause, assume that $\varphi = \heartsuit_\lambda \psi$ (for notational simplicity we assume that λ is unary). Then

$$\begin{array}{rll} [\![\varphi]\!]^{\mathbb{S}'} & = (\breve{P}\sigma')\lambda_{S'}([\![\psi]\!]^{\mathbb{S}'}) & \text{(semantics of } \heartsuit_\lambda) \\ & = (\breve{P}\sigma')\lambda_{S'}(\breve{P}f)([\![\psi]\!]^{\mathbb{S}}) & \text{(induction hypothesis)} \\ & = (\breve{P}\sigma')(\breve{P}Tf)\lambda_S([\![\psi]\!]^{\mathbb{S}}) & \text{(naturality of } \lambda \text{ (9))} \\ & = (\breve{P}f)(\breve{P}\sigma)\lambda_S([\![\psi]\!]^{\mathbb{S}}) & (f \text{ a morphism}) \\ & = (\breve{P}f)[\![\varphi]\!]^{\mathbb{S}'} & \text{(semantics of } \heartsuit_\lambda) \end{array}$$

QED

We leave it as an exercise for the reader to check that all predicate liftings of Example 2.16 are natural, except ∞ and $\star$.

For concreteness, we define (basic) modal logic and the modal μ-calculus as follows.

Definition 2.25 We define $ML(\mathsf{Q})$, *basic modal logic* over Q, as the coalgebraic logic $L_{\Diamond,\Box}(\mathsf{Q})$, and $\mu ML(\mathsf{Q})$, the *modal μ-calculus* over Q, as its fixpoint extension: $\mu ML(\mathsf{Q}) := \mu L_{\Diamond,\Box}(\mathsf{Q})$. ◁

3 Coalgebra Automata & MSO

As usual in the theory of fixpoint logics, it will be easier to work with *automata* rather than with formulas. In this section we define the notion of a *coalgebra automaton* associated with a set Λ of monotone predicate liftings (and a set Q of proposition letters). These devices will provide the automata-theoretic counterpart to the coalgebraic μ-calculus, and we will see how monadic second-order can be captured by automata (and thus correspond to a coalgebraic fixed point logic), when we restrict attention to the class of tree models. This is also a good place to introduce the one-step perspective on coalgebraic logic, a key coalgebraic concept.

3.1 One-step syntax and semantics

Definition 3.1 Given a set A of propositional variables and a collection Λ of predicate liftings, we define the set $L^1_\Lambda(A)$ via the following grammar:

$$\varphi \ ::= \ \heartsuit_\lambda(a_1,\ldots,a_n) \mid \bot \mid \top \mid \varphi_0 \vee \varphi_1 \mid \varphi_0 \wedge \varphi_1$$

where $\lambda \in \Lambda$ is n-ary, and $a_i \in A$, for each i. Elements of $L^1_\Lambda(A)$ will be called *rank-1 Λ-formulas over A*, or simply: *rank-1* formulas. ◁

Observe that we do not allow negations to occur in rank-1 formulas. Given a set S, we can interpret rank-1 formulas over A as subsets of TS, once we have been given a valuation assigning a meaning to the variables in A.

Definition 3.2 Given sets A and S, an *A-valuation* or *A-marking* on S is a map $V : A \to PS$. Given such a valuation, we inductively define the *one-step* satisfaction relation $\Vdash^1_V \subseteq TS \times L^1_\Lambda(A)$. For the basic formulas of the form $\heartsuit_\lambda(a_1,\ldots,a_n)$ we put, for $\tau \in TS$,

$$\tau \Vdash^1_V \heartsuit_\lambda(a_1,\ldots,a_n) \quad \text{iff} \quad \tau \in \lambda_S(V(a_1),\ldots,V(a_n)),$$

while inductively each Boolean connective receives its standard set-theoretic interpretation. Frequently we will write $TS,V,\tau \Vdash^1 \varphi$ rather than $\tau \Vdash^1_V \varphi$. ◁

The link with the ordinary semantics for coalgebraic logic is given by the coalgebra map. That is, given a T-coalgebra $\mathbb{S} = (S, \sigma : S \to TS)$ and a valuation $V : A \to PS$, we have

$$(\mathbb{S}, V), s \Vdash \heartsuit_\lambda(a_1, \ldots, a_n) \quad \text{iff} \quad TS, V, \sigma(s) \Vdash^1 \heartsuit_\lambda(a_1, \ldots, a_n).$$

3.2 Coalgebra Automata

We are now ready to introduce coalgebra automata, which will be parametrized by a set Λ of predicate liftings, and a set Q of proposition letters.

Definition 3.3 Let Λ be a set of monotone predicate liftings for T. A (Λ, Q)-automaton $\mathbb{A}$ is a triple $\mathbb{A} = (A, \delta, \Omega)$, where A is a finite set of states, $\delta : A \times P\mathsf{Q} \to L^1_\Lambda(A)$ is the transition map, and $\Omega : A \to \mathbb{N}$ is a parity map.

An *initialized* automaton is a pair $(\mathbb{A}, a_I)$ where $a_I \in A$. The class of initialized (Λ, Q)-automata is denoted as $\mathrm{Aut}_\Lambda(\mathsf{Q})$. $\lhd$

The semantics of these automata is defined in terms of an infinite parity graph game. We assume that the reader has some familiarity with these games, and with associated notions such as matches, (positional) strategies, etc. (Details can be found in [11]).

Definition 3.4 Let $\mathbb{S} = \langle S, \sigma, V \rangle$ be a T-model and let $\mathbb{A} = (A, \delta, \Omega)$ be a (Λ, Q)-automaton. The associated *acceptance game* $\mathscr{A}(\mathbb{A}, \mathbb{S})$ is the parity game given by the table below.

Position	Player	Admissible moves	Priority
$(a,s) \in A \times S$	$\exists$	$\{M : A \to PS \mid TS, V, \sigma(s) \Vdash^1 \delta(a, M^\flat(s))\}$	$\Omega(a)$
$M \in (PS)^A$	$\forall$	$\{(b,t) \mid t \in M(b)\}$	0

A pointed coalgebra $(\mathbb{S}, r)$ is *accepted* by an initialized automaton $(\mathbb{A}, a_I)$, notation: $\mathbb{S}, r \Vdash (\mathbb{A}, a_I)$, if the pair (a_I, r) is a winning position for player $\exists$ in $\mathscr{A}(\mathbb{A}, \mathbb{S})$. $\lhd$

Observe that the acceptance game of (Λ, Q)-automata proceeds in *rounds* moving from one basic position in $A \times S$ to another. In each round, at position (a, s) first $\exists$ picks an A-marking M on S that makes the depth-one formula $\delta(a, V^\flat(s))$ true at $\sigma(s)$. Looking at this $M : A \to PS$ as a binary relation $\{(b,t) \mid t \in M(b)\}$ between A and S consisting of *witnesses* picked by $\exists$, it is $\forall$ who closes the round by choosing a witness (b, t) from this relation, which will then serve as the starting position of the next round of the game.

Acceptance games feature both maps of the form $V : \mathsf{Q} \to PS$ (valuations that are part of the models $\mathbb{S}$ on which the automaton operates) and maps $M : A \to PS$ (that correspond to sets of witnesses picked by $\exists$). To emphasize the distinct roles that these two kinds of maps play, we will refer to the first ones as *valuations* and to the second ones as *markings* in situations where both types occur.

The following proposition instantiates the connection between fixpoint logics and parity automata in our setting of coalgebraic logic. Observe that we may think of the set $\mathrm{Aut}_{\Lambda,\mathsf{Q}}$ of initialized (Λ,Q)-automata, together with the acceptance relation, as a coalgebraic logic.

Proposition 3.5 *Let Λ be a set of monotone predicate liftings for T which is closed under taking Boolean duals. Then $\mu L_\Lambda(\mathsf{Q}) \equiv \mathrm{Aut}_\Lambda(\mathsf{Q})$.*

We omit the (completely routine) proof.

3.3 MSO as a coalgebraic fixpoint logic

We will now see how we may think of MSO as a coalgebraic fixpoint logic, at least if we restrict our attention to tree models.

Definition 3.6 A *tree model* is a Kripke model $\langle S,R,V\rangle$ in which there is a unique path to each point from a certain, fixed, state called the *root* of the tree. ◁

Throughout this subsection we fix a set A of syntactic objects that, as mentioned in the introduction, we may think of as either propositional variables or monadic *predicate symbols* of some first-order language. In other words, we will see A as a first-order signature (that we will also denote as A). The formulas in this language are given by the following grammar:

$$\alpha \ ::= \ x = y \mid a(x) \mid \neg\alpha \mid \alpha \vee \alpha \mid \exists x.\,\alpha$$

If we use the notation $\alpha(a_1,\ldots,a_n)$ for a sentence in this language, this indicates that the predicate symbols in α (not the first-order variables) are among $a_1,\ldots,a_n$.

Definition 3.7 We let $\Phi^{=}(A)$ and $\Phi(A)$ denote the sets of all first-order sentences over the signature A, respectively with and without identity. $\Pi^{=}(A)$ and $\Pi(A)$ are the positive fragments of $\Phi^{=}(A)$ and $\Phi(A)$, respectively, consisting of those sentences in which all occurrences of atomic formulas are positive. ◁

We have now arrived at the key observation underlying this paper. In the case that our coalgebra functor T is the *power set functor*, given an A-valuation V on S and an element $D \in PS$, we may think of the pair $\langle D,V\rangle$ as a *structure* for the signature A (in the sense of first-order model theory), where the predicate symbol $a \in A$ is interpreted as the subset $V(a)\cap D \subseteq D$. Consequently, we will now see that each first-order sentence $\alpha(a_1,\ldots,a_n)$ of this signature induces a (not necessarily natural) n-ary predicate lifting $\widehat{\alpha}$ for the power set functor.

Definition 3.8 Let $\alpha(a_1,\ldots,a_n) \in \Phi^{=}(A)$. For any set S, α induces a map $\widehat{\alpha}$: $(PS)^n \to PPS$, given by

$$\widehat{\alpha}(X_1,\ldots,X_n) := \{D \in PS \mid \langle D,V_{\overline{X}}\rangle \models \alpha\},$$

where $V_{\overline{X}}$ is the A-valuation on S given by $V_{\overline{X}}(a_i) := X_i$. By a slight abuse of notation, for any fragment Θ of $\Phi^=$, we let Θ also denote the corresponding set of predicate liftings $\{\widehat{\alpha} \mid \alpha \in \Theta\}$. ◁

That this approach makes sense follows by the following Proposition which states that the coalgebraic and the first-order perspective coincide.

Proposition 3.9 *Let $\alpha(a_1, \dots, a_n) \in \Phi^=(A)$. For any set S, any valuation $V : A \to PS$ and any subset $D \subseteq S$ we have*

$$PS, V, D \Vdash^1 \heartsuit_{\widehat{\alpha}}(a_1, \dots, a_n) \quad \textit{iff} \quad \langle D, V \rangle \models \alpha(a_1, \dots, a_n). \tag{12}$$

Proof. The proof of Proposition 3.9 consists of a four line unravelling of the definitions. Consider a first-order sentence $\alpha(a_1, \dots, a_n) \in \Phi^=$. Then

$$\begin{array}{llll} PS, V, D \Vdash^1 \heartsuit_{\widehat{\alpha}}(a_1, \dots, a_n) & \text{iff} & D \in \widehat{a}(V(a_1), \dots, V(a_n)) & \text{(semantics } \heartsuit) \\ & \text{iff} & \langle D, V_{V(a_1),\dots,V(a_n)} \rangle \models \alpha & \text{(definition } \widehat{\alpha}) \\ & \text{iff} & \langle D, V \rangle \models \alpha & (\dagger) \end{array}$$

where the last equivalence (†) follows from the fact that the predicate symbols in α are among $a_1, \dots, a_n$. QED

In section 4 we will see that all predicate liftings in Φ and Π are *natural* for P, while this is definitely not the case for $\Phi^=$ and Π (cf. Theorem 4.9).

As we will see further on, the following theorem is the reason why our coalgebraic approach can be applied to the Janin-Walukiewicz theorem. It states that, on the class of tree models, monadic second order logic can be captured by automata-theoretic means. As a corollary of this result and Proposition 3.5, MSO (see Example 2.9) is in fact a coalgebraic fixpoint logic. Theorem 3.10 below can be seen as a more precise formulation of Fact 1.3; as mentioned there, the two statements can be found in Walukiewicz [23] and Janin & Walukiewicz [12], respectively.

Theorem 3.10 *1. $MSO_v(\mathsf{Q}) \equiv_P \mu L_{\Pi^=}(\mathsf{Q})$ on tree models*[1]*;*
2. $\mu ML(\mathsf{Q}) \equiv_P \mu L_{\Pi}(\mathsf{Q})$.

Unfortunately, a proof of this Theorem would go beyond the scope of this paper. For proof details, the reader is referred to the above-mentioned papers, or to Chapter 16 of [11].

Remark 3.11 For the interested reader, we give a very rough sketch of the proof for part 1, which consists of two parts. First, rather than working with $MSO(\mathsf{Q})$ one defines a variant MSO' with only second-order variables. For a definition of the set of MSO'-formulas, consider the following grammar:

$$\varphi \ ::= \ p \sqsubseteq q \mid p \lhd q \mid \Downarrow p \mid \neg\varphi \mid \varphi \vee \psi \mid \exists p.\varphi$$

[1] Here the tacit understanding is that the variable v is interpreted as the *root* of the tree.

where p,q belong to some set $\mathsf{Q}' \supseteq \mathsf{Q}$ of variables. Then we define $MSO'(\mathsf{Q})$ as the set of MSO'-formulas whose free variables belong to Q.

Intuitively, it may be useful to think of MSO' as a *first-order* logic in which the variables are interpreted on the power set of (the state space of) a Kripke frame. The valuation of a Kripke model is then to be seen as a first-order assignment of an element $V(p) \in PS$ to an arbitrary letter $p \in \mathsf{Q}$. More precisely, the semantics of this language on a pointed Kripke model $(\mathbb{S}, r)$ is defined inductively — we only give the clause of the atomic formulas:

$$
\begin{array}{lll}
(\mathbb{S},r) \models p \sqsubseteq q & \text{iff} & V(p) \subseteq V(q) \\
(\mathbb{S},r) \models p \lhd q & \text{iff} & \text{for all } s \in V(p) \text{ there is a } t \in V(q) \text{ with } Rst. \\
(\mathbb{S},r) \models \Downarrow p & \text{iff} & V(p) = \{r\}
\end{array}
$$

It is not too difficult to see why this language corresponds to standard MSO. To start with, it is easy to interpret MSO'-formulas in standard MSO; for a translation in the opposite direction, the key idea is to encode elements of S as the corresponding singleton sets, and define a formula $\mathrm{sing}(p) \in MSO'$ characterizing the singleton subsets of S in the sense that $\mathbb{S}, r \models \mathrm{sing}(p)$ iff $V(p)$ is a singleton.

In the second part of the proof of Theorem 3.10(1) one defines, by induction on the complexity of a formula $\varphi \in MSO'(\mathsf{Q}')$, an automaton $\mathbb{A}_\varphi \in \mathrm{Aut}_{\Pi^=,\mathsf{Q}'}$ which is equivalent to φ in the sense that for any tree model $\mathbb{T}$ with root r, we have $(\mathbb{T}, r) \models \varphi$ iff $\mathbb{A}_\varphi$ accepts $(\mathbb{T}, r)$. This part of the proof is nontrivial, involving closure properties of specific classes of automata $\langle A, \delta, \Omega \rangle$ that are defined by restricting the range of the transition map δ to fragments of $\Phi^=$ (such as, in particular, the set $N^{=,+}$ defined in the next section).

4 One-step adequacy

4.1 The general case

In this section we define and compare various one-step versions of the notion of adequacy, and of the notion of one logic corresponding the one-step behaviorally invariant fragment of another. First we need a notion of one-step equivalence; to understand this notion, consider an A-valuation $V : A \to PS$. Lifting the associated coloring $V^\flat : S \to PA$, we obtain a map $TV^\flat : TS \to TPA$, which associates, with an element $\tau \in TS$ an object $TV^\flat(\tau)$ that one may think of as a 'T-color'.

Definition 4.1 Given two A-valuations $V_i : A \to PS_i$ $(i = 0,1)$, we define a relation $\sim_{V_0,V_1} \subseteq TS_0 \times TS_1$ by putting $\tau_0 \sim_{V_0,V_1} \tau_1$ iff τ_0 and τ_1 have the same T-color, that is,

$$\tau_0 \sim_{V_0,V_1} \tau_1 \quad \text{iff} \quad (TV_0^\flat)\tau_0 = (TV_1^\flat)\tau_1.$$

We call a rank-1 formula φ *one-step T-invariant* if for all pairs of valuations $V_i : A \to PS_i$, and all pairs of elements $\tau_i \in TS_i$ $(i = 0,1)$ such that $\tau_0 \sim_{V_0,V_1} \tau_1$ it holds that $TS_0, V_0, \tau_0 \Vdash^1 \varphi$ iff $TS_1, V_1, \tau_1 \Vdash^1 \varphi$. A coalgebraic logic is called *one-step behaviorally invariant* if each of its rank-1 formulas is one-step T-invariant. $\lhd$

In the sequel we will need a characterization of the notion of one-step adequacy that involves pairs of valuations on two sets that are linked by some function.

Definition 4.2 Fix two sets S, S' and a map $f : S' \to S$. Then with every valuation $V : A \to PS$ we may associate an A-valuation V_f on S' given by

$$V_f := \breve{P}f \circ V,$$

while for a valuation $U : A \to PS'$ defining

$$U^f := Pf \circ U.$$

we obtain an A-valuation U^f on S. $\lhd$

Concerning these definitions we need the following fact, which can be proved via routine verification.

Proposition 4.3 *Let $f : S' \to S$ be some map, and let $V : A \to PS$ and $U : A \to PS'$ be two valuations. Then*

1. $V = (V_f)^f$ *and* $U \subseteq (U^f)_f$ *(in the sense that* $U(a) \subseteq (U^f)_f(a)$ *for all* $a \in A$*).*
2. $U = V_f$ *iff* $U^\flat = V^\flat \circ f$.

The following proposition provides a useful characterization of one-step invariance.

Proposition 4.4 *Let Λ and A be sets of predicate liftings and proposition letters, respectively. A formula $\varphi \in L^1_\Lambda(A)$ is one-step T-invariant iff for each map $f : S' \to S$, for each $V : A \to PS$, and for each $\sigma' \in TS'$:*

$$TS, V, (Tf)\sigma' \Vdash^1 \varphi \quad \textit{iff} \quad TS', V_f, \sigma' \Vdash^1 \varphi. \tag{13}$$

Proof. For the direction from left to right, assume that $\varphi \in L^1_\Lambda$ is one-step T-invariant, and let f, V, and σ' be as in the formulation of the proposition. It follows from Proposition 4.3(2) that $V^\flat_f = V^\flat \circ f$, and from this it is immediate that $(Tf)\sigma' \sim_{V,V_f} \sigma'$. But then (13) follows from the one-step T-invariance of φ.

Conversely, suppose that φ satisfies the condition on the right hand side of the Proposition. Let $V_i : A \to PS_i$ $(i = 0,1)$ be two A-valuations, and let $\sigma_i \in TS_i$ be objects such that $\sigma_0 \sim_{V_0,V_1} \sigma_1$. Our aim is to prove that

$$TS_0, V_0, \sigma_0 \Vdash^1 \varphi \quad \text{iff} \quad TS_1, V_1, \sigma_1 \Vdash^1 \varphi. \tag{14}$$

For this purpose, consider the *natural* valuation $N : A \to PPA$ on PA given by $N(a) := \{B \in PA \mid a \in B\}$. We leave it as an exercise for the reader to verify that (i) $N^\flat = id_{PA}$, that (ii) $(\breve{P}V^\flat)N(a) = V(a)$ for all $a \in A$, and that (iii) $V_{V_i^\flat} = V_i$. From this we conclude that, taking $S = PA$, $S' = S_i$, $f = V_i^\flat$ and $\sigma_i = \sigma'$, we may read equation (13) as follows:

$$TPA, N, (TV_i^\flat)\sigma_i \Vdash^1 \varphi \quad \text{iff} \quad TS_i, V_i, \sigma_i \Vdash^1 \varphi.$$

From this, (14) is immediate by the assumption that $(TV_0^\flat)\sigma_0 = (TV_1^\flat)\sigma_1$. QED

The next theorem makes a link between some of the notions we have been discussing.

Theorem 4.5 *The following are equivalent, for any set Λ of predicate liftings:*

1. *Λ is natural;*
2. *Λ is one-step behaviorally invariant;*
3. *for each set A of proposition letters, for each function $f : S' \to S$, for each $V : A \to PS$, and for each $\sigma' \in TS'$,* (13) *holds for each formula $\varphi \in L_\Lambda(A)$.*

Proof. Since the equivalence of (2) and (3) is an immediate consequence of Proposition 4.4, it suffices to show that (1) $\iff$ (3).

For this purpose, first assume that (1) Λ is natural, and let A, S', S, and f be as in item 3. Clearly it suffices to prove (13) for an arbitrary atomic rank-1 formula $\heartsuit_\lambda(\overline{a})$:

$$\begin{array}{llll} TS, V, (Tf)\sigma' \Vdash^1 \heartsuit_\lambda(\overline{a}) & \text{iff} & (Tf)\sigma' \in \lambda_S(V(a_1), \ldots, V(a_n)) & \text{(semantics } \heartsuit_\lambda) \\ & \text{iff} & \sigma' \in (\breve{P}Tf)\lambda_S(V(a_1), \ldots, V(a_n)) & \text{(definition } \breve{P}Tf) \\ & \text{iff} & \sigma' \in \lambda_{S'}\big((\breve{P}f)V(a_1), \ldots, (\breve{P}f)V(a_n)\big) & \text{(naturality of } \lambda) \\ & \text{iff} & \sigma' \in \lambda_{S'}\big(V_f(a_1), \ldots, V_f(a_n)\big) & \text{(definition } V_f) \\ & \text{iff} & TS', V_f, \sigma' \Vdash^1 \heartsuit_\lambda(\overline{a}) & \text{(semantics } \heartsuit_\lambda) \end{array}$$

Conversely, assume (3) and consider an arbitrary n-ary predicate lifting $\lambda \in \Lambda$. In order to prove that λ is natural, take an arbitrary function $f : S' \to S$, and an arbitrary n-tuple $\overline{X} = (X_1, \ldots, X_n)$ of subsets of S. We need to show that

$$\lambda_{S'}\big((\breve{P}f)X_1, \ldots, (\breve{P}f)X_n\big) = (\breve{P}Tf)\lambda_S(X_1, \ldots, X_n). \tag{15}$$

For this purpose, define $A := \{a_1, \ldots, a_n\}$, and consider the valuation $V : A \to PS$ such that $V(a_i) = X_i$; observe that by definition of V_f, this implies that $V_f(a_i) = (\breve{P}f)X_i$. Hence it suffices to prove, for an arbitrary element $\sigma' \in S'$, that

$$\sigma' \in \lambda_{S'}\big(V_f(a_1), \ldots, V_f(a_n)\big) \quad \text{iff} \quad \sigma' \in (\breve{P}Tf)\lambda_S(V(a_1), \ldots, V(a_n)).$$

This we prove as follows:

$$
\begin{array}{llll}
\sigma' \in \lambda_{S'}\big(V_f(a_1),\ldots,V_f(a_n)\big) & \text{iff} & TS', V_f, \sigma' \Vdash^1 \heartsuit_\lambda(\overline{a}) & \text{(semantics } \heartsuit_\lambda) \\
 & \text{iff} & TS, V, (Tf)\sigma' \Vdash^1 \heartsuit_\lambda(\overline{a}) & \text{(assumption)} \\
 & \text{iff} & (Tf)\sigma' \in \lambda_S(V(a_1),\ldots,V(a_n)) & \text{(semantics } \heartsuit_\lambda) \\
 & \text{iff} & \sigma' \in (\check{P}Tf)\lambda_S(V(a_1),\ldots,V(a_n)) & \text{(definition } \check{P}Tf)
\end{array}
$$

QED

We now turn to the one-step version of one coalgebraic logic corresponding to the T-invariant fragment of another. On the basis of Theorem 4.5, we may nicely formulate this property in terms of predicate liftings.

Definition 4.6 Let Λ and Λ' be two sets of predicate liftings for the functor T. We say that Λ' *corresponds to the behaviorally invariant fragment of* Λ *at the one-step level*, notation: $\Lambda' \equiv_1 \Lambda/T$, if

1. $\Lambda' \subseteq \Lambda$,
2. Λ' is natural, and
3. every one-step T-invariant formula $\varphi \in L^1_\Lambda(A)$ is equivalent to a formula φ^* in $L^1_{\Lambda'}(A)$.

If Λ and Λ' satisfy the conditions 1, 2 and 3′ below:

3′. there is a translation $(\cdot)^* : L^1_\Lambda(A) \to L^1_{\Lambda'}(A)$ and a construction associating, with each set S, a set S' together with a map $f : S' \to S$, such that for each $\sigma \in TS$ there is a $\sigma' \in TS'$ with $\sigma = (Tf)\sigma'$, and for each valuation $V : A \to PS$, and each formula $\varphi \in L^1_\Lambda(A)$ it holds that

$$TS, V, \sigma \Vdash^1 \varphi^* \quad \text{iff} \quad TS', V_f, \sigma' \Vdash^1 \varphi. \tag{16}$$

we say that Λ' *uniformly corresponds to the behaviorally invariant fragment of* Λ *at the one-step level*, notation: $\Lambda' \equiv^u_1 \Lambda/T$. ◁

Similarly to Proposition 2.14, the following Proposition states that uniform correspondence implies correspondence.

Proposition 4.7 *If* $\Lambda' \equiv^u_1 \Lambda/T$, *then* $\Lambda' \equiv_1 \Lambda/T$.

Proof. Similarly to the proof of Proposition 2.14, one may show that if $\varphi \in L^1_\Lambda$ is one-step T-invariant, then φ is equivalent to φ^*, where $(\cdot)^* : L^1_\Lambda(A) \to L^1_{\Lambda'}(A)$ is the translation given by uniform correspondence. QED

Remark 4.8 There are many variants of the notion of uniform one-step correspondence that may be of interest as well. For instance, it makes sense to weaken the condition (1), stating that Λ' is an actual *subset* of Λ, to a condition requiring that there is a translation $(\cdot)^\dagger$ mapping any formula $\varphi \in L^1_{\Lambda'}(A)$ to an equivalent formula $\varphi^\dagger \in L^1_\Lambda(A)$. As a second example, all of the results in this paper still hold if we weaken condition 3′ to a non-uniform version in which the set S' and the function $f : S' \to S$ depend on the object $\sigma \in TS$. In detail, this condition would read as follows

3″. there is a translation $(\cdot)^* : L^1_\Lambda(A) \to L^1_{\Lambda'}(A)$ and a construction associating, with each set S, and each $\sigma \in TS$, a set S', a map $f : S' \to S$, and an object $\sigma' \in TS'$ such that $\sigma = (Tf)\sigma'$, and for each valuation $V : A \to PS$, (16) holds for each formula $\varphi \in L^1_\Lambda(A)$.

Finally, in some cases it may be convenient to consider (one-step) languages in which we admit only a (not necessarily functionally complete) selection of Boolean connectives.

4.2 The case of Kripke models

Our key example of the notions just defined is given by the predicate liftings for Kripke structures, that are induced by monadic first-order sentences. Our main goal in this section will be to prove the following Theorem, which states that Π uniformly corresponds to the behaviorally invariant fragment of $\Pi^=$. This theorem will be crucial in our proof of the Janin-Walukiewicz theorem.

Theorem 4.9 $\Pi \equiv^u_1 \Pi^=/P$.

Let us first see what one-step invariance means in this context. Recall from the introduction the notions of P-equivalence of structures, and of P-invariance of monadic sentences. We will say that a monadic first-order sentence $\alpha \in \Phi^=$ is *invariant under surjective homomorphisms* iff for each valuation $V : A \to PD$ and each surjection $f : D' \to D$,

$$\langle D, V\rangle \models \alpha \quad \text{iff} \quad \langle D', V_f\rangle \models \alpha.$$

Proposition 4.10 *For any first-order sentence $\alpha(\overline{a}) \in \Phi^=$, the following are equivalent:*

1. *the predicate lifting $\widehat{a}$ is natural;*
2. *the rank-1 formula $\heartsuit_{\widehat{a}}(\overline{a})$ is P-invariant;*
3. *α is P-invariant;*
4. *α is invariant under surjective homomorphisms.*

Proof. This Proposition is a straightforward consequence of Proposition 3.9 and Proposition 4.4. QED

As a first corollary of this, we obtain the following.

Corollary 1. *Φ and Π are P-invariant sets of predicate liftings.*

Proof. By Proposition 4.10 it suffices to show that all identity-free sentences of monadic first-order logic are invariant under surjective homomorphisms. This is a routine exercise in first-order logic. QED

As a more important consequence of Proposition 4.10, we may prove Theorem 4.9 as a corollary of the following result on monadic first-order logic.

Proposition 4.11 *There is a translation $(\cdot)^* : \Pi^=(A) \to \Pi(A)$ such that for all structures (D,V) and all sentences $\alpha \in \Pi^=$ we have*

$$(D,V) \models \alpha^* \textit{ iff } (D \times \omega, V_\pi) \models \alpha, \tag{17}$$

where $\pi : D \times \omega \to D$ is the first projection function, $\pi : (d,n) \mapsto d$.

In order to prove Proposition 4.11, we will need certain normal forms for monadic first-order sentences. First we supply some preliminary definitions.

Definition 4.12 For a sequence $\overline{x} = x_1, \ldots, x_n$ of variables, write $\mathrm{diff}(\overline{x}) := \bigwedge_{1<j} x_i \neq x_j$. Given a set $B \subseteq A$ and a variable x, abbreviate $\tau_B(x) := \bigwedge_{a \in B} a(x) \wedge \bigwedge_{a \notin B} \neg a(x)$ and $\tau_B^+(x) := \bigwedge_{a \in B} a(x)$. ◁

In words, $\mathrm{diff}(\overline{x})$ states that the variables $x_1, \ldots x_n$ denote *distinct* elements of the domain. The formulas $\tau_B(x)$ and $\tau_B^+(x)$ state, respectively, that the *type* of the element denoted by x is equal to (contains, respectively) B. Here the *type* of an element d in a structure $\langle D,V \rangle$ for A is the set $V^\flat(d)$.

Definition 4.13 Fix a set A of propositional variables. Let $\overline{B} = B_1, \ldots, B_n$ and $\overline{C} = C_1, \ldots, C_m$ be two sequences of subsets of A, respectively. We define the following formulas:

$$\begin{aligned}
\chi^{=,+}(\overline{B},\overline{C}) &:= \exists y_1 \cdots y_n \Big(\mathrm{diff}(\overline{y}) \wedge \textstyle\bigwedge_i \tau_{B_i}^+(y_i) \wedge \forall z \big(\mathrm{diff}(\overline{y}z) \to \bigvee_j \tau_{C_j}^+(z)\big)\Big) \\
\chi^{=}(\overline{B},\overline{C}) &:= \exists y_1 \cdots y_n \Big(\mathrm{diff}(\overline{y}) \wedge \textstyle\bigwedge_i \tau_{B_i}(y_i) \wedge \forall z \big(\mathrm{diff}(\overline{y}z) \to \bigvee_j \tau_{C_j}(z)\big)\Big) \\
\chi^{+}(\overline{B},\overline{C}) &:= \exists y_1 \cdots y_n \Big(\textstyle\bigwedge_i \tau_{B_i}^+(y_i) \wedge \forall z \bigvee_j \tau_{C_j}^+(z)\Big) \\
\chi(\overline{B},\overline{C}) &:= \exists y_1 \cdots y_n \Big(\textstyle\bigwedge_i \tau_{B_i}(y_i) \wedge \forall z \bigvee_j \tau_{C_j}(z)\Big)
\end{aligned}$$

We let $N^{=,+}(A)$ denote the set of sentences of the form $\chi^{=,+}(\overline{B},\overline{C})$, and proceed similarly for the sets $N^=(A)$, $N^+(A)$ and $N(A)$. ◁

We need the following fact from first-order logic, which explains why we may think of (disjunctions of) χ-type formulas as providing *normal forms* for monadic first-order logic.

Proposition 4.14 *Every sentence in $\Phi^=$ is equivalent to a disjunction of sentences in $N^=$, and similarly for Φ and N, $\Pi^=$ and $N^{=,+}$ and Π and N^+, respectively.*

Proof. The proof of this Proposition can be seen as an exercise in the theory of Ehrenfeucht-Fraïssé games. We confine ourselves to a sketch (rephrasing the proof of Lemma 16.23 in [11]), and we only consider the case of $\Phi^=$.

Given a set $B \subseteq A$, and a first-order structure $\mathbb{D} = \langle D,V \rangle$ for A, let $N_{B,\mathbb{D}}$ be the number of elements in D of type B. We say that two such structures $\mathbb{D}$ and $\mathbb{D}'$ are n-equivalent, notation $\mathbb{D} \sim_n \mathbb{D}'$, if for every $B \subseteq A$, either $N_{B,\mathbb{D}} = N_{B,\mathbb{D}'} \leq n$, or both $N_{B,\mathbb{D}} > n$ and $N_{B,\mathbb{D}'} > n$. Clearly $\sim_n$ is an equivalence relation of finite index, and each equivalence class of $\sim_n$ is described by a formula in $N^=$. Using Ehrenfeucht-Fraïssé games it is not difficult to show that $\mathbb{D} \sim_n \mathbb{D}'$ implies that $\mathbb{D}$ and $\mathbb{D}'$ satisfy the same sentences of quantifier rank at most n. From this it follows that the class of models of such a sentence is the union of a (finite) number of $\sim_n$-cells, and that the sentence itself is thus equivalent to the disjunction of the formulas asssociated with these $\sim_n$-cells. QED

Now we are ready for the proof of Proposition 4.11.

Proof of Proposition 4.11. Given a formula $\alpha \in \Pi^{=}(A)$, we need to come up with a translation $\alpha^* \in \Pi(A)$ such that (17) holds.

First assume that α is of the form $\chi^{=,+}(\overline{B},\overline{C}) \in N^{=,+}$, and define

$$\alpha^* := \chi^{+}(\overline{B},\overline{C}).$$

Proving (17) in this situation boils down to showing that

$$\langle D,V\rangle \models \chi^{+}(\overline{B},\overline{C}) \quad \text{iff} \quad \langle D\times\omega, V_\pi\rangle \models \chi^{=,+}(\overline{B},\overline{C}). \tag{18}$$

For this purpose, first observe that V_π satisfies

$$d \in V(a) \quad \text{iff} \quad (d,n) \in V_\pi(a) \tag{19}$$

for each $d \in S$, $a \in A$, and $n \in \omega$. Suppose that $\overline{B} = B_1,\ldots,B_n$ and $\overline{C} = C_1,\ldots,C_m$.

For the left-to-right direction of (18), assume that $\langle D,V\rangle \models \chi^{+}(\overline{B},\overline{C})$. Let $d_1,\ldots,d_n$ be elements in D satisfying the existential part of $\chi^{+}(\overline{B},\overline{C})$, that is, for each i we find $d_i \in \bigcap_{b\in B_i} V(b)$. From the universal part of the formula it follows that for each $d \in D$ there is a subset $C_d \subseteq A$ such that $d \in \bigcap_{c\in C_d} V(c)$. Now we move to $D\times\omega$; it is easy to see that its elements $(d_1,1),\ldots,(d_n,n)$ provide a sequence of n distinct elements that satisfy $(d_i,i) \in \bigcap_{b\in B_i} V_\pi(b)$ for each i. In addition, every element (d,n) distinct from the ones in the mentioned tuple will satisfy $(d,n) \in \bigcap_{c\in C_d} V_\pi(c)$. From these observations it is immediate that $\langle D\times\omega, V_\pi\rangle \models \chi^{=,+}(\overline{B},\overline{C})$.

For the opposite direction of (18), assume that $\langle D\times\omega, V_\pi\rangle \models \chi^{=,+}(\overline{B},\overline{C})$. Let (d_1,k_1),..., (d_n,k_n) be the sequence of distinct elements of $D\times\omega$ witnessing the existential part of $\chi^{=,+}(\overline{B},\overline{C})$ in $\mathbb{D}'$. Then clearly, $d_1,\ldots,d_n$ witness the existential part of $\chi^{+}(\overline{B},\overline{C})$ in $\langle D,V\rangle$. In order to show that $\langle D,V\rangle$ also satisfies the universal part $\forall z \bigvee_j \tau^+_{C_j}(z)$ of χ^+, consider an arbitrary element $d \in D$. Take any $m \in \omega \setminus \{k_1,\ldots,k_n\}$, then (d,m) is distinct from each (d_i,k_i). It follows that for some j we have $(d,m) \in \bigcap_{c\in C_j} V_\pi(c)$, and so we obtain $d \in \bigcap_{c\in C_j} V(c)$. Since d was arbitrary this shows that indeed $\langle D,V\rangle \models \forall z \bigvee_j \tau^+_{C_j}(z)$. So we have proved that $\langle D,V\rangle \models \chi^{+}(\overline{B},\overline{C})$.

Now consider the general case, where α is arbitrary. It follows from Proposition 4.14 that α is equivalent to a formula $\alpha \equiv \bigvee_i \alpha_i$, with each formula α_i belongs to $N^{=,+}$. With

$$\alpha^* := \bigvee_i \alpha_i^*$$

it is straightforward to verify (17). QED

Both Proposition 1.4 and Theorem 4.9 are straightforward corollaries of Proposition 4.11.

Proof of Proposition 1.4. Assume that $\alpha \in \Pi^{=}$ is a P-invariant monadic sentence, and let $\alpha^* \in \Pi$ be the formula given by Proposition 4.11. Consider an arbitrary structure (D,V), and observe that $(D,V) \equiv_P (D \times \omega, V_\pi)$. But then we obtain the following equivalences:

$$\begin{aligned} (D,V) \models \alpha \quad &\text{iff} \quad (D \times \omega, V_\pi) \models \alpha && \text{(assumption on } \alpha) \\ &\text{iff} \quad (D,V) \models \alpha && (17) \end{aligned}$$

From this it is immediate that α and α^* are equivalent, which suffices to prove Proposition 1.4. QED

Proof of Theorem 4.9. It is obvious that $\Pi \subseteq \Pi^{=}$, and Proposition 1 states the one-step P-invariance of Π . Hence we may focus on item $3'$ of Definition 4.6.

We need to define a translation $(\cdot)^* : L^1_{\Pi^{=}} \to L^1_{\Pi}$. By the definition of rank-1 formulas, it suffices to come up with a translation for *atomic* rank-1 formulas, that is, formulas of the form $\heartsuit_{\widehat{\alpha}}(\overline{a})$ for some sentence $\alpha \in \Pi^{=}$. But for such a formula, we can simply put

$$\Big(\heartsuit_{\widehat{\alpha}}(\overline{a})\Big)^* := \heartsuit_{\alpha^*}(\overline{a}).$$

We leave it for the reader to verify that this defines a formula of the right shape and with the right properties. QED

5 Main result

We are now ready for the main technical result of the paper. Intuitively, Theorem 5.1 states that, given two sets Λ, Λ' of monotone predicate liftings for a functor T, if Λ corresponds to the T-invariant fragment of Λ' at the one-step level, then the coalgebraic μ-calculus $\mu L_{\Lambda,\mathsf{Q}}$ is the bisimulation-invariant fragment of $\mu L_{\Lambda',\mathsf{Q}}$.

Theorem 5.1 *Let T be some set functor, and let Λ, Λ' be two sets of monotone predicate liftings for T such that $\Lambda \equiv^u_1 \Lambda'/T$. Then for any set Q,*

$$\mathrm{Aut}_{\Lambda,\mathsf{Q}} \equiv^s \mathrm{Aut}_{\Lambda',\mathsf{Q}}/{\simeq_T}.$$

As a corollary, if Λ' is closed under Boolean duals, then $\mu L_{\Lambda,\mathsf{Q}} \equiv^s \mu L_{\Lambda',\mathsf{Q}}/{\simeq_T}$.

Proof. Fix a set Q of proposition letters, and assume that $\Lambda \equiv^u_1 \Lambda'/T$. It easily follows from this assumption that $\mathrm{Aut}_{\Lambda,\mathsf{Q}} \subseteq \mathrm{Aut}_{\Lambda',\mathsf{Q}}$ and that $\mathrm{Aut}_{\Lambda,\mathsf{Q}}$ is invariant under behavioral equivalence. This leaves the following tasks:

1. define a translation from initialized Λ,Q-automata to initialized Λ',Q-automata,
2. outline a construction, that assocates with an arbitrary pointed T-model $(\mathbb{S}, r)$, a pointed T-model $(\mathbb{S}', r')$ and a morphism $f : \mathbb{S}' \to \mathbb{S}$, and

3. prove, for every initialized Λ, Q-automaton $(\mathbb{A}, a_i)$, and every pointed model $(\mathbb{S}, r)$ that
$$\mathbb{S}, r \Vdash (\mathbb{A}, a_I)^* \quad \text{iff} \quad \mathbb{S}', r' \Vdash (\mathbb{A}, a_I). \tag{20}$$

The first of these tasks is easy to accomplish. Given a Λ, Q-automaton $\mathbb{A} = \langle A, \delta, \Omega \rangle$, recall that $\delta(a, \Pi)$ is a rank-1 Λ-formula for each $a \in A$ and $\Pi \in P\mathsf{Q}$. Hence we obtain a Λ', Q-automaton $\mathbb{A}^*$ by putting $\mathbb{A}^* := \langle A, \delta^*, \Omega \rangle$, where $\delta^* : A \times P\mathsf{Q} \to L^1_{\Lambda'}(A)$ is given by Definition 4.6.3′: $\delta^*(a, \Pi) := (\delta(a, \Pi))^*$. For the initialized automaton $(\mathbb{A}, a_I)$ we put $(\mathbb{A}, a_I)^* := (\mathbb{A}^*, a_I)$.

Concerning the second task, consider an arbitrary T-model $\mathbb{S} = \langle S, \sigma, W \rangle$. Take the set S' and the map $f : S' \to S$ provided by clause 3 of Definition 4.6. In order to endow the set S' with coalgebra structure, consider an arbitrary element $s' \in S'$. Applying the properties of S, S' and f (given by the mentioned clause) to the element $\sigma(fs') \in TS$, we obtain an element $\sigma' s' \in TS'$ such that

$$(Tf)(\sigma' s') = \sigma(fs') \tag{21}$$

and such that for every rank-1 formula $\varphi \in L^1_\Lambda(A)$ and every marking $V : A \to PS$ we have
$$TS, V, \sigma(fs') \Vdash^1 \varphi^* \quad \text{iff} \quad TS', V_f, \sigma' s' \Vdash \varphi. \tag{22}$$

Clearly this procedure defines a coalgebra structure $\sigma' : S' \to TS'$. For the valuation W' on $\mathbb{S}'$ we take $W' := W_f$. It is immediate by (21) and the fact that $W_f^\flat = W^\flat \circ f$, that the map $f : S' \to S$ is in fact a T_Q-coalgebra morphism.

Finally, we need to come up with a designated point r' of $\mathbb{S}' := \langle S', \sigma' \rangle$ which is mapped to r by f. Clearly if S' already contains such an element we are done; if not, then we can simply *adjoin* a fresh element r' to $\mathbb{S}'$. We define $\sigma' r'$ so that $(Tf)(\sigma' r') = \sigma r$ (this is possible by the assumptions), adapt the valuation W' so that the type of r' is that of r in $\mathbb{S}$, and add the pair (r', r) to (the graph of) f. Modulo some renaming, this ensures that we obtain a pointed coalgebra $(\mathbb{S}', r')$, with a map $f : S' \to S$ satisfying (21) and (22), and such that $W' = W_f$ and $fr' = r$. (Formally, we define a model $\mathbb{S}''$ based on the set $S'' := S' \uplus \{r'\}$, and in the sequel work with the pointed model $(\mathbb{S}'', r')$. We omit the details of this construction which are coalgebraically obvious but somewhat tedious).

We are now ready to prove (20). Fix an initialized Λ, Q-automaton $(\mathbb{A}, a_I)$ and a pointed T-model $(\mathbb{S}, r)$. Clearly it suffices to show that

$$(a_I, r) \in \mathrm{Win}_\exists(\mathscr{A}(\mathbb{A}^*, \mathbb{S})) \quad \text{iff} \quad (a_I, r') \in \mathrm{Win}_\exists(\mathscr{A}(\mathbb{A}, \mathbb{S}')). \tag{23}$$

Abbreviate $\mathscr{A}^* = \mathscr{A}(\mathbb{A}^*, \mathbb{S})$ and $\mathscr{A}' = \mathscr{A}(\mathbb{A}, \mathbb{S}')$.

For the direction from left to right of (23), without loss of generality we may assume that in $\mathscr{A}^*$, $\exists$ has a *positional* strategy $\theta : A \times S \to (PS)^A$ which is winning when played from each position $(a, s) \in \mathrm{Win}_\exists(\mathscr{A}^*)$. Now consider the following (positional) strategy θ_f for $\exists$ in $\mathscr{A}'$:

at position (a,s') $\exists$ picks the marking V_f,
where $V : A \to PS$ is the marking $V = \theta(a, fs')$ provided in $\mathscr{A}'$ by θ at (a, fs').

The legitimacy of this move is immediate by (22).

In order to show that θ_f is in fact a winning strategy in $\mathscr{A}'@(a_I, r)$, consider an arbitary match

$$\pi = (a_I, r')U_1(a_1, s'_1)U_2(a_2, s'_2)\dots$$

in which $\exists$ plays the strategy θ_f just defined. The point is that there is an associated θ-conform $\mathscr{A}^*$-match

$$\pi^* = (a_I, r)V_1(a_1, s_1)V_2(a_2, s_2)\dots$$

such that $U_i = (V_i)_f$ and $fs'_i = s_i$ for all $i < \omega$. To see this, consider a round of the game, starting at position $(a,s') \in A \times S'$ with $(a, fs') \in \mathrm{Win}_\exists(\mathscr{A}^*)$. If $\exists$ plays her strategy θ_f, picking the marking V_f with $V = \theta(a, fs')$, then for every pair (b,t') picked by $\forall$, by definition of V_f, the pair (b, ft') is a legitimate move for $\forall$ in $\mathscr{A}^*$.

By our assumptions on (a_I, r) and θ, the match π^* is won by $\exists$. But since π and π^* project to exactly the same sequence of A-states, and the winning conditions of $\mathscr{A}^*$ and $\mathscr{A}'$ are the same, this means that $\exists$ also wins π. Thus we conclude that θ_f is a winning strategy for $\exists$ in the game $\mathscr{A}'$ initialized at (a_I, r').

For the opposite direction '$\Leftarrow$' of (23), we may assume that in $\mathscr{A}^*$, $\exists$ has a *positional* strategy $\eta : A \times S \to (PS)^A$ which is winning from all positions $(a,s') \in \mathrm{Win}_\exists(\mathscr{A}')$. We will use this η to define a (partial) strategy η^f for $\exists$ in $\mathscr{A}^*$.

For the definition of η^f, consider a position in $\mathscr{A}^*$ of the form (a, fs') for some $s' \in S'$ such that $(a,s') \in \mathrm{Win}_\exists(\mathscr{A}')$. (Note that the position (a_I, r) has this shape.) Suppose that in $\mathscr{A}'$, at position (a,s'), $\exists$'s strategy η tells her to pick a marking $U : A \to PS'$. Our first claim is that the marking $U^f : A \to PS$ constitutes a legitimate move for $\forall$ in the game $\mathscr{A}^*$ at position (a, fs').

To see this we need to verify that $TS, U^f, \sigma(fs') \Vdash^1 \delta^*(a, W^\flat(fs'))$. But because U is a legitimate move at (a,s') in $\mathscr{A}'$, we know that $TS', U, \sigma's' \Vdash^1 \delta(a, W^\flat_f(s'))$. Observe that $U \subseteq (U^f)_f$ (Proposition 4.3), so that by monotonicity it follows that $TS, (U^f)_f, \sigma's' \Vdash^1 \delta(a, W^\flat_f(s'))$. From this it is immediate by (22) and the fact that $W^\flat(fs') = W^\flat_f(s')$ (Proposition 4.3), that $TS, U^f, \sigma(fs') \Vdash^1 \delta^*(a, W^\flat(fs'))$, as required. Now consider an arbitrary response (b,t) of $\forall$ to $\exists$'s move U^f at position (a, fs'). It follows from $t \in U^f(b)$ that t is of the form ft' for some $t' \in S'$ such that $t = ft'$. This means that in $\mathscr{A}'$, the move (b,t') is legitimate at position U. Furthermore, since we assumed that U was given by a winning strategy, the position (b,t') belongs to the set $\mathrm{Win}_\exists(\mathscr{A}')$. Summarizing, this shows that in any round of $\mathscr{A}^*$ starting at a position (a, fs') with $(a,s') \in \mathrm{Win}_\exists(\mathscr{A}')$, $\exists$ has the power to end the round at a position (b,t) of the same kind; and more specifically, she maintains an η-conform 'shadow round' of the game $\mathscr{A}'$ starting at (a,s') and ending at a position $(b,t') \in \mathrm{Win}_\exists(\mathscr{A}')$ with $ft' = t$.

On the basis of the above observations, we may easily equip $\exists$ with a (partial) strategy η^f with the property, that for any η^f-conform match

$$\pi = (a_I, r)V_1(a_1, s_1)V_2(a_2, s_2)\cdots$$

there is an η-conform 'shadow match'

$$\pi' = (a_I, r')U_1(a_1, s_1')U_2(a_2, s_2')\cdots$$

such that $s_i = fs_i'$ and $V_i = U_i^f$ for all $i \in \omega$. From this we may derive, using a similar argument as given before, that π is won by $\exists$. Thus in this case we conclude that η^f is a winning strategy for $\exists$ in the game $\mathscr{A}^*$ initialized at (a_I, r). QED

Finally we show how to derive the Janin-Walukiewicz theorem, stating that μML is the bisimulation-invariant fragment of *MSO*, from the results obtained (or mentioned) above.

Corollary 2 (Janin & Walukiewicz). *For any set* Q *of proposition letters,* $\mu ML(\mathsf{Q}) \equiv^s MSO_v(\mathsf{Q})/P$.

Proof. This result is a straightforward corollary of the Theorems 3.10, 4.9 and 5.1, together with Proposition 3.5.

To see this, fix a set Q of proposition letters, and note that by Theorem 3.10(1), there is an initialized automaton $(\mathbb{A}_\varphi, a_\varphi)$ in $\mathrm{Aut}_{\Pi^=}(\mathsf{Q})$ such that

$$\varphi \equiv (\mathbb{A}_\varphi, a_\varphi) \text{ on trees.} \tag{24}$$

By the Theorems 3.10(2), 4.9 and 5.1 and by Proposition 3.5, there is a translation $\xi : \mathrm{Aut}_{\Pi^=}(\mathsf{Q}) \to \mu ML$ such that for all pointed Kripke models $(\mathbb{S}, s)$, there is a pointed Kripke model $(\mathbb{S}', s')$ and a morphism $f : (\mathbb{S}', s') \to (\mathbb{S}, s)$ such that for all initialized automata $(\mathbb{A}, a)$ it holds that

$$\mathbb{S}, s \Vdash \xi(\mathbb{A}, a) \quad \text{iff} \quad \mathbb{S}', s' \Vdash (\mathbb{A}, a). \tag{25}$$

Now let $\varphi(v) \in MSO_v(\mathsf{Q})$ be invariant under bisimilarity, or behavioral equivalence (these are the same for the power set functor P). We claim that $\varphi' := \xi(\mathbb{A}_\varphi, a_\varphi) \in \mu ML$ is equivalent to φ. To see this, let $(\mathbb{S}_0, s_0)$ be an arbitrary pointed Kripke model, and let $(\mathbb{S}_1, s_1)$ be a tree model bisimilar (or behaviorally equivalent) to $(\mathbb{S}_0, s_0)$.

Then we have the following chain of equivalences:

$$\begin{array}{rlll}
\mathbb{S}_0, s_0 \Vdash \varphi & \text{iff} & \mathbb{S}_1, s_1 \Vdash \varphi & \text{(assumption on } \varphi) \\
& \text{iff} & \mathbb{S}_1', s_1' \Vdash \varphi & \text{(assumption on } \varphi) \\
& \text{iff} & \mathbb{S}_1', s_1' \Vdash (\mathbb{A}_\varphi, a_\varphi) & (24) \\
& \text{iff} & \mathbb{S}_1, s_1 \Vdash \varphi' & (25) \\
& \text{iff} & \mathbb{S}_0, s_0 \Vdash \varphi' & \text{(adequacy of } \mu ML)
\end{array}$$

which shows that $\varphi \equiv \varphi' \in \mu ML$ indeed. QED

6 Conclusion

We finish the paper with some general observations and questions for further research.

First of all, given the fact that it is an open problem whether the Janin-Walukiewicz theorem also holds in the setting of finite models, it may be interesting to note that both Proposition 1.4 and Theorem 1.5 can be proved in that setting, as can Fact 1.3(2). Hence, the 'only' hurdle to prove a finite model theory version of their result is the fact that the correspondence between monadic second-order logic and $\Pi^{=}$-automata is only proven for tree models (Fact 1.3(1)).

Second, commenting on an earlier version of this chapter, van Benthem asked some questions concerning the translation $(\cdot)^*$ from *MSO* to μML. His question concerning interpolation can be answered positively: given two *MSO*-formulas φ and χ, one may show that φ implies χ 'along bisimilarity' iff φ and χ have an interpolant ψ in the modal μ-calculus.

The analysis of fixed-point logics at the level of syntax for the transition functions of automata, which started with the work of Janin & Walukiewicz, has yielded some other basic results about the modal μ-calculus. For instance, it was used by d'Agostino & Hollenberg to prove uniform interpolation [6] and by Fontaine & Venema to obtain various preservation results, such as the characterization of the continuous fragment of μML [9].

Finally, it would be interesting to extend the coalgebraic analysis of the modal μ-calculus from model-theoretic aspects to axiomatics and proof theory.

Acknowledgements The research of this author has been made possible by VICI grant 639.073.501 of the Netherlands Organization for Scientific Research (NWO). Furthermore, the author is grateful to Facundo Carreiro for many comments on an earlier version of the paper.

References

1. H. Andréka, J. van Benthem, and I. Németi. Modal languages and bounded fragments of predicate logic. *Journal of Philosophical Logic*, 27:217–274, 1998.
2. J. van Benthem. *Modal Correspondence Theory*. PhD thesis, Mathematisch Instituut & Instituut voor Grondslagenonderzoek, University of Amsterdam, 1976.
3. J. van Benthem. Correspondence theory. In D.M. Gabbay and F. Guenthner, editors, *Handbook of Philosophical Logic*, volume 2, pages 167–247. Reidel, 1984.
4. C. Cîrstea, C. Kupke, and D. Pattinson. EXPTIME tableaux for the coalgebraic μ-calculus. In E. Grädel and R. Kahle, editors, *Computer Science Logic 2009*, volume 5771 of *Lecture Notes in Computer Science*, pages 179–193. Springer, 2009.
5. C. Cîrstea, A. Kurz, D. Pattinson, L. Schröder, and Y. Venema. Modal logics are coalgebraic. *The Computer Journal*, 54:524–538, 2011.
6. G. D'Agostino and M. Hollenberg. Logical questions concerning the μ-calculus. *Journal of Symbolic Logic*, 65:310–332, 2000.
7. A. Dawar and M. Otto. Modal characterisation theorems over special classes of frames. *Annals of Pure and Applied Logic*, 161:1–42, 2009.

8. G. Fontaine, R. Leal, and Y. Venema. Automata for coalgebras: An approach using predicate liftings. In S. Abramsky et alii, editor, *Proceedings of ICALP 2010, part II*, volume 6199 of *LNCS*, pages 381–392. Springer, 2010.
9. G. Fontaine and Y. Venema. Some model theory for the modal mu-calculus: syntactic characterizations of semantic properties. Manuscript, submitted, 2010.
10. E. Grädel, C. Hirsch, and M. Otto. Back and forth between guarded and modal logics. *ACM Transactions on Computational Logic*, 3:418–463, 2002.
11. E. Grädel, W. Thomas, and T. Wilke, editors. *Automata, Logic, and Infinite Games*, volume 2500 of *LNCS*. Springer, 2002.
12. D. Janin and I. Walukiewicz. Automata for the modal μ-calculus and related results. In *Proceedings of the Twentieth International Symposium on Mathematical Foundations of Computer Science, MFCS'95*, volume 969 of *LNCS*, pages 552–562. Springer, 1995.
13. D. Janin and I. Walukiewicz. On the expressive completeness of the propositional μ-calculus w.r.t. monadic second-order logic. In *Proceedings CONCUR '96*, 1996.
14. T. Litak, D. Pattinson, K. Sano, and L. Schröder. Coalgebraic predicate logic. In A. Czumaj et alii, editor, *ICALP 2012*, volume 7392 of *Lecture Notes in Computer Science*, pages 299–311. Springer, 2012.
15. L. Moss. Coalgebraic logic. *Annals of Pure and Applied Logic*, 96:277–317, 1999. (Erratum published *Ann.P.Appl.Log.* 99:241–259, 1999).
16. M. Otto. Model theoretic methods for fragments of fo and special classes of (finite) structures. In J. Esparza, C. Michaux, and C. Steinhorn, editors, *Finite and Algorithmic Model Theory*, volume 379 of *LMS Lecture Notes Series*. Cambridge University Press, 2011.
17. D. Pattinson. Coalgebraic modal logic: soundness, completeness and decidability of local consequence. *Theoretical Computer Science*, 309(1–3):177–193, 2003.
18. E. Rosen. Modal logic over finite structures. *Journal of Logic, Language and Information*, 6:427–439, 1997.
19. J. Rutten. Universal coalgebra: a theory of systems. *Theoretical Computer Science*, 249:3–80, 2000.
20. B. ten Cate, D. Gabelaia, and D. Sustretov. Modal languages for topology: Expressivity and definability. *Annals of Pure and Applied Logic*, 159(1-2):146–170, 2009.
21. Y. Venema. Algebras and coalgebras. In J. van Benthem, P. Blackburn, and F. Wolter, editors, *Handbook of Modal Logic*, pages 331–426. Elsevier, 2006.
22. Y. Venema. Automata and fixed point logic: a coalgebraic perspective. *Information and Computation*, 204:637–678, 2006.
23. I. Walukiewicz. Monadic second-order logic on tree-like structures. *Theoretical Computer Science*, 275:311–346, 2002.