Available online at www.sciencedirect.com
SCIENCE DIRECT®
Electronic Notes in Theoretical Computer Science
ELSEVIER Electronic Notes in Theoretical Computer Science 106 (2004) 355–375
www.elsevier.com/locate/entcs

Automata and Fixed Point Logics for Coalgebras

Yde Venema[1]

Institute for Logic, Language and Computation
University of Amsterdam
Plantage Muidergracht 24, NL–1018 TV Amsterdam.

Abstract

It is the aim of this paper to generalize existing connections between automata and logic to a more general, coalgebraic level.
Let $\mathsf{F} : \mathsf{Set} \to \mathsf{Set}$ be a standard functor that preserves weak pullbacks. We introduce the notion of an F-automaton, a device that operates on pointed F-coalgebras; the criterion under which such an automaton accepts or rejects a pointed coalgebra is formulated in terms of an infinite two-player graph game.
We also introduce a language of coalgebraic fixed point logic for F-coalgebras, and we provide a game semantics for this language. Finally we show that any formula p of the language can be transformed into an F-automaton $\mathbb{A}_p$ which is equivalent to p in the sense that $\mathbb{A}_p$ accepts precisely those pointed F-coalgebras in which p holds.

Keywords: coalgebra, automata, modal logic, fixed point operators, game semantics, bisimulation, parity games

1 Introduction

There is a long and respectable tradition in theoretical computer science linking the research fields of automata theory and logic. This link becomes particularly strong when automata are used to classify *infinite* objects like words, trees or graphs. Interestingly, this research area has provided not only fundamental theoretical results, such as Rabin's decidability theorem [13], but also quite concrete applications in computer science, such as tools for the automatic verification of reactive systems, see for instance [4] on model checking.

[1] Email: yde@science.uva.nl

doi:10.1016/j.entcs.2004.02.038

Over the last ten years, the links between logic and automata theory have only grown stronger, to the effect that in many cases, the distinction between automata and formulas has almost disappeared. Of the many interesting results that have been obtained we just mention the connection that Janin & Walukiewicz [9] established between modal fixpoint logics, such as the modal μ-calculus, and alternating parity automata operating on labeled transition systems. For an up to date introduction to the world of automata, logic and infinite games, we refer the reader to Grädel, Thomas & Wilke [5].

Although this has to our knowledge never been exploited, or even made explicit, much of the work relating logic and automata theory has a strong coalgebraic flavour. In itself this should not come as a surprise since both (modal) logic and automata theory admit a lucrative coalgebraic perspective.

This certainly applies to logic, and to modal logic in particular. Since coalgebra can be seen as a very general model of state-based dynamics, and modal logic as a logic for dynamic systems, the relation between modal logic and coalgebra is rather tight. Starting with the work of Moss [11], the development and study of modal languages for the specification of properties of coalgebras has been actively pursued and studied by various authors, including Jacobs [6], Kurz [10], Pattinson [12], and Rössiger [14]. However, given the intended application of coalgebraic modal languages as specification formalisms restricting the behavior of state-based systems, it is rather surprising that until now no languages have been developed that incorporate explicit fixed point operators. In addition, the only work on coalgebraic modal languages in which specimens of fixed point formulas are admitted, or in which the need for coalgebraic modal fixed point logics is discussed, seems to be by Jacobs ([8] and [7], respectively).

The coalgebraic perspective on automata may not have been developed so systematically, automata theory contains some of the paradigmatic examples of coalgebras, as any introduction to the field of coalgebra witnesses. As examples we confine ourselves to mentioning Rutten's work on automata working on finite and infinite words ([15] and [17], respectively).

Summarizing the above discussion, we find that the relation between automata theory and (modal) logic has been investigated intensively and successfully, but not uniformly or systematically. Various modal languages have been developed uniformly for coalgebras of arbitrary type, but none of these languages admits explicit fixed point operators. And lastly, we see that certain kinds of automata have been studied from a coalgebraic perspective, but automata for arbitrary coalgebras have not been developed. It thus seems that there is a clear gap here, and it is precisely this gap that we intend to start filling with this paper.

We believe that the connections between automata and logic could and perhaps should be studied from a general, coalgebraic perspective, and it is the main purpose of this paper to introduce a framework for doing so. We confine our attention to the functors $\mathsf{F} : \mathsf{Set} \to \mathsf{Set}$ which are standard and preserve weak pullbacks — such functors will be called *R-standard*. For each such functor F, we will define the notion of an F-automaton; the purpose of these devices is to classify pointed F-coalgebras (pairs consisting of an F-coalgebra and an element of the carrier set of the coalgebra). The criterion under which such an automaton $\mathbb{A}$ accepts or rejects such a pointed coalgebra $(\mathbb{S}, s)$ is formulated in terms of an infinite two-player game, to be played on a certain graph induced by $\mathbb{A}$, $\mathbb{S}$ and s.

We also introduce a language $\mu\mathcal{L}^{\mathsf{F}}$ of coalgebraic fixed point logic for F-coalgebras. This language is finitary in the sense that every formula comes with a *finite* set of subformulas. Combining ideas from the game semantics for the modal μ-calculus as formulated by Janin & Walukiewicz [9], and the semantic games for coalgebraic languages introduced by Baltag [2], we provide a game-theoretical semantics for this language $\mu\mathcal{L}^{\mathsf{F}}$. Finally, the resemblance between these games and the acceptance games for F-automata leads to the main result of the paper: Theorem 2 states that any $\mu\mathcal{L}^{\mathsf{F}}$-formula can be transformed into an F-automaton that accepts precisely those pointed F-coalgebras in which the formula is true.

It should be mentioned that there are other approaches that study automata from a category-theoretic perspective. For instance, there is a series of articles by Arbib and Manes and a theory of functorial automata developed by Adámek, Trnková and others, see [1] (also for references). This work is certainly related to ours, but two differences are that the mentioned research focuses on an algebraic rather than a coalgebraic framework, and that it generalizes automata for finite rather than for infinite objects. The precise connection with this work remains to be investigated though.

Overview

We first fix notation and terminology on Set-based functors and coalgebras, and define R-standard functors; we also give a brief introduction to two-person infinite parity games. Section 3 introduces F-automata for R-standard functors, and gives a detailed description of the acceptance games for F-automata. Then we move to logic: in section 4 we introduce the syntax and semantics of the coalgebraic fixed point logic $\mu\mathcal{L}^{\mathsf{F}}$ for coalgebras over an R-standard functor F. The following section provides the details of the game-theoretic approach to the semantics of this language. Section 6 is both the most important and the briefest section of the paper: here we state the above-mentioned main result of the paper. We finish the paper with a list of ideas for further research.

2 Preliminaries

This paper presupposes some familiarity with the basic concepts of category theory and universal coalgebra. The main purpose of this section is to fix notation and terminology. We also give a very brief introduction to so-called *graph games*.

2.1 Set*-based functors and coalgebras*

Basics

We let Set denote the category of sets with functions. For an endofunctor $\mathsf{F} : \mathsf{Set} \to \mathsf{Set}$, an F-*coalgebra* is a pair $\mathbb{S} = (S, \sigma)$ consisting of a set S and a function $\sigma : S \to \mathsf{F}S$. Given two F-coalgebras $\mathbb{S} = (S, \sigma)$ and $\mathbb{T} = (T, \tau)$, a function $f : S \to T$ is an F-*coalgebra morphism* or F-*homomorphism* if $\mathsf{F}(f) \circ \sigma = \tau \circ f$. The category $\mathsf{Coalg}(\mathsf{F})$ has the F-coalgebras as objects and the F-homomorphisms as arrows. A relation $Z \subseteq S \times T$ is an F-*bisimulation* if we can impose coalgebra structure $\zeta : Z \to \mathsf{F}Z$ on Z in such a way that the two projections $\pi_1 : Z \to S$ and $\pi_2 : Z \to T$ are F-coalgebra morphisms. We write $Z : \mathbb{S}, s \leftrightarroweq \mathbb{T}, t$ if Z is a bisimulation between $\mathbb{S}$ and $\mathbb{T}$ that links $s \in S$ to $t \in T$, and $\mathbb{S}, s \leftrightarroweq \mathbb{T}, t$ if there is such a Z.

Functors and relators

Let Rel denote the category with sets as objects and binary relations as morphisms. Identity arrows in this category are given, for any set S, by $\Delta_S = \{(s, s) \mid s \in S\}$; composition of arrows in this category is ordinary relation composition, but we will write composition as is usual for functions. A functor $\mathsf{Q} : \mathsf{Rel} \to \mathsf{Rel}$ is called a *relator*.

It is well-known that Set can be embedded in Rel by the graph functor φ which is the identity on sets and maps a function $f : S \to T$ to its *graph* $\varphi(f) = \{(s, f(s)) | s \in S\}$. We say that a relator $\mathsf{Q} : \mathsf{Rel} \to \mathsf{Rel}$ *extends* a functor $F : \mathsf{Set} \to \mathsf{Set}$ if is satisfies (i) $\mathsf{Q}S = \mathsf{F}S$ for all sets S, and (ii) $\mathsf{Q}(\varphi(f)) = \varphi(\mathsf{F}(f))$ for all functions $f : S \to T$. Extensions need not always exist, but are unique if they do; we denote the extension of the functor F by $\overline{\mathsf{F}}$. It follows from a result by Carboni, Kelly and Wood [3] that an endofunctor on Set can be extended to a relator if and only if it preserves weak pullbacks. In the sequel we will need the following fact; for details, consult Rutten [16].

Fact 2.1 *Let* $\mathsf{F} : \mathsf{Set} \to \mathsf{Set}$ *be a functor that preserves weak pullbacks. Then*

1. The unique relator $\overline{\mathsf{F}}$ *extending* F *is given, for* $R \subseteq S \times T$*, by* $\overline{\mathsf{F}}(R) = \mathsf{F}(\pi_2) \circ \mathsf{F}(\pi_1)^{-1}$*.*

2. $\overline{\mathsf{F}}$ *is monotone, that is, if* $R \subseteq Q$ *then* $\overline{\mathsf{F}}(R) \subseteq \overline{\mathsf{F}}(Q)$*.*

3. Z *is a bisimulation between* $\mathbb{S}$ *and* $\mathbb{T}$ *iff* $(s,t) \in Z$ *implies* $(\sigma(s), \tau(t)) \in \overline{\mathsf{F}}Z$*, for all* s, t*.*

R-standard functors

A functor $\mathsf{F} : \mathsf{Set} \to \mathsf{Set}$ is called *standard* if it preserves inclusions; that is, whenever $f : A \hookrightarrow B$ is an inclusion, then so is $\mathsf{F}(f) : \mathsf{F}A \hookrightarrow \mathsf{F}B$. We need the following property, proved in Adámek & Trnková [1].

Fact 2.2 *Let* F *be a standard endofunctor on* Set*. Then* F *preserves finite intersections, that is:* $\mathsf{F}(A \cap B) = \mathsf{F}A \cap \mathsf{F}B$*.*

During most of this paper we will be working with endofunctors on Set that are both standard and preserve weak pullbacks. Hence, it is convenient to introduce terminology.

Definition 2.3 A functor $\mathsf{F} : \mathsf{Set} \to \mathsf{Set}$ is called *R-standard* if it is standard and preserves weak pullbacks.

2.2 Graph games

Two-player infinite graph games, or *graph games* for short, are defined as follows. For a more comprehensive account of these games, the reader is referred to Grädel, Thomas & Wilke [5].

First some preliminaries on sequences. Given a set A, let A^*, A^ω and $A^\star$ denote the collections of finite, infinite, and all, sequences over A, respectively. (Thus, $A^\star = A^* \cup A^\omega$.) Given $\alpha \in A^*$ and $\beta \in A^\star$ we define the *concatenation* of α and β in the obvious way, and we denote this element of $A^\star$ simply by juxtaposition: $\alpha\beta$. Given an infinite sequence $\alpha \in A^\omega$, let $\mathit{Inf}(\alpha)$ denote the set of elements $a \in A$ that occur infinitely often in α.

A graph game is played on a *board* B, that is, a set of *positions*. Each position $b \in B$ *belongs* to one of the two *players*, $\exists$ (Éloise) and $\forall$ (Abélard). Formally we write $B = B_\exists \cup B_\forall$, and for each position b we use $P(b)$ to denote the player i such that $b \in B_i$. Furthermore, the board is endowed with a binary relation E, so that each position $b \in B$ comes with a set $E[b] \subseteq B$ of *successors*. Formally, we say that the *arena* of the game consists of a directed bipartite graph $\mathbb{B} = (B_\exists \cup B_\forall, E)$.

A *match* of the game consists of the two players moving a pebble around the board, starting from some *initial position* b_0. When the pebble arrives at a position $b \in B$, it is player $P(b)$'s turn to move; (s)he can move the pebble to a new position of their liking, but the choice is restricted to a successor of b. Should $E[b]$ be empty then we say that player $P(b)$ *got stuck* at the position. A *match* or *play* of the game thus constitutes a (finite or infinite) sequence of positions $b_0 b_1 b_2 \ldots$ such that $b_i E b_{i+1}$ (for each i such that b_i and b_{i+1} are defined). A *full play* is either (i) an infinite play or (ii) a finite play in which the last player got stuck. A non-full play is called a *partial* play.

The rules of the game associate a *winner* and (thus) a *looser* for each full play of the game. A finite full play is lost by the player who got stuck; the winning conditions of infinite games is given by a subset *Ref* of B^ω (*Ref* is short for 'referee'): our convention is that $\exists$ is the winner of $\beta \in B^\omega$ precisely if $\beta \in \mathit{Ref}$. A *graph game* is thus formally defined as a structure $\mathcal{G} = (B_\exists \cup B_\forall, E, \mathit{Ref})$. Sometimes we want to restrict our attention to matches of a game with a certain initial position; in this case we will speak of a game that is *initialized* at this position.

Just like automata, there are various well-known kinds of winning conditions; here, we will restrict our attention to *parity games*, that is, games in which the set *Ref* is defined in terms of a parity function. A *parity function* on a set A is a map $\Omega : A \to \omega$ with finite range; put differently, a parity map on A is a map $\Omega : A \to \{0, \ldots, k\}$ for some natural number k. Given a parity map on A, we put

$$(1) \qquad A^\omega_\Omega := \{\alpha \in A^\omega \mid \max\{\Omega(a) : a \in \mathit{Inf}(\alpha)\} \text{ is even}\}.$$

In a parity game, the set *Ref* is of the form B^ω_Ω for some parity function Ω on the board B.

A *strategy* for player i is a function mapping partial plays $\beta = b_0 \cdots b_n$ with $P(b_n) = i$ to admissible next positions, that is, to elements of $E[b_n]$. In such a way, a strategy tells i how to play: a play β is *conform* or *consistent with* strategy f if for every proper initial sequence $b_0 \cdots b_n$ of β with $P(b_n) = i$, we have that $b_{n+1} = f(b_0 \cdots b_n)$. A strategy is *winning* from position $b \in B$ if it guarantees i to win any match with initial position b, no matter how the adversary plays — note that it is not required that $P(b) = i$. A position $b \in B$ is called a *winning position* for player i if i has a winning strategy from position b; the set of winning positions for i in a game $\mathcal{G}$ is denoted as $\mathit{Win}_i(\mathcal{G})$.

Parity games form an attractive and important game model because they have many nice properties, such as *history-free determinacy*. However, none of these are needed in the present paper — the interested reader is again referred to [5].

3 Coalgebraic automata theory

3.1 Basic definitions

Our first definition concerns the most important notion of the paper: F-automata.

Definition 3.1 Let F be an R-standard endofunctor on Set. An *(alternating)* F-*automaton* is a quadruple $\mathbb{A} = (A, a_I, \Delta, Acc)$, with A some finite set of objects called *states*, $a_I \in A$ the *initial state*, $\Delta : A \to \mathcal{P}\mathcal{P}\mathsf{F}A$ the *step function*, and $Acc \subseteq A^\omega$ the *acceptance condition*.

An F-automaton is called *solitary* (or *non-deterministic*) if all members of each $\Delta(a)$ are singletons. An F-automaton is called *deterministic* if for each $a \in A$ there is an element $\delta(a) \in \mathsf{F}A$ such that $\Delta(a) = \{\{\delta(a)\}\}$ (in particular, such an automaton is solitary).

The meaning of this definition should become clear below when we discuss the acceptance games. In the sequel we will never explicitly use the adjective 'alternating' when describing an automaton. We just mentioned it in the definition to make clear that in our framework, the generic automaton is alternating, and deterministic and solitary automata are special instances of alternating ones. This issue will be discussed in more detail further on.

There are various kinds of acceptance conditions known from the literature. For almost all of these, the criterion, whether an infinite sequence $\alpha \in A^\omega$ belongs Acc or not, is formulated in terms of the set $Inf(\alpha)$. For instance, a *Büchi* condition puts $\alpha \in Acc$ if and only if $Inf(\alpha)$ contains at least one of a set of special acceptance states. In this paper we will work exclusively with *parity* automata.

Definition 3.2 Let F be an R-standard endofunctor on Set. A *parity* F-*automaton* is an F-automaton $\mathbb{A} = (A, a_I, \Delta, Acc)$, such that $Acc = A^\omega_\Omega$ for some parity map $\Omega : A \to \omega$, see (1). Such an automaton is usually presented as $\mathbb{A} = (A, a_I, \Delta, \Omega)$. The map Ω is called the *parity function* of the automaton.

3.2 Acceptance game

F-automata are supposed to operate on pointed F-coalgebras. A pointed F-coalgebra is a pair $(\mathbb{S}, s)$ such that $\mathbb{S}$ is an F-coalgebra and s is an element of the (underlying set of) $\mathbb{S}$. Basically, the idea is that the F-automaton will either *accept* or *reject* a given pointed F-coalgebra. The best way to express the evaluation process leading to either acceptance or rejection, is in terms of a two-player infinite graph game, or *graph game*, see section 2. However, it is useful to first consider another example of a graph game.

Example 3.3 There are various ways to put the notion of bisimulation into this game-theoretic framework. At this stage it is very instructive to consider the following approach from BALTAG [2].

Let $\mathbb{S} = (S, \sigma)$ and $\mathbb{S}' = (S', \sigma')$ be two F-coalgebras for some endofunctor F on Set which preserves weak pullbacks. The *bisimulation game* $\mathcal{B}(\mathbb{S}, \mathbb{S}')$ between $\mathbb{S}$ and $\mathbb{S}'$ is defined as the graph game $(B_\exists, B_\forall, E, \mathit{Ref})$ with $B_\exists := S \times S'$, $B_\forall := \mathcal{P}(S \times S')$, $\mathit{Ref} := B^\omega$ (i.e., all infinite matches are winning for $\exists$), while the edge relation E is given as follows:

- in position (s, s') $\exists$ may choose any set $Z \subseteq S \times S'$ with $(\sigma(s), \sigma'(s')) \in \overline{\mathsf{F}}Z$;
- in position $Z \subseteq S \times S'$, $\forall$ may choose any element (t, t') of Z.

We leave it to the reader to verify that

$$(s, s') \in \mathit{Win}_\exists(\mathcal{B}) \text{ iff } \mathbb{S}, s \leftrightarroweq \mathbb{S}', s'.$$

The key observation for the direction from left to right is that the relation $\mathit{Win}_\exists(\mathcal{B})$ itself is a bisimulation between $\mathbb{S}$ and $\mathbb{S}$. For the other direction, let $\exists$ choose, at an arbitrary position (t, t'), any bisimulation between $\mathbb{S}$ and $\mathbb{S}'$ that links t to t', cf. Fact 2.1(3).

Definition 3.4 Let $\mathbb{A} = (A, a_I, \Delta, \Omega)$ be an F-automaton, and let $\mathbb{S} = (S, \sigma)$ be an F-coalgebra. The *acceptance game* $\mathcal{G}(\mathbb{A}, \mathbb{S})$ associated with $\mathbb{A}$ and $\mathbb{S}$ is the parity graph game $(B_\exists, B_\forall, E, \overline{\Omega})$ with

$$B_\exists := A \times S \qquad \cup\ \mathsf{F}A \times \mathsf{F}S$$
$$B_\forall := \mathcal{P}(\mathsf{F}A) \times S \cup \mathcal{P}(A \times S),$$

while E and $\overline{\Omega}$ are given by the table below:

Position: b	$P(b)$	Admissible moves: $E[b]$	$\overline{\Omega}(b)$
$(a, s) \in A \times S$	$\exists$	$\{(\Xi, s) \in \mathcal{P}(\mathsf{F}A) \times S \mid \Xi \in \Delta(a)\}$	$\Omega(a)$
$(\Xi, s) \in \mathcal{P}(\mathsf{F}A) \times S$	$\forall$	$\{(\xi, \tau) \in \mathsf{F}A \times \mathsf{F}S \mid \xi \in \Xi \text{ and } \tau = \sigma(s)\}$	0
$(\xi, \tau) \in \mathsf{F}A \times \mathsf{F}S$	$\exists$	$\{Z \in \mathcal{P}(A \times S) \mid (\xi, \tau) \in \overline{\mathsf{F}}Z\}$	0
$Z \in \mathcal{P}(A \times S)$	$\forall$	Z	0

Finally, $\mathbb{A}$ *accepts* the pointed F-coalgebra $(\mathbb{S}, s)$ if (a_I, s) is a winning position for $\exists$ in the game $\mathcal{G}(\mathbb{A}, \mathbb{S})$.

In order to get an understanding of this game, consider an F-automaton $\mathbb{A}$ and an F-coalgebra $\mathbb{S}$. Of all the positions in the game $\mathcal{G} = \mathcal{G}(\mathbb{A}, \mathbb{S})$, those in $A \times S$ are the basic ones — the other positions are just intermediate stages. Roughly, one should see a pair $(a, s) \in A \times S$ as a situation in which the automaton is in state a, inspecting the point s of the coalgebra. The aim of

$\exists$ is to show that this description 'fits'; while the aim of $\forall$ is to convince her that this is not the case. Going into detail we first look at two special cases.

First suppose that the automaton $\mathbb{A}$ is *deterministic*. That is, there is a map $\delta : A \to \mathsf{F}A$ such that for each $a \in A$ it holds that $\Delta(a) = \{\{\delta(a)\}\}$. Now at any position $(a, s) \in A \times S$ of the game $\mathcal{G}$, $\exists$ can only make one move, namely, to the position $\{(\delta(a), s)\} \in \mathcal{P}(\mathsf{F}A \times S)$; after that, $\forall$ has no choice either: he has to move the pebble to $(\delta(a), \sigma(s)) \in \mathsf{F}A \times \mathsf{F}S$. Note that this position is completely determined by the first position — hence the name 'deterministic'. A position of the form $(\delta(a), \sigma(s))$ is like the position (a, s) of the bisimulation game of Example 3.3: $\exists$ chooses a relation $Z \subseteq A \times S$ such that $(\delta(a), \sigma(s)) \in \mathsf{F}Z$, after that, $\forall$ chooses a new pair $(b, t) \in Z$, and we are back in one of the basic positions. So in the deterministic case, a parity automaton itself can be represented as a 'decorated' F-coalgebra: apart from an initial state it also carries an acceptance condition $\Omega : A \to \omega$. Likewise, the acceptance game $\mathcal{G}(\mathbb{A}, \mathbb{S})$ for such an automaton is like a 'decorated' bisimulation game. But of course, much of the power of automata working on infinite objects precisely stems from the intricacies of the 'decorations'.

Now take the more general case in which we only know that $\mathbb{A}$ is *solitary*, and consider a position $(a, s) \in A \times S$. Here $\exists$ has a real choice: she can pick any singleton $\{\alpha\}$ from $\Delta(a)$ and move the pebble to position $\{(\alpha, s)\} \in \mathcal{P}(\mathsf{F}A \times S)$. After that, $\forall$s choice is forced: he must move the pebble to position $(\alpha, \sigma(s)) \in \mathsf{F}A \times S$. Effectively then, at position (a, s) it is $\exists$ on her own who determines the later position $(\alpha, \sigma(s)) \in \mathsf{F}A \times S$ - this explains why we call such an automaton 'solitary'. Note that at positions of the form $(\alpha, \sigma(s)) \in \mathsf{F}A \times S$ the game proceeds as in the deterministic case, until another central position is reached.

Finally, we consider the most general case, in which $\mathbb{A}$ is an arbitrary automaton. Here it is still the aim to arrive, starting from a position $(a, s) \in A \times S$, at a position $(\alpha, \sigma(s)) \in \mathsf{F}A \times S$, but now $\exists$ and $\forall$ play a little 'subgame' in order to get there. In the version presented here, first $\exists$ makes a preselection, that is, she chooses some subset $\Xi \subset \mathsf{F}A$; then $\forall$ picks an element $\xi \in \Xi$, and the new position is $(\xi, \sigma(s))$; from here, play proceeds as before. In this most general case we are thus dealing with an *alternating* automaton.

3.3 Variation: chromatic F*-automata*

The reader may not have recognized his or her favorite, or at least familiar, type of automaton in Definition 3.1. In particular, the transition function or relation of standard automata operating on (infinite) words or trees take input from an alphabet or set of labels. Here we briefly indicate how this can easily be incorporated into our approach.

Definition 3.5 Let F be an endofunctor on the category Set, and C an arbitrary finite set of objects that we shall call *colors*. We let F_C denote the functor $\mathsf{F}_C S = C \times \mathsf{F}S$. F_C-coalgebras will also be called *C-colored F-coalgebras*.

Note that F_C-coalgebras are pairs of the form $\mathbb{S} = (S, \sigma)$ with $\sigma : S \to C \times \mathsf{F}S$. We use π_1 and π_2 to denote the two projection functions, and call $\pi_1\sigma(s) \in C$ the *color* of s. Now obviously, we can use F_C-automata for recognizing F_C-coalgebras, but the following definition seems to be more in line with standard usage in automata theory.

Definition 3.6 Let F be an R-standard endofunctor on Set. A *chromatic* F-*automaton over* C is a quintuple $\mathbb{A} = (A, a_I, C, \Delta, Acc)$ such that $\Delta : A \times C \to \mathcal{P}\mathcal{P}\mathsf{F}A$ (and A, a_I, and Acc are as before).

Given such an automaton and an F_C-coalgebra $\mathbb{S} = (S, \sigma)$, we define the acceptance game $\mathcal{G}_C(\mathbb{A}, \mathbb{S})$ in a very similar way as before, witnessed by the following table:

Position: b	$P(b)$	Admissible moves: $E[b]$	$\overline{\Omega}(b)$
$(a, s) \in A \times S$	$\exists$	$\{(\Xi, s) \in \mathcal{P}(\mathsf{F}A) \times S \mid \Xi \in \Delta(a, \pi_1\sigma(s))\}$	$\Omega(a)$
$(\Xi, s) \in \mathcal{P}(\mathsf{F}A) \times S$	$\forall$	$\{(\xi, \tau) \in \mathsf{F}A \times \mathsf{F}S \mid \xi \in \Xi \text{ and } \tau = \pi_2\sigma(s)\}$	0
$(\xi, \tau) \in \mathsf{F}A \times \mathsf{F}S$	$\exists$	$\{Z \in \mathcal{P}(A \times S) \mid (\xi, \tau) \in \overline{\mathsf{F}}Z\}$	0
$Z \in \mathcal{P}(A \times S)$	$\forall$	Z	0

Example 3.7 Unfortunately, we do not have the space here for a detailed example. It is not very hard, however, to show that, say, non-deterministic Büchi automata on C-labeled binary trees, can be represented as solitary, chromatic Büchi automata over C, for the binary tree functor $\mathsf{B}S = S \times S$.

It is also good to note that the differences between the two kinds of automata for recognizing C-colored F-coalgebras are only superficial:

Proposition 3.8 *F_C-automata and chromatic F-automata over C recognize the same classes of pointed F_C-coalgebras.*

In fact, there are fairly direct procedures to turn an F_C-automaton into an equivalent chromatic F-automaton over C, and vice versa. For lack of space we cannot go into the details.

3.4 Variation: logical automata

A different perspective on the step function Δ of an F-automaton $\mathbb{A}$ is that for all states a, $\Delta(a)$ is a *disjunction* of *conjunctions* of elements of $\mathsf{F}A$. In the acceptance game we see that $\exists$ chooses between the disjuncts, and $\forall$ chooses between the conjuncts. This suggests the following generalization.

Definition 3.9 Given a set X, let $\mathcal{DL}(X)$ be the smallest collection of objects that includes X and contains $\bigwedge P$ and $\bigvee P$ whenever P is a set of objects in $\mathcal{DL}(X)$.

Let F be an R-standard endofunctor on Set. A *logical* F-*automaton* is a quadruple $\mathbb{A} = (A, a_I, \Delta, Acc)$ with A, a_I and Acc as before, and $\Delta : A \to \mathcal{DL}(\mathsf{F}A)$.

The acceptance game for this $\mathbb{A}$ is defined in a completely obvious way, making $\exists$ choose between disjuncts, moving from $(\bigvee P, s)$ to (p, s) for some $p \in P$, and making $\forall$ choose between conjuncts, moving from $(\bigwedge P, s)$ to a position (p, s) with $p \in P$, until a position (α, s) is reached with $\alpha \in \mathsf{F}A$.

This generalization to logical automata is nice and useful, but it does not add any recognizing power to our automata:

Proposition 3.10 *F-automata and logical F-automata recognize the same classes of pointed F-coalgebras.*

The proposition can be proved using some standard game-theoretical argumentation (basically, it just involves applying the distributive laws of disjunction over conjunction, and vice versa).

4 Coalgebraic fixed point logic

4.1 Syntax

Definition 4.1 Let F be an R-standard endofunctor on Set, and let X be a set of objects to be called *variables*. Inductively we define, for each natural number n, the set $\mu\mathcal{L}^{\mathsf{F}}_n(X)$ of *coalgebraic fixed point formulas over X of depth n*:

- $\mu\mathcal{L}^{\mathsf{F}}_0(X)$ is the smallest set S which contains $\top$, $\bot$, and all variables in X and satisfies (i) if p and q belong to S, then so do $p \wedge q$ and $p \vee q$; and (ii) if p belongs to S, then so do $\mu x.p$ and $\nu x.p$, for each $x \in X$.
- $\mu\mathcal{L}^{\mathsf{F}}_{n+1}(X)$ is the smallest superset of $\mu\mathcal{L}^{\mathsf{F}}_n(X)$ containing the formula $\nabla\pi$ for each π that belongs to $\mathsf{F}Q$ for some finite $Q \subseteq \mu\mathcal{L}^{\mathsf{F}}_n(X)$, which is closed under the same formation rules (i) and (ii).

The union $\mu\mathcal{L}^{\mathsf{F}}(X) = \bigcup_{n\in\omega} \mu\mathcal{L}^{\mathsf{F}}_n(X)$ is the set of all coalgebraic fixed point formulas over X. Given a formula $p \in \mu\mathcal{L}^{\mathsf{F}}(X)$, we define the *depth* of p as the least natural number n such that $p \in \mu\mathcal{L}^{\mathsf{F}}_n(X)$.

Quite often we have no reason to make the set X of variables explicit and so we will frequently write $\mu\mathcal{L}^{\mathsf{F}}$ rather than $\mu\mathcal{L}^{\mathsf{F}}(X)$.

Example 4.2 Our definition is intended to generalize that of the modal μ-calculus to arbitrary R-standard endofunctors on Set. Recall that the modal μ-calculus is a language for coalgebras for the functor $\mathsf{F}S = \mathcal{P}(\mathsf{Prop}) \times \mathcal{P}(S)^{\mathsf{Act}}$, where Prop is some set of propositional variables and Act some set of atomic actions. In the formulation of the modal μ-calculus of Janin & Walukiewicz [9], the modal operators $\langle a \rangle$ and $[a]$ are replaced with a single connective '$a \to \cdot$' operating on finite sets of formulas: if Φ is a finite set of formulas, then $a \to \Phi$ is a formula. The meaning of $a \to \Phi$ can be expressed in terms of $\langle a \rangle$ and $[a]$: $a \to \Phi$ is equivalent to $\bigwedge\{\langle a \rangle p \mid p \in \Phi\} \wedge [a] \bigvee\{p \mid p \in \Phi\}$. This is of course quite familiar in coalgebraic logic, and it would not be difficult to rephrase the language of Janin & Walukiewicz in such a way that a family of modal operators remains, each expressing a condition of the form

$$\bigwedge_{q \in \mathsf{Prop}} \pm q \wedge \bigwedge_{a \in \mathsf{Act}} (a \to \Phi_a)$$

with $\pm q$ denoting either q or $\neg q$. Doing so, we would have brought the language of the modal μ-calculus exactly in the format of our definition.

Before we turn to the coalgebraic semantics of this language, there are a number of syntactic issues to be settled.

We start with the important observation that every coalgebraic fixed point formula comes with a unique *construction tree*; the key insight here is that every formula p comes with a unique, naturally defined set of 'immediate subformulas'. In case p is of the form $\nabla\pi \in \mu\mathcal{L}_n^{\mathsf{F}}$ this insight is based on the fact that for all finite sets $Q \subseteq \mu\mathcal{L}_n^{\mathsf{F}}$, and all $\pi \in \mathsf{F}Q$ there is a (unique) *smallest* set $Q' \subseteq \mu\mathcal{L}_n^{\mathsf{F}}$ such that $\pi \in \mathsf{F}Q'$ — the existence of such a set easily follows from Fact 2.2. We leave it for the reader to give a formal definition of construction trees; we do provide an explicit definition of the notion of subformula.

Definition 4.3 We will write $q \trianglelefteq p$ if q is a *subformula* of p. Inductively we define the set $\mathit{Sfor}(p)$ of subformulas of p as follows:

$$\begin{array}{lll} \mathit{Sfor}(p) & := \{p\} & \text{if } p \in \{\top, \bot\} \cup X, \\ \mathit{Sfor}(p \heartsuit q) & := \{p \heartsuit q\} \cup \mathit{Sfor}(p) \cup \mathit{Sfor}(q) & \text{if } \heartsuit \in \{\wedge, \vee\}, \\ \mathit{Sfor}(\eta x.p) & := \{\eta x.p\} \cup \mathit{Sfor}(p) & \text{if } \eta \in \{\mu, \nu\}, \\ \mathit{Sfor}(\nabla\pi) & := \{\nabla\pi\} \cup \bigcup_{p \in \mathit{Base}(\pi)} \mathit{Sfor}(p), & \end{array}$$

where $\mathit{Base}(\pi)$ denotes the smallest set Q such that $\pi \in \mathsf{F}Q$; the elements of $\mathit{Base}(\pi)$ will be called the *immediate subformulas* of $\nabla\pi$.

The following proposition can then be proved by a straightforward induction on the complexity of formulas.

Proposition 4.4 *Every formula $p \in \mu\mathcal{L}^{\mathsf{F}}$ has finitely many subformulas.*

Definition 4.5 The *fixed point operators* μ and ν *bind* the variable that they occur with, everywhere in the subformula to which they are applied. This notion of binding is completely standard, and so are the definitions of the sets $FVar(p)$ and $BVar(p)$ of *free* and *bound* variables, respectively, of a formula $p \in \mu\mathcal{L}^{\mathsf{F}}$. The set $Var(p) = FVar(p) \cup BVar(p)$ denotes the collection of *all* variables occurring in p, free or bound. As in first order logic, we will call a formula without free variables, a *sentence*.

A formula $p \in \mu\mathcal{L}^{\mathsf{F}}$ is called *clean* if no variable occurs both free and bound in p, and no two distinct occurrences of fixed point operators bind the same variable. Hence, in a clean formula p, with each $x \in BVar(p)$ we may associate a unique subformula of p where x is bound; we will denote this formula as $\eta_x x.q_x$, and call x a *μ-variable* if $\eta_x = \mu$, and a *ν-variable* if $\eta_x = \nu$. A formula $p \in \mu\mathcal{L}^{\mathsf{F}}$ is called *guarded* if every subformula $\eta x.q$ of p has the property that all occurrences of x inside q are within the scope of a ∇.

Now let p be a clean formula. We define the following relation $FrOcc \subseteq BVar(p) \times BVar(p)$:

$$FrOcc(x, y) :\iff y \text{ occurs freely in } FVar(q_x),$$

and let $\leq_p \subseteq BV(p) \times BV(p)$ denote the transitive closure of the relation $FrOcc$. This relation is called the *dependency order* of p.

4.2 Semantics

We now introduce the semantics of coalgebraic fixed point logic. Although we are primarily interested in the interpretation of sentences, we also need to worry about the semantics of formulas with free variables. For this purpose we define the notion of an F-model over a set of variables.

Definition 4.6 Let F be an R-standard endofunctor on Set, and let X be a set of variables. An F-*model* over X is a triple (S, σ, V) such that $\mathbb{S} = (S, \sigma)$ is an F-coalgebra, and $V : X \to \mathcal{P}(S)$ is a *valuation* on $\mathbb{S}$.

Given such a valuation on $\mathbb{S}$, a variable $x \in X$ and a subset $T \subseteq S$, we define the valuation $V[x \mapsto T]$ as the map given by $V[x \mapsto T](x) = T$ while $V[x \mapsto T](y) = V(y)$ for all variables $y \in X$ that are distinct from x.

Of course, it would be more in style with the coalgebraic paradigm to present an F-model (S, σ, V) as a coalgebra for the functor $\mathsf{F}_{\mathcal{P}(X)}$ (cf. Definition 3.5). We follow the present approach because it seems to lend itself better towards the treatment of fixed point operators.

Definition 4.7 Inductively we define the notion of *truth*, i.e., we define when a $\mu\mathcal{L}^{\mathsf{F}}(X)$-formula p *is true* or *holds* at a state s of a coalgebra $\mathbb{S} = (S, \sigma)$ under the valuation V.

More precisely, we define a relation $\Vdash^V \subseteq S \times \mu\mathcal{L}^{\mathsf{F}}(X)$; when the pair (s, p) belongs to $\Vdash^V$, we say that p *is true at* or *holds in* $s \in \mathbb{S}$ under the valuation V, and usually write $\mathbb{S}, V, s \Vdash p$. We also use $[\![\cdot]\!]$ for the extension of a formula in a coalgebra: $[\![p]\!]_{\mathbb{S},V} := \{s \in S \mid \mathbb{S}, V, s \Vdash p\}$.

The clauses of the inductive truth definition are as follows:

$$
\begin{array}{ll}
\mathbb{S}, V, s \Vdash \top, & \\
\mathbb{S}, V, s \not\Vdash \bot & \\
\mathbb{S}, V, s \Vdash x & \text{if } s \in V(x) \\
\mathbb{S}, V, s \Vdash p \wedge q & \text{if } \mathbb{S}, V, s \Vdash p \text{ and } \mathbb{S}, V, s \Vdash q, \\
\mathbb{S}, V, s \Vdash p \vee q & \text{if } \mathbb{S}, V, s \Vdash p \text{ or } \mathbb{S}, V, s \Vdash q, \\
\mathbb{S}, V, s \Vdash \mu x.p & \text{if } s \in \bigcap\{T \subseteq S \mid [\![p]\!]_{\mathbb{S},V[x\mapsto T]} \subseteq T\}, \\
\mathbb{S}, V, s \Vdash \nu x.p & \text{if } s \in \bigcup\{T \subseteq S \mid T \subseteq [\![p]\!]_{\mathbb{S},V[x\mapsto T]}\}, \\
\mathbb{S}, V, s \Vdash \nabla\pi & \text{if } (\sigma(s), \pi) \in \overline{\mathsf{F}}(\Vdash\restriction_{Base(\pi)}),
\end{array}
$$

where, in the last clause, the set $\Vdash\restriction_{Base(\pi)} \subseteq S \times \mu\mathcal{L}^{\mathsf{F}}(X)$ is given as $\Vdash\restriction_{Base(\pi)} = \Vdash \cap (S \times Base(\pi))$.

We say that a formula p is true *throughout* a model $\mathbb{M} = (\mathbb{S}, V)$, notation: $\mathbb{M} \Vdash p$, if $[\![p]\!]_{\mathbb{M}} \subseteq S$. A formula is *valid*, notation: $\models p$, if it is true throughout every model; two formulas p and q are called *equivalent*, notation: $p \equiv q$, if $[\![p]\!]_{\mathbb{M}} = [\![q]\!]_{\mathbb{M}}$ for every model $\mathbb{M}$.

All clauses of this truth definition are completely standard, with the possible exception of the one for $\nabla\pi$. The standard definition from the literature (cf. Moss [11]) would require that $\mathbb{S}, V, s \Vdash \nabla\pi$ if $(\sigma(s), \pi) \in \overline{\mathsf{F}}(\Vdash)$. However, given our definition of the language, and the assumption that the truth of a formula should only depend on the interpretation of its immediate subformulas, the truth definition of $\nabla\pi$ seems to be quite natural. We don't know whether there are instances in which our definition would really deviate from Moss'.

Concerning the fixed point operators, it will be convenient to introduce some further terminology.

Definition 4.8 Let S be a set, and $\varphi : \mathcal{P}(S) \to \mathcal{P}(S)$ a map. A subset $X \subseteq S$ is called a *pre-fixed point* of φ if $\varphi(X) \subseteq X$, a *post-fixed point* if $X \subseteq \varphi(X)$, and a *fixed point* if $X = \varphi(X)$.

It then immediately follows from the definitions that the set $[\![\mu x.p]\!]_{\mathbb{M}}$ is the intersection of the collection of all pre-fixed points of the map $\lambda X \subseteq S.[\![p]\!]_{\mathbb{M}[x\mapsto X]}$, while $[\![\nu x.p]\!]_{\mathbb{M}}$ is the union of the collection of all post-fixed points of this map.

4.3 Basic semantic results

Before we can do anything interesting, there are some a few technicalities that we have to get out of the way. First, we need a Finiteness Lemma stating that the truth of a formula only depends on its free variables. We omit the proof for lack of space.

Proposition 4.9 (Finiteness) *Let F be an R-standard endofunctor on Set, let $Y \subseteq X$ be two sets of variables, and let (S, σ) be an F-coalgebra. Now suppose that V and V' are two X-valuations on $\mathbb{S}$ such that $V(y) = V'(y)$ for all $y \in Y$. Then for all p with $FVar(p) \subseteq Y$, and all $s \in S$ it holds that*

$$S, \sigma, V \Vdash p \textit{ iff } S, \sigma, V' \Vdash p.$$

For *sentences* in particular, it follows from the previous proposition that it does not matter which valuation we take into consideration. This inspires the following definition.

Definition 4.10 Let F be an R-standard endofunctor on Set, p a $\mu\mathcal{L}^{\mathsf{F}}$-sentence, $\mathbb{S}$ an F-coalgebra and s a point in $\mathbb{S}$. Then we say that p is *true at s* in $\mathbb{S}$, notation: $\mathbb{S}, s \Vdash p$, if $\mathbb{S}, V, s \Vdash p$ for some valuation V, (or, equivalently, for all valuations V).

Next we turn to the Monotonicity Lemma.

Proposition 4.11 (Monotonicity) *Let F be an R-standard endofunctor on Set, X a set of variables, and $\mathbb{S}$ an F-coalgebra. Now suppose that V and V' are two X-valuations on $\mathbb{S}$ such that $V(x) \subseteq V'(x)$ for all $x \in X$. Then for all p with $FVar(p) \subseteq X$ it holds that*

$$[\![p]\!]_{\mathbb{S},V} \subseteq [\![p]\!]_{\mathbb{S},V'},$$

that is: for all $s \in S$ we have that $S, \sigma, V \Vdash p$ only if $S, \sigma, V' \Vdash p$.

Proof. This can be proved by a standard induction on the complexity of p. The proof in the inductive case of $p = \nabla\pi$ is based on the fact that $\overline{\mathsf{F}}$ is monotone (Fact 2.1). □

Remark 4.12 The Monotonicity Lemma justifies the terminology *fixed point* in the name of our formalism: by the Knaster-Tarski Theorem in fixed point theory, every monotone operation φ on a complete lattice (such as a full power set) has a least and a greatest fixed point, and these can be obtained as the intersection of the collections of pre-fixed points and post-fixed points of φ, respectively. In particular, for every formula p and every model $\mathbb{M} = (S, \sigma, V)$, the set $[\![\mu x.p]\!]_{\mathbb{M}}$ is the least fixed point of the operation $\lambda X \in \mathcal{P}(S).\, [\![p]\!]_{\mathbb{M}[x \mapsto X]}$, and the set $[\![\nu x.p]\!]_{\mathbb{M}}$ is the greatest fixed point of this operation.

Remark 4.13 It also follows from standard fixed point theory that least and greatest fixed points of monotone operations on complete lattices (such as full power set algebras) can be approximated by ordinal unfoldings. Using this, there is a nice connection between our coalgebraic fixed point logic, and more standard coalgebraic logics.

Let $\mathcal{L}^{\mathsf{F}}_{\infty}(X)$, the language of infinitary coalgebraic F-logic, be the smallest collection of formulas which includes the set $\{\top, \bot\} \cup X$ and satisfies (i) if β is some ordinal, and $\{p_\alpha \mid \alpha < \beta\}$ is a set of $\mathcal{L}^{\mathsf{F}}_{\infty}(X)$-formulas, then both $\bigwedge_{\alpha<\beta} p_\alpha$ and $\bigvee_{\alpha<\beta} p_\alpha$ belong to $\mathcal{L}^{\mathsf{F}}_{\infty}(X)$, and (ii) if π belongs to $\mathsf{F}Q$ for some $Q \subseteq S$, then $\nabla\pi$ belongs to S. Note that F-models, with the obvious interpretation for $\bigwedge$ and $\bigvee$, form a natural semantics for this language.

Now for each ordinal α there is a translation t^α mapping $\mu\mathcal{L}^{\mathsf{F}}$-sentences to $\mathcal{L}^{\mathsf{F}}_{\infty}$-formulas. This translation is defined as follows; first, we define, for any $\mathcal{L}^{\mathsf{F}}_{\infty}(X)$-formula p, any variable $x \in X$, and any ordinal α, the formulas $\mu_\alpha.p$ and $\nu_\alpha x.p$ via transfinite induction:

$$
\begin{aligned}
\mu_0 x.p &:= \bot, & \nu_0 x.p &:= \top,\\
\mu_{\alpha+1} x.p &:= p[\mu_\alpha x.p/x], & \nu_{\alpha+1} x.p &:= p[\nu_\alpha x.p/x],\\
\mu_\lambda x.p &:= \textstyle\bigvee_{\alpha<\lambda} \mu_\alpha x.p, & \nu_\lambda x.p &:= \textstyle\bigwedge_{\alpha<\lambda} \nu_\alpha x.p.
\end{aligned}
$$

Using these formulas, one puts

$$
\begin{array}{lll}
t^\alpha p & := p & \text{for } p \in \{\top, \bot\} \cup X,\\
t^\alpha(p \heartsuit q) & := t^\alpha p \heartsuit t^\alpha q & \text{for } \heartsuit \in \{\wedge, \vee\},\\
t^\alpha(\eta x.p) & := \eta_\alpha x.t^\alpha p & \text{for } \eta \in \{\mu, \nu\},\\
t^\alpha(\nabla\pi) & := \nabla(\mathsf{F}t^\alpha)(\pi). &
\end{array}
$$

Observe that t^α translates $\mu\mathcal{L}^{\mathsf{F}}$-sentences into variable-free $\mathcal{L}^{\mathsf{F}}_{\infty}$-formulas.

One can show that these translations locally embed $\mu\mathcal{L}^{\mathsf{F}}$ inside $\mathcal{L}^{\mathsf{F}}_{\infty}$, in the following sense:

(2) $[\![p]\!]_{\mathbb{M}} = [\![t^\alpha p]\!]_{\mathbb{M}}$, for any F-model $\mathbb{M} = (S, \sigma, V)$ and any ordinal $\alpha > |S|^+$.

Note however, that in general, the 'unfolding ordinal' α of (2) depends on the size of the model $\mathbb{M}$. Coalgebraic fixed point logic cannot be embedded in infinitary coalgebraic logic, as is known from the modal μ-calculus.

An important property of our coalgebraic fixed point logic is that truth is bisimulation invariant. Using the appropriate notion of bisimulation for F-models this can be proven for arbitrary $\mu\mathcal{L}^{\mathsf{F}}$-formulas, but here we state it just for sentences.

Proposition 4.14 *Let $\mathbb{S}$ and $\mathbb{S}'$ be two F-coalgebras. Then for any bisimulation $Z \subseteq S \times S'$ and any two points $s \in S$, $s' \in S'$ with $(s, s') \in Z$, and any $\mu\mathcal{L}^{\mathsf{F}}$-sentence p it holds that*

$$\mathbb{S}, s \Vdash p \text{ iff } \mathbb{S}', s' \Vdash p.$$

Proof. A simple proof for this proposition uses the ordinal unfolding of Remark 4.13, and the easily established fact that truth of $\mathcal{L}^{\mathsf{F}}_{\infty}$-sentences is a bisimulation invariant property. □

We are now ready to state our last basic semantic result; for lack of space we have to omit the (fairly standard) proof.

Proposition 4.15 (Normal Form) *Let F be an R-standard endofunctor on* Set. *Then every formula $p \in \mu\mathcal{L}^{\mathsf{F}}$ is equivalent to some clean, guarded formula p' of the same depth.*

5 Game semantics

In this section we develop a game-theoretic characterization of the semantics of our coalgebraic fixed point logics, generalizing results on for instance the modal μ-calculus to a general coalgebraic framework.

5.1 Evaluation games

Given an F-model (S, σ, V) and a coalgebraic fixed point formula q, we will define the *evaluation game* $\mathcal{E} = \mathcal{E}(S, \sigma, p)$ as the following infinite two-player graph game.

Definition 5.1 Let F be an R-standard endofunctor on Set. Given an F-model $\mathbb{M} = (S, \sigma, V)$ and a clean coalgebraic fixed point formula q, we first define the *arena* of the *evaluation game* $\mathcal{E} = \mathcal{E}(\mathbb{M}, q)$.
The board of $\mathcal{E}$ is given as the set

$$B = Sfor(q) \times S \cup \mathcal{P}(Sfor(q) \times S).$$

The partition of B into positions for $\exists$ and $\forall$, respectively, and the edge relation E of the graph are given by the table of Figure 1.

Note that positions of the form (x, s) or $(\eta x.p, s)$ have a *unique* successor, whence the moves that are made at such positions are completely determined. Thus it does not matter to which player these positions are assigned.

Position: b	Player: $P(b)$	Admissible moves: $E[b]$
$(\bot, s)$	$\exists$	$\varnothing$
$(\top, s)$	$\forall$	$\varnothing$
$(p_1 \wedge p_2, s)$	$\forall$	$\{(p_1, s), (p_2, s)\}$
$(p_1 \vee p_2, s)$	$\exists$	$\{(p_1, s), (p_2, s)\}$
(x, s) with $x \notin BVar(q)$, $s \in V(x)$	$\forall$	$\varnothing$
(x, s) with $x \notin BVar(q)$, $s \notin V(x)$	$\exists$	$\varnothing$
(x, s) with $x \in BVar(q)$	-	(q_x, s)
$(\eta x.p, s)$	-	(p, s)
$(\nabla\pi, s)$	$\exists$	$\{Z \subseteq Base(\pi) \times S \mid (\pi, \sigma(s)) \in \overline{\mathsf{F}}(Z)\}$
$Z \subseteq Sfor(q) \times S$	$\forall$	Z

Fig. 1. Admissible moves in the evaluation game

In order to get some intuitions for this kind of game, the reader is advised to assign the following *aims* to the players. Basically, in a position (p, s) it is the aim of $\exists$ to show that p is actually *true* at s, while $\forall$ tries to convince her that this is not the case. This already explains the rules for positions of the form (p, s) with p an atomic constant, a conjunction, or a disjunction. For instance, in $(p_1 \vee p_2, s)$, $\exists$ may win by winning either (p_1, s) or (p_2, s), because $p_1 \vee p_2$ holds at s if either p_1 or p_2 does.

Each time during a match when the pebble moves from a position (x, s) to its successor (q_x, s), we say that the fixed point variable x is *unfolded*. Roughly spoken, the intuition behind this is that the formula $\eta_x.q_x$ (represented by x) is equivalent to the formula $q_x[\eta_x.q_x/x]$ (represented by q_x). This applies to both μ and ν-variables. The difference between the two kinds of fixed point variables only comes out in infinite matches. We need the following observation, which is not hard to see.

Proposition 5.2 *Let* F *be an R-standard endofunctor on* Set*,* q *a clean* $\mu\mathcal{L}^{\mathsf{F}}$*-formula and* $\mathbb{M}$ *an* F*-model. Then in any infinite match* β *of the game* $\mathcal{E}(\mathbb{M}, q)$*, the set of variables that are unfolded infinitely often during* β *contains a maximal member (in the dependency order).*

Definition 5.3 Let F be an R-standard endofunctor on Set. Given an F-model $\mathbb{M} = (S, \sigma, V)$ and a clean coalgebraic fixed point formula q, we now define the *winning conditions* of the *evaluation game* $\mathcal{E} = \mathcal{E}(\mathbb{M}, q)$.

Let β be a full match played on the arena of $\mathcal{E}$, and let x be the highest ranking fixed point variable that got unfolded infinitely often during β. Then

- β is winning for $\exists$ if either (i) β is finite and $\forall$ got stuck, or (ii) β is infinite and x is a ν-variable;
- β is winning for $\forall$ if either (i) β is finite and $\exists$ got stuck, or (ii) β is infinite

and x is a μ-variable.

5.2 Adequacy of game semantics

The following theorem states that the evaluation games as introduced above, indeed constitute an equivalent characterization for the semantics of coalgebraic fixed point formulas.

Theorem 1 (Adequacy) *Let* F *be an R-standard endofunctor on* Set*. Then for any* $\mu\mathcal{L}^{\mathsf{F}}$*-formula* q*, any* F*-model* (S, σ, V) *and any state* $s \in S$ *it holds that*

$$S, \sigma, V, s \Vdash q \text{ iff } (q, s) \in Win_{\exists}(\mathcal{E}(\mathbb{M}, q)).$$

6 Automata and fixed point logic

The reader will have noticed the similarity between the evaluation game of a formula and the acceptance game of an automaton. But the connection is much tighter than a mere resemblance, witness the Theorem, which forms the main result of the paper:

Theorem 2 (Formulas are automata) *Let* F *be an R-standard endofunctor on* Set*. Then any* $\mu\mathcal{L}^{\mathsf{F}}$*-sentence* q *can be transformed into an* F*-automaton* $\mathbb{A}_q$ *such that for any pointed* F*-coalgebra* $(\mathbb{S}, s)$*:*

$$\mathbb{S}, s \Vdash q \text{ iff } \mathbb{A}_q \text{ accepts } (\mathbb{S}, s).$$

Unfortunately we cannot go into the details of the proof. Let us just mention that roughly, the first step of the proof is to turn p into a clean, guarded equivalent p'. In the second and most important step of the construction we construct an automaton-like object based on the set $Sfor(p)$ of subformulas of p, and in the next step this structure is tidied up into a *logical* automaton as presented in Definition 3.9. The final step of the construction then simply consists of replacing this logical automaton with a standard one, as in Proposition 3.10.

7 Further research

We believe that it is interesting and useful to develop the automata theory for coalgebras on an (almost) arbitrary functor $\mathsf{F} : \mathsf{Set} \to \mathsf{Set}$, and to apply this theory to the study of coalgebraic fixed point logics. It is obvious that in this paper we have only scratched the surface of these topics. Of the many questions that naturally arise I just mention the following.

(i) As we already mentioned in the introduction, there are earlier studies of automata that are based on categories and functors, see for instance Adámek & Trnková [1]. This connection clearly has to be investigated further.

(ii) Important issues in the theory of automata include the question whether a given automaton can be replaced with an equivalent one that satisfies some additional properties, and the closure properties of the class of recognizable languages (here defined in a broad sense). All such questions can be studied for *other* types of coalgebras, and from a *general* coalgebraic perspective. It could be hoped that coalgebraic methods would produce some new insights.

As particularly interesting questions in this line we mention the following:

'determinization' and 'solitarification' For which functors F can we replace each solitary automata with an equivalent deterministic one, and/or each alternating automata with an equivalent solitary one?

'closure under complementation' For which functors F, and which kinds of F-automata, can we find, for a given F-automaton $\mathbb{A}$ of the mentioned type, another F-automaton $\overline{\mathbb{A}}$ of the same type, with the property that $\overline{\mathbb{A}}$ recognizes precisely the *complement* of the class of pointed automata accepted by $\mathbb{A}$?

(iii) Our parity F-automata have a coalgebraic shape themselves: the automaton $\mathbb{A} = (A, a_I, \Delta, \Omega)$ can be represented as a pointed coalgebra over the functor $\mathsf{F}_{\mathsf{Aut}}S = \mathcal{P}(\mathcal{P}(\mathsf{F}S)) \times \omega$. This perspective clearly needs investigation – recall that the coalgebraic perspective on ordinary automata (operating on finite words) has already proven to be very enlightening, see Rutten [15].

(iv) Our definition of coalgebraic fixed point logic is only one out of many. In fact, fixed point operators may be added to any kind of language of coalgebraic logic. It would be good to see more case studies on coalgebraic fixed point logics from an automata-theoretic perspective. Related to one of the above questions, I would like to understand what happens if we add negation to the language $\mu\mathcal{L}^{\mathsf{F}}$ discussed in section 4. But also, the relation between the modal μ-calculus and a fixed point extension of the coalgebraic modal logics developed in Pattinson [12] might be an intriguing object of study.

References

[1] J. Adámek and V. Trnková. *Automata and Algebras in Categories.* Kluwer Academic Publishers, 1990.

[2] A. Baltag. A logic for coalgebraic simulation. In H. Reichel, editor, *Coalgebraic Methods in Computer Science (CMCS'00)*, volume 33 of *Electronic Notes in Theoretical Computer Science*, pages 41–60, 2000.

[3] A. Carboni, G. Kelly, and R. Wood. A 2-categorical approach to change of base and geometric morphisms I. Technical Report 90-1, Department of Pure Mathematics, University of Sydney, 1990.

[4] E. Clarke, O. Grumberg, and D. Peled, editors. *Model Checking*. The MIT Press, 2000.

[5] E.Graedel, W. Thomas, and T. Wilke, editors. *Automata, Logic, and Infinite Games*, volume 2500 of *LNCS*. Springer, 2002.

[6] B. Jacobs. Many-sorted coalgebraic modal logic: a model-theoretic study. *Theoretical Informatics and Applications*, 35(1):31–59, 2001.

[7] B. Jacobs. Exercises in coalgebraic specification. In R. Backhouse, R. Crole, and J. Gibbons, editors, *Algebraic and Coalgebraic Methods in the Mathematics of Program Construction*, volume 2297 of *LNCS*, pages 237–280, Berlin, 2002. Springer.

[8] B. Jacobs. The temporal logic of coalgebras via galois algebras. *Mathematical Structure in Computer Science*, 12:875–903, 2002.

[9] D. Janin and I. Walukiewicz. Automata for the modal μ-calculus and related results. In *Proc. 20th International Symposium on Mathematical Foundations of Computer Science (MFCS'95)*, volume 969 of *LNCS*, pages 552–562, Berlin, 1995. Springer.

[10] A. Kurz. A co-variety-theorem for modal logic. In M. Zakharyaschev, K. Segerberg, M. de Rijke, and H. Wansing, editors, *Advances in Modal Logic 2*. Center for the Study of Language and Information, Stanford University, 2001.

[11] L. Moss. Coalgebraic logic. *Annals of Pure and Applied Logic*, 96:277–317, 1999. Erratum published APAL **99** (1999) 241–259.

[12] D. Pattinson. Semantical principles in the modal logic of coalgebras. In H. Reichel and A. Ferreira, editors, *Proc. 18th International Symposium on Theoretical Aspects of Computer Science (STACS 2001)*, volume 2010 of *LNCS*, Berlin, 2001. Springer.

[13] M.O. Rabin. Decidability of second-order theories and automata on infinite trees. *Transactions of the American Mathematical Society*, 141:1–35, 1969.

[14] M. Rößiger. Coalgebras and modal logic. In H. Reichel, editor, *Coalgebraic Methods in Computer Science (CMCS'00)*, volume 33 of *Electronic Notes in Theoretical Computer Science*, pages 299–320, 2000.

[15] J. Rutten. Automata and coinduction (an exercise in coalgebra). In D. Sangiorigi and R. de Simone, editors, *Proc. 9th International Conference on Concurrency Theory (CONCUR'98)*, volume 1466 of *LNCS*, pages 194–218. Springer, 1998.

[16] J. Rutten. Relators and metric bisimulations. In B. Jacobs, L. Moss, H. Reichel, and J. Rutten, editors, *Coalgebraic Methods in Computer Science (CMCS'98)*, volume 11 of *Electronic Notes in Theoretical Computer Science*, pages 257–263, 1998.

[17] J. Rutten. Behavioural differential equations: a coinductive calculus of streams, automata, and power series. *Theoretical Computer Science*, 308:1–53, 2003.