Automata and Fixed Point Logic: a Coalgebraic Perspective

YDE VENEMA*

Abstract

This paper generalizes existing connections between automata and logic to a coalgebraic abstraction level.

Let $\mathsf{F} : \mathsf{Set} \to \mathsf{Set}$ be a standard functor that preserves weak pullbacks. We introduce various notions of F-automata, devices that operate 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 the two approaches are equivalent in expressive power. We prove that any coalgebraic fixed point formula can be transformed into an F-automaton that accepts precisely those pointed F-coalgebras in which the formula holds. And conversely, we prove that any F-automaton can be converted into an equivalent fixed point formula that characterizes the pointed F-coalgebras accepted by the automaton.

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 [22], but also quite concrete applications in computer science, such as tools for the automatic verification of reactive systems, see for instance [6] on model checking. Of the many results that have been obtained in recent years, let us just mention the characterization, by JANIN & WALUKIEWICZ [13], of the modal μ-calculus as the bisimulation invariant fragment of monadic second order logic over the class of all labelled transition systems. Applications in logic of fundamental automata theoretic results are generally based on the observation that there is no fundamental distinction between automata and formulas. This holds of the results of Janin and Walukiewicz, who introduce the notion of an alternating parity automaton operating on labelled transition systems in order to capture the formulas of the modal μ-calculus. For an

*Institute for Logic, Language and Computation, University of Amsterdam, Plantage Muidergracht 24, NL–1018 TV Amsterdam. E-mail: yde@science.uva.nl.

up to date introduction to the world of automata, logic and infinite games, we refer the reader to Grädel, Thomas & Wilke [7].

Now in general, it has come out that most of the key results in automata theory can be proved for word and tree automata alike, and many of these results can even be formulated and proved for automata that operate on more complex objects such as graphs or labelled transition systems. The approach of Niwiński [20] lifts the theory to a higher level of generality where automata are devices operating on arbitrary relational structures. It is our intention here take a natural further abstraction step, namely, to study the area from a *coalgebraic perspective.* In fact, much of the work in this particular border area of (modal) logic and automata theory has a strong coalgebraic flavor. In itself this should not come as a surprise, since the kind of objects (words, trees, graphs, ...) that are studied here admit a natural coalgebraic presentation, and 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, there are interesting links between the two fields. One of the first to realize this, in the late 1980s, was Abramsky [1]. Barwise & Moss [4], a rich source of material on a great variety of circular systems, contains the outline of Moss' general approach towards coalgebraic logic, discussed in more detail by Moss [17]. Over subsequent years, the development and study of modal languages for the specification of properties of coalgebras has been actively pursued and studied by various authors, including Baltag [3], Jacobs [9], Kurz [15], Pattinson [21], and Rössiger [23]. 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 ([11] and [10], respectively).

When it comes to the coalgebraic perspective on automata theory, the standard deterministic and non-deterministic automata operating on finite words have been recognized as paradigmatic examples of coalgebras, as any introduction to the field of coalgebra witnesses. As an example of more substantial work in this area we refer the reader to Rutten [24, 26]. However, as far as we are aware, automata as (finitary) objects classifying possibly infinite coalgebras, have until now not been studied from an explicitly coalgebraic perspective.

Summarizing the above discussion, we find that the relation between automata theory and (modal) logic has been investigated intensively and successfully, and has a strong coalgebraic flavor. Various modal languages have been developed, in a uniform fashion, for coalgebras of arbitrary type, but none of these languages admit 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. The aim for developing such a coalgebraic framework is not so much to develop new ideas in automata theory, as to provide a common generalization for existing notions that are known from the theory of more specific kinds of automata. This abstract perspective could then be of use for many purposes. For instance, it could be instrumental to find the right notion of automaton for other kinds of coalgebras. It could also be employed to prove interesting results on coalgebraic logics. And finally, it may find applications in the form of *uniform* proofs for key results in automata theory, and hence, increase our understanding of the field.

The main purpose of this article, which grew out of the conference paper [28], is to introduce such a coalgebraic perspective on automata theory. We confine our attention to 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 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 $\mathbb{A}$ and $\mathbb{S}$.

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 [12], and the semantic games for coalgebraic languages introduced by Baltag [3], in Theorem 1 we provide a game-theoretic 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 technical results of the paper: Theorem 2 states that any $\mu\mathcal{L}^{\mathsf{F}}$-formula can be transformed into a certain kind of F-automaton that accepts precisely those pointed F-coalgebras in which the formula is true. And Theorem 3 states that, conversely, with any F-automaton we may associate a $\mu\mathcal{L}^{\mathsf{F}}$-formula holding precisely at those pointed F-coalgebras that are accepted by the automaton. However, we do not put much focus on technical results, since we believe that the main contribution of the paper is of a conceptual nature.

It should be mentioned that there are other approaches in which the notion of automaton is lifted to a category-theoretic level. 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 [2] (also for further references). Although this work bears some resemblance to ours, there are at least two differences: first, the mentioned research focuses on an algebraic rather than a coalgebraic framework, and second, it generalizes automata for finite rather than for infinite objects. Nevertheless, it would be useful to investigate the precise connection with this line of research.

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 graph games. In section 3 we introduce our coalgebraic perspective on automata theory by reviewing some of the more familiar kinds of automata. Section 4 provides the general definition of F-automata for R-standard functors (in many different but equivalent flavors), and gives a detailed description of the acceptance games for F-automata. Then we move to logic: in section 5 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 next section provides the details of the game-theoretic approach to the semantics of this language. Section 7 states and proves the above-mentioned main results of the paper. We finish the paper with a list of ideas for further research.

Acknowledgements I would like to thank Alexandru Baltag, Clemens Kupke, and the two anonymous referees, for many useful comments and suggestions for improvement of the paper.

2 Preliminaries

This paper presupposes some familiarity with the basic concepts of coalgebra and automata theory. 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 $\sigma : S \to \mathsf{F}S$. 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) $\mathbb{S}$. Given two F-coalgebras $\mathbb{S} = (S, \sigma)$ and $\mathbb{S}' = (S', \sigma')$, a function $f : S \to S'$ is an F-*coalgebra morphism* or F-*homomorphism* if $\mathsf{F}(f) \circ \sigma = \sigma' \circ f$. The category $\mathsf{Coalg}(\mathsf{F})$ has the F-coalgebras as objects and the F-homomorphisms as arrows. A relation $Z \subseteq S \times S'$ 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 $\pi : Z \to S$ and $\pi' : Z \to S'$ are F-coalgebra morphisms. We write $Z : \mathbb{S}, s \leftrightarrows \mathbb{S}', s'$ if Z is a bisimulation between $\mathbb{S}$ and $\mathbb{S}'$ that links $s \in S$ to $s' \in S'$, and $\mathbb{S}, s \leftrightarrows \mathbb{S}', s'$ 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, denoted by $\circ$. 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 which is the identity on sets and maps a function $f : S \to T$ to its *graph* $Gr(f) = \{(s, f(s)) | s \in S\}$. We say that a relator $\mathsf{Q} : \mathsf{Rel} \to \mathsf{Rel}$ *extends* a functor $F : \mathsf{Set} \to \mathsf{Set}$ if it satisfies (i) $\mathsf{Q}S = \mathsf{F}S$ for all sets S, and (ii) $\mathsf{Q}(Gr(f)) = Gr(\mathsf{F}(f))$ 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 $\overline{\mathsf{F}}$. The image $\overline{\mathsf{F}}(R) \subseteq \mathsf{F}S \times \mathsf{F}T$ of a relation $R \subseteq S \times T$ is called the *relation lifting* of R under F. It can be proved that an endofunctor on Set can be extended to a relator if and only if it preserves weak pullbacks. This result is usually attributed to Carboni, Kelly & Wood [5], but it also follows as a special case of an earlier result by Trnková, see [27, Observation 2.10], or [2, section V.2.10] for a proof. In the sequel we will need the following fact; details can be found in Rutten [25] (or be proved easily). We use $(\cdot)^{\smile}$ to denote relation converse.

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

1. *The relator* $\overline{\mathsf{F}}$ *extending* F *is given, for* $R \subseteq S \times S'$*, by* $\overline{\mathsf{F}}(R) = Gr(\mathsf{F}(\pi)) \circ Gr(\mathsf{F}(\pi'))^{\smile}$.
2. $\overline{\mathsf{F}}$ *is monotone, that is, if* $R \subseteq Q$ *then* $\overline{\mathsf{F}}(R) \subseteq \overline{\mathsf{F}}(Q)$.
3. $\overline{\mathsf{F}}$ *commutes with taking relation converse:* $\overline{\mathsf{F}}(R^{\smile}) = (\overline{\mathsf{F}}R)^{\smile}$.
4. Z *is a bisimulation between* $\mathbb{S}$ *and* $\mathbb{S}'$ *iff* $(\sigma(s), \sigma'(s')) \in \overline{\mathsf{F}}Z$ *for all* $(s, s') \in Z$.

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 [2].

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

As an immediate consequence of this, one can show that if an object ξ belongs to a set $\mathsf{F}A$ for some finite set A, then there is a *smallest* subset $Base(\xi) \subseteq A$ such that $\xi \in \mathsf{F}Base(\xi)$.

Through 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. $\lhd$

As a useful property of R-standard functors, we mention the following.

Proposition 2.4 *Let* F *be an R-standard endofunctor on* Set. *Suppose that* $(\alpha, \beta) \in \overline{\mathsf{F}}Z$ *for some relation* $Z \subseteq A \times B$*, and that* A' *and* B' *are subsets of* A *and* B *respectively, such that* $\alpha \in \mathsf{F}A'$ *and* $\beta \in \mathsf{F}B'$. *Then we have* $(\alpha, \beta) \in \overline{\mathsf{F}}(Z \cap (A' \times B'))$.

Proof. Let, for two sets $X \subseteq Y$, $\iota_{XY} : X \hookrightarrow Y$ denote the inclusion map. Then by standardness of F we have that $\mathsf{F}(\iota_{XY}) = \iota_{\mathsf{F}X\mathsf{F}Y}$. Now let A, A', B, B', Z, α and β be as stated above. Then it is immediate that $(\alpha, \beta) \in Gr(\iota_{\mathsf{F}A'\mathsf{F}A}) \circ \overline{\mathsf{F}}Z \circ Gr(\iota_{\mathsf{F}B'\mathsf{F}B})^{\smile}$. But a straightforward calculation, using earlier mentioned properties of $\overline{\mathsf{F}}$, shows that $Gr(\iota_{\mathsf{F}A'\mathsf{F}A}) \circ \overline{\mathsf{F}}Z \circ Gr(\iota_{\mathsf{F}B'\mathsf{F}B})^{\smile} = \overline{\mathsf{F}}(Gr(\iota_{A'A}) \circ Z \circ Gr(\iota_{B'B})^{\smile})$. From this the proposition is immediate. QED

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 [7].

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 $Inf(\alpha)$ 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 $\mathbb{B} = (B_\exists, B_\forall, E)$.

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 $b_0b_1b_2\dots$ such that b_iEb_{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 condition for 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 $\mathcal{G} = (B_\exists, B_\forall, E, \mathit{Ref})$. 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.

Various kinds of winning conditions are known. In a *parity game*, the set *Ref* is defined in terms of a *parity function* on the board B, that is, a map $\Omega : B \to \omega$ with finite range. More specifically, the set *Ref* is of the form

$$B^\omega_\Omega := \{\beta \in B^\omega \mid \max\{\Omega(b) : b \in \mathit{Inf}(\beta)\} \text{ is even}\}. \tag{1}$$

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 for i 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 *history free* if it only depends on the current position of the match, that is, $f(\beta) = f(\beta')$ whenever β and β' are partial plays with the same last element (which belongs to the appropriate player). A strategy is *winning for player i* 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 this definition also applies to positions b for which $P(b) \neq i$. 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 important game model because they have many attractive properties, such as *history-free determinacy.*

Fact 2.5 *Let $\mathcal{G} = (B_\exists, B_\forall, E, \Omega)$ be a parity graph game. Then*

1. *$\mathcal{G}$ is determined: $B = \mathit{Win}_\exists(\mathcal{G}) \cup \mathit{Win}_\forall(\mathcal{G})$.*
2. *Each player i has a history free strategy which is winning from any position in $\mathit{Win}_i(\mathcal{G})$.*

The determinacy of parity games follows from a far more general game-theoretic result concerning Borel games, due to Martin [16]. The fact that winning strategies in parity games can always be taken to be history free, was independently proved in MOSTOWSKI [18] and EMERSON & JUTLA [8].

3 Automata on infinite objects

In this section we intend to supply a gentle introduction to our general definition of an automaton operating on coalgebras, by discussing the shape of some finite automata that are well known from the literature. While we subsequently increase the (conceptual) complexity of these machines, their overall shape will be fixed as 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*, Δ some kind of *transition function*, and $Acc \subseteq A^\omega$ the *acceptance condition*.

There are in fact quite a few dimensions along which one may classify such automata. For instance, an important criterion, and one that we will encounter here as well, concerns the flavor of the transition function; this flavor makes whether we call the automaton *deterministic*, *non-deterministic*, or *alternating*. A second useful criterion is based on the *acceptance condition* of the device; examples include the *Büchi* condition, and the *parity* condition that we will focus on in this paper.

However, both of these criteria are fairly orthogonal to the aim of this paper. Our purpose here is to start with a classification of finite automata according to the *kind of object* on which the device operates (words, trees, or graphs). We hope that our presentation will convince the reader that the obvious similarities in the definition of an automaton *accepting* an object, are essentially coalgebraic in nature. This naturally leads to the general definition of an automaton that operates on pointed coalgebras of type F, where F is an arbitrary R-standard endofunctor on Set.

Let us first fix some terminology and notation. Throughout this section, we will work with a fixed alphabet, or color set, C.

Definition 3.1 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 $\mathsf{F}_C S = C \times \mathsf{F}S$; that is, F_C maps a set S to the set $C \times \mathsf{F}S$ (and a function $f : S \to S'$ to the function $id_C \times \mathsf{F}f : C \times \mathsf{F}S \to C \times \mathsf{F}S'$). F_C-coalgebras will also be called *C-colored F-coalgebras*.

We will usually denote F_C-coalgebras as triples $\mathbb{S} = (S, \gamma, \sigma)$, with $\gamma : S \to C$ the *coloring* and $\sigma : S \to \mathsf{F}S$ the F-coalgebra map. $\lhd$

Infinite words over an alphabet Σ can thus be seen as (special) Id_Σ-coalgebras, where Id is the identity functor on Set. Likewise, infinite Σ-labeled binary trees are special coalgebras for the functor $(\mathsf{Id} \times \mathsf{Id})_\Sigma$. The third and last kind of objects for which we will consider finite automata in this section are the C-colored coalgebras for the power set functor $\mathcal{P}$. Recall that there are plenty of examples of $\mathcal{P}$-coalgebras in the literature, since any binary relation $R \subseteq S \times S$ can be presented as the $\mathcal{P}$-coalgebra map sending a point $s \in S$ to the collection $\{t \in S \mid (s,t) \in R\}$ of its R-successors. As particular examples we mention graphs and Kripke frames; Kripke models, say, over a collection Prop of proposition letters, can be seen as Kripke frames that are colored by the collection of subsets of Prop.

The process of an automaton traversing and scanning a coalgebra structure needs a starting point: the first letter of a word, the root of a tree. In general, it is *pointed* coalgebras (see section 2.1) rather than coalgebras per se that are the objects of investigation for our automata.

Automata on infinite words

To start with, consider simple automata operating on infinite words. In the deterministic flavor, these are objects $\mathbb{A} = (A, a_I, \delta, Acc)$ where the transition function is of the form

$$\delta : A \times C \to A.$$

If we let such a device operate on an infinite C-word $\gamma = c_0c_1c_2\ldots$, the result is a so-called *run*, that is, a sequence $\rho = a_0a_1a_2\ldots$ such that $a_0 = a_I$ and $a_{i+1} = \delta(a_i, c_i)$ for all $i \in \omega$. Now $\mathbb{A}$ is defined to *accept* γ if and only if this run, which is uniquely determined by $\mathbb{A}$ and γ, belongs to the set Acc. (In the case of a standard finite automaton, the acceptance condition is given by a subset $F \subseteq A$ of *final*, or more appropriately, *accepting* states. An infinite word is accepted by such a machine if at least one of these accepting states occurs infinitely often in the run. This relatively simple concept is called *Büchi* acceptance.)

In the non-deterministic variant of a word automaton, we are dealing with a transition function

$$\Delta : A \times C \to \mathcal{P}A.$$

Runs of such machines are no longer uniquely determined: a *run* of $\mathbb{A}$ on an infinite word $\gamma = c_0c_1c_2\ldots$ may be any ω-word $\rho = a_0a_1a_2\ldots$ over A satisfying $a_0 = a_I$ and $a_{i+1} \in \Delta(a_i, c_i)$ for all $i \in \omega$. $\mathbb{A}$ *accepts* γ if at least one of these runs meets the acceptance condition. A good way to envisage this is to think of the automaton traversing γ and at each time *choosing* a new state a_{i+1} from the set $\Delta(a_i, c_i)$.

It is completely straightforward to generalize these notions from infinite words over C to arbitrary Id_C-coalgebras. For instance, in the non-deterministic variant, a *run* of $\mathbb{A}$ on a Id_C-coalgebra $\mathbb{S} = (S, \gamma, \sigma)$ starting at $s \in S$ is an ω-word $\rho = a_0a_1a_2\ldots$ such that $a_0 = a_I$ and $a_{i+1} \in \Delta(a_i, \gamma(\sigma^n(s)))$. $\mathbb{A}$ *accepts* $(\mathbb{S}, s)$ if one of these runs is accepting.

Automata on binary trees

Changing the type of the coalgebra functor, we move on to automata that operate on C-labeled binary trees (or on arbitrary structures that can be represented as coalgebras for the functor $(\mathsf{Id} \times \mathsf{Id})_\Sigma$). The basic new idea here is that an automaton which scans such a structure, starting at the root of the tree, at each node *splits* into two copies, each of which continues the investigation of the tree at one of the two successors of the current node. Formally, we denote a binary tree as a structure $\mathbb{T} = (2^*, \gamma)$, where 2^* denotes the set of finite words over the alphabet $2 = \{0, 1\}$ and $\gamma : 2^* \to C$ is the coloring of the tree. A (deterministic) tree automaton $\mathbb{A} = (A, a_I, \delta, Acc)$ has a transition function

$$\delta : A \times C \to A \times A.$$

A *run* of such an automaton on a C-labeled binary tree $\mathbb{T} = (2^*, \gamma)$ is now an A-labeled binary tree $\rho = (2^*, L : 2^* \to A)$ such that for all nodes $s \in 2^*$, $(L(s0), L(s1)) = \delta(L(s), \gamma(s))$. The automaton is defined to *accept* the tree β if *each* path of the run ρ, seen as an infinite word over A, belongs to A.

Deterministic tree automata, just like deterministic word automata, have a unique run on each input tree. And similarly as for word automata, we obtain a non-deterministic variant by taking, as the transition map of the automaton, a function

$$\Delta : A \times C \to \mathcal{P}(A \times A).$$

Acceptance for such automata may again be formulated in terms of the existence of an accepting run, where an accepting run is now defined as an A-labeled binary tree ρ satisfying $(L(s0), L(s1)) \in \delta(L(s), \gamma(s))$ for every node $s \in 2^*$. It is more convenient however, to rephrase the definition of acceptance within the framework of *game theory*. The combination of the existential ('for some run ...') and universal quantification ('for all paths ...') can be explained quite naturally in terms of the interaction of two players.

With any tree automaton $\mathbb{A}$ and tree $\mathbb{T}$ we associate an *acceptance game* $\mathcal{G}(\mathbb{A}, \mathbb{T})$, which has two players, $\exists$ (Eloise) and $\forall$ (Abélard). For an intuitive understanding of this game, think of $\exists$ as aiming for the automaton to accept the tree, and of $\forall$ as trying to prevent this. Basically, a position of the game is a pair $(a, s) \in A \times 2^*$, which codes the situation of the automaton being in state a, inspecting node s of the tree. In such a position, $\exists$ chooses a pair $(a_0, a_1) \in \Delta(a, \gamma(s))$, after which $\forall$ chooses to move either left or right, thus determining the next node of the tree to be either $s0$ or $s1$, and the next state of the automaton to be either a_0 or a_1. Any full match of the game thus provides an infinite sequence $\alpha = a_0 a_1 a_2 \dots$ of states in $\mathbb{A}$ (with $a_0 = a_I$), which in its turn determines the winner of the match: it is $\exists$ if $\alpha \in Acc$, and $\forall$ otherwise. The automaton *accepts* the tree $\mathbb{T}$ in case $\exists$ has a winning strategy for the associated game initiated at the pair (a_I, ϵ), where ϵ is the root of the tree.

Formally, we may represent this game as the graph game (see section 2.2) of which the game positions are given in the following table, and the acceptance condition is given as before.

Position: b	$P(b)$	Admissible moves: $E[b]$
$(a, s) \in A \times 2^*$	$\exists$	$\{(\xi, s) \in \mathcal{P}(A \times A) \times 2^* \mid \xi \in \Delta(a, \gamma(s))\}$
$((a_0, a_1), s) \in \mathcal{P}(A \times A) \times 2^*$	$\forall$	$\{(a_0, s0), (a_1, s1)\}$

This game theoretic perspective on acceptance opens up various ways to *generalize* the notion of a tree automaton. A standard way to do so proceeds as follows. First, read the pair $(a_0, a_1) \in \Delta(a, \gamma(s))$ as a *conjunction* of the statements 'go left, and switch to state a_0', and 'go right, and switch to state a_1'. Abbreviate this as $L{:}a_0 \wedge R{:}a_1$. Similarly, read the set $\Delta(a, c) = \{(a^i_0, a^i_1) \mid i \in I\}$ *disjunctively*, that is, as the formula $\bigvee_i (L{:}a^i_0 \wedge R{:}a^i_1)$. The concept of *alternation* then naturally arises if we allow *arbitrary* conjunctions and disjunctions over the set $LR_A := \{L{:}a, R{:}a \mid a \in A\}$ in the range of the transition function of the automaton. That is, we let the transition function be of the form

$$\Delta : A \times C \to \mathcal{DL}(LR_A),$$

where, for any set X, we let $\mathcal{DL}(X)$ denote the set of (distributive) *lattice terms* over X, that is, the smallest collection of objects that includes X and contains the expressions $\bigwedge P$ and $\bigvee P$ for any finite set P of objects in $\mathcal{DL}(X)$. The board of this graph game looks as follows.

Position: b	$P(b)$	Admissible moves: $E[b]$
$(a, s) \in A \times S$	$-$	$\{\Delta(a, \gamma(s))\}$
$(\bigvee P, s) \in \mathcal{DL}(A \times A) \times S$	$\exists$	$\{(p, s) \in \mathcal{DL}(A \times A) \times S \mid p \in P\}$
$(\bigwedge P, s) \in \mathcal{DL}(A \times A) \times S$	$\forall$	$\{(p, s) \in \mathcal{DL}(A \times A) \times S \mid p \in P\}$
$(L{:}a, s) \in LR_A \times S$	$-$	$\{(a, s0)\}$
$(R{:}a, s) \in LR_A \times S$	$-$	$\{(a, s1)\}$

Note that in the cases where we do not associate a player with a position, the next position of the game is uniquely determined by the current one, and thus it does not matter which player owns this position. Special attention is needed for positions of the form $(\bigvee P, s)$ and $(\bigwedge P, s)$ in case P is the empty set. In a position $(\bigvee \varnothing, s)$, $\exists$ gets stuck since there is no move available for her. Thus, in accordance with the definition of parity games, $\exists$ immediately *looses* the match. Intuitively, this is correct since the disjunction over an empty set of propositions is usually taken to be the falsum formula $\bot$. Likewise, $\forall$ looses any match ending at a position of the form $(\bigwedge \varnothing, s)$, which is in accordance with the convention that $\bigwedge \varnothing$ is equivalent to the formula $\top$. In any case, it is important to note that in the acceptance game for alternating tree automata, full matches may be finite. The winner and looser of an infinite match are provided by the acceptance condition of the automaton. Given an infinite match β, consider the infinite sequence of 'basic' positions $(a_0, s_0)(a_1, s_1)(a_2, s_2)\ldots$ occurring in β. The match β is won by $\exists$ if the induced infinite word $a_0a_1a_2\ldots$ belongs to Acc, and by $\forall$, otherwise.

For our purposes however, it is more convenient to define the notion of an alternating tree automaton in a slightly different (but equivalent) way. We require the transition function to be of the form

$$\Delta : A \times C \to \mathcal{DL}(A \times A),$$

with the board of the acceptance game looking as follows.

Position: b	$P(b)$	Admissible moves: $E[b]$
$(a, s) \in A \times S$	$-$	$\{\Delta(a, \gamma(s))\}$
$(\bigvee P, s) \in \mathcal{DL}(A \times A) \times S$	$\exists$	$\{(p, s) \in \mathcal{DL}(A \times A) \times S \mid p \in P\}$
$(\bigwedge P, s) \in \mathcal{DL}(A \times A) \times S$	$\forall$	$\{(p, s) \in \mathcal{DL}(A \times A) \times S \mid p \in P\}$
$((a_0, a_1), s) \in (A \times A) \times S$	$\forall$	$\{(a_0, s0), (a_1, s1)\}$

In order to see why these two approaches are equivalent, recall that the pair $(a_0, a_1) \in A \times A$ can be represented as the conjunction $L{:}a_0 \wedge R{:}a_1$. In the other direction, the atomic formula $L{:}a$ can be represented by the pair $(a, a_\top)$ where $a_\top$ is a special 'true' state, that is, it has $\Delta(a_\top, c) = \{\varnothing\}$ for all colors c.

It is not hard to show that the distributive laws (between $\bigvee$ and $\bigwedge$) apply to this kind of game, in the sense that replacing a position (φ, s) with (φ', s), in case φ and φ' are propositionally equivalent formulas, makes no essential change to the game. From this it follows that instead of allowing arbitrary formulas as the value of the transition function, we may confine ourselves to formulas in disjunctive normal form. This enables the following set-theoretic, 'logic-free', presentation of alternating tree automata, namely, in which the transition map has the form

$$\Delta : A \times C \to \mathcal{P}\mathcal{P}(\mathsf{F}A),$$

where F denotes the functor $\mathsf{Id} \times \mathsf{Id}$. Under this definition of the automaton we may present the board of the acceptance game as in the table below. Here we have also made the amendments necessary to enable the automaton to operate on arbitrary $(\mathsf{Id} \times \mathsf{Id})_C$-coalgebras.

Position: b	$P(b)$	Admissible moves: $E[b]$
$(a, s) \in A \times S$	$\exists$	$\{(\Xi, s) \in \mathcal{P}(\mathsf{F}A) \times S \mid \Xi \in \Delta(a)\}$
$(\Xi, s) \in \mathcal{P}(\mathsf{F}A) \times S$	$\forall$	$\{(\xi, \tau) \in \mathsf{F}A \times \mathsf{F}S \mid \xi \in \Xi \text{ and } \tau = \sigma(s)\}$
$((a_0, a_1), (s_0, s_1)) \in \mathsf{F}A \times \mathsf{F}S$	$\forall$	$\{(a_0, s_0), (a_1, s_1)\}$

Automata on graphs/amorphous trees

The last kind of objects for which we consider finite automata are the C-colored coalgebras for the power set functor $\mathcal{P}$. It is important to realize that $\mathcal{P}$-coalgebras differ from coalgebras for the functor Id^k (which sends a set X to its k-ary Cartesian power) in one important aspect. In a $\mathcal{P}$-coalgebra, the collection of successors of a point s is *amorphous* in the sense that one does not have explicit access to the individual points of this set. This means that whereas it is completely trivial to modify the definition of binary tree automata to the case of k-ary trees, some new ideas are required to extend the definition to capture automata for amorphous trees.

There is in fact more than one way to go here. Probably the most intuitive solution generalizes the first-mentioned approach towards alternation for tree automata, i.e., the one in which the transition function takes values in the set of lattice expressions over the set LR_A of 'atomic formulas'. Think of the atomic formula $L{:}a$ as a *modal expression* stating that a applies to the L-successor of the current node of the tree. In the case of the power set functor $\mathcal{P}$, without explicit reference to individual successors, one may use the formulas $\Diamond a$ and $\Box a$ as the basic building blocks of the distributive lattice expressions. The meaning of these formulas would then be to send a copy of the automaton, switched to state a, to some successor of the current point of the graph or tree. Here the difference is of course that in the case of a diamond formula $\Diamond a$, this successor is chosen by $\exists$, while it is $\forall$ who chooses the next node in the case of a box formula $\Box a$. Thus the net effect is that in the case of a diamond formula, a *single* copy of the automaton is sent out to one successor of the current point in the tree, whereas in the case of a box formula, a copy of the automaton is moving to *each* successor node.

The perspective on graph automata that we discuss now is equivalent but different. Our approach roughly follows Janin & Walukiewicz [12], but we have streamlined the presentation quite a bit in order to bring out the coalgebraic aspect of the definition more clearly. This facilitates the generalization towards arbitrary coalgebras.

The basic idea of this second approach is to use *sets* of states of $\mathbb{A}$ as 'descriptions' of *sets* of nodes of the colored graph $\mathbb{S}$ under inspection. Such a 'description', say, by a set $B \subseteq A$ of a set $T \subseteq S$, needs to be substantiated by a relation $Z \subseteq A \times S$ which is *full* on B and T, in the sense that for every $b \in B$ there is a $t \in T$ such that $(b, t) \in Z$, and for every $t \in T$ there is a $b \in B$ such that $(b, t) \in Z$.

In the case of a *deterministic* automaton then, we may simply take the transition map to be of the form

$$\delta : A \times C \to \mathcal{P}A,$$

and the idea is that when the automaton, in state $a \in A$, inspects a node $s \in S$, the set $\delta(a, \gamma(s))$ provides a description of the collection $\sigma(s)$ of successors of s. (Recall that we use γ to denote the coloring of the graph.) It is the task of $\exists$ to come up with a full relation $Z \subseteq \delta(a, \gamma(s)) \times \sigma(s)$ to substantiate the claim that $\delta(a, \gamma(s))$ is an adequate description of $\sigma(s)$. After she has chosen such a relation Z, it is $\forall$'s turn to pick a pair $(b, t) \in Z$. The automaton then switches to state b and moves to successor t of s, and the acceptance game continues. Thus, different from the case of bounded trees, here even the acceptance game associated with a deterministic automaton may witness some nontrivial interaction between the two players.

In passing we note that the equivalence of both approaches can be seen quite easily in terms of coalgebraic modal logic. For, our notion of a set $P \subseteq A$ 'describing' the set $\sigma(s)$ of successors of s, can be very succinctly formulated by the formula

$$\nabla P \ := \ \Box \bigvee \{p \mid p \in P\} \wedge \bigwedge \{\Diamond p \mid p \in P\} \tag{2}$$

holding at s. And conversely, the standard modal operators can be expressed using the ∇ operation: $\Diamond\varphi \equiv \nabla\{\top, \varphi\}$ and $\Box\varphi \equiv \nabla\varnothing \vee \nabla\{\varphi\}$. Thus, at least in the cases where we have conjunctions and disjunctions at our disposal, we may freely switch between ∇ on the one hand, and $\Diamond$ and $\Box$ on the other.

Given the above description of the deterministic graph automata, it is straightforward to come up with the definition of its nondeterministic and alternating variants. Concerning the latter, one could define the transition function of an alternating automaton $\mathbb{A}$ to take values in the set $\mathcal{DL}(\mathcal{P}A)$ of lattice expressions over the set $\mathcal{P}A$, but it seems cleaner to take the equivalent set-theoretic formulation that is based on the disjunctive normal form of such expressions. That is, the transition function of an alternating graph automaton has the form

$$\Delta : A \times C \to \mathcal{PPP}A.$$

The triple occurrence of the power set operation may seem rather confusing at first sight. Probably the best way to understand this feature is by recalling that there is one $\mathcal{P}$ for $\exists$, one for $\forall$, and one for the functor. A better way to type this transition function is as

$$\Delta : A \times C \to \mathcal{PP}\mathsf{F}A.$$

Formally, the acceptance game of such an automaton is played on the following graph:

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

As always, the winning conditions of the game are completely determined by the acceptance condition Acc of the automaton. That is, each match of the game induces, in the most obvious way, a sequence of states of the automaton, and (at least, in case we talking about an infinite match), the winner of the match is $\exists$ if this sequence belongs to Acc, and $\forall$ otherwise.

A coalgebraic perspective

Our presentation of graph automata has provided almost all of the ingredients needed for generalizing the definition of an automaton to a general, coalgebraic level. The key observation still to be made is that our fairly vague story of a subset $P \subseteq A$ 'describing' the set $\sigma(s) \in \mathcal{P}S$ is in fact an instance of the coalgebraic notion of *relation lifting* (see section 2.1). More precisely, a relation $Z \subseteq A \times S$ is full on $P \subseteq Q$ and $T \subseteq S$ if and only if the pair $(P, T) \in \mathsf{F}A \times \mathsf{F}S$ belongs to the relation lifting $\overline{\mathsf{F}}Z$.

Also in the cases of word and tree automata, it is relation lifting that determines how a match of the game proceeds. In these cases however, there is no real choice for $\exists$ when it comes to the 'witnessing relation' Z. For instance, given $((a_0, a_1), (s_0, s_1)) \in \mathsf{F}A \times \mathsf{F}S$, where F is the binary tree functor $\mathsf{Id} \times \mathsf{Id}$, a relation $Z \subseteq \mathsf{F}A \times \mathsf{F}S$ satisfies $((a_0, a_1), (s_0, s_1)) \in \overline{\mathsf{F}}Z$ if and only if Z contains both (a_0, s_0) and (a_1, s_1). But since $\exists$ will always choose the witnessing relation Z as small as possible, this means that without loss of generality we may assume that she picks *exactly* the set $\{(a_0, s_0), (a_1, s_1)\}$, and thus effectively, has no choice at all. The reader is invited to check how this is reflected in the definition of the acceptance game of tree automata.

Thus we have arrived at a natural notion of an automaton operating on pointed F_C-coalgebras — at least, for any functor F for which relation lifting 'works'. In the deterministic case, the transition function could be defined to be of the form

$$\delta : A \times C \to \mathsf{F}A.$$

The acceptance game for such an automaton $\mathbb{A}$ operating on an F_C-coalgebra $\mathbb{S} = (S, \gamma : S \to C, \sigma : S \to \mathsf{F}S)$ is given by the following table:

Position: b	$P(b)$	Admissible moves: $E[b]$
$(a, s) \in A \times S$	$\exists$	$\{Z \in \mathcal{P}(A \times S) \mid (\delta(a, \gamma(s)), \sigma(s)) \in \overline{\mathsf{F}}Z\}$
$Z \in \mathcal{P}(A \times S)$	$\forall$	Z

In the acceptance game for the alternating version such an automaton, the players first play, at a position $(a, s) \in A \times S$, a little 'subgame' in order to arrive at a position $(\alpha, \sigma(s)) \in \mathsf{F}A \times \mathsf{F}S$. From there, play proceeds as in the deterministic version. In general, it is interesting to observe that the alternating game proceeds in *rounds*, and that each round witnesses two fairly different kinds of interaction between $\exists$ and $\forall$.

Finally, it turns out that we can simplify our discussion somewhat by disposing of the colors. In the following section, we develop a framework of F-automata as devices for inspecting coalgebras based on an arbitrary functor, rather than colored coalgebras only. This enables us to work with transition functions that are of the form

$$\Delta : A \to \mathcal{P}\mathcal{P}\mathsf{F}A.$$

Obviously, this theory applies to functors of the form F_C as well, and, hence, it does provide us with a notion of an F_C-automaton that will operate on C-colored F-coalgebras, see section 4.6 for more details.

4 Coalgebra Automata

4.1 Basic definition

The following definition concerns the most important notion of the paper: F-automata.

Definition 4.1 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 *transition function*, and $Acc \subseteq A^\omega$ the *acceptance condition*.

An F-automaton is called *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 non-deterministic). $\lhd$

The meaning of this definition should become clear below when we discuss the acceptance games. In the sequel we may drop the adjective 'alternating' when referring to such an automaton: in our terminology, the generic automaton is alternating, and deterministic and non-deterministic automata are special instances of alternating ones.

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 $\mathit{Inf}(\alpha)$. For instance, a *Büchi* condition puts $\alpha \in Acc$ if and only if $\mathit{Inf}(\alpha)$ contains at least one of a set of special acceptance states. In the remainder of this paper we will work exclusively with *parity* automata.

Definition 4.2 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. $\lhd$

4.2 Acceptance game

F-automata are supposed to operate on pointed F-coalgebras. Basically, the idea is that an 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 briefly: *graph game*, see section 2. It is useful to first consider the following coalgebraic example of a graph game.

Example 4.3 There are various ways to put the notion of bisimulation into this game-theoretic framework. At this stage it is instructive to consider the following approach from Baltag [3].

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 $Z \subseteq S \times S'$ satisfying $(\sigma(s), \sigma'(s')) \in \overline{\mathsf{F}}Z$;
- 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

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

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 4.4 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

$$\begin{array}{lcl} B_\exists & := & A \times S \quad\quad\;\; \cup \;\; \mathsf{F}A \times \mathsf{F}S \\ B_\forall & := & \mathcal{P}(\mathsf{F}A) \times S \;\; \cup \;\; \mathcal{P}(A \times S). \end{array}$$

(For the sake of a smooth presentation, we will occasionally represent a position of the form $(\xi, \sigma(s)) \in \mathsf{F}A \times \mathsf{F}S$ as $(\xi, s) \in \mathsf{F}A \times S$.)

The edge relation E and the parity map $\overline{\Omega}$ of the game 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 $\mathcal{G}(\mathbb{A}, \mathbb{S})$. $\lhd$

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 $\Delta(a) = \{\{\delta(a)\}\}$ for each $a \in A$. 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 4.3: $\exists$ chooses a relation $Z \subseteq A \times S$ such that $(\delta(a), \sigma(s)) \in \overline{\mathsf{F}}Z$, 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. Note however, that much of the power of automata working on infinite objects precisely stems from the intricacies of these 'decorations'.

Now take the more general case in which we only know that $\mathbb{A}$ is *non-deterministic*, 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 $(\alpha,\sigma(s)) \in \mathsf{F}A\times\mathsf{F}S$. Effectively then, at position (a,s) it is $\exists$ on her own who determines the later position $(\alpha,\sigma(s)) \in \mathsf{F}A \times \mathsf{F}S$. Note that at positions of the form $(\alpha,\sigma(s)) \in \mathsf{F}A \times \mathsf{F}S$ 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 $(\alpha,\sigma(s)) \in \mathsf{F}A\times\mathsf{F}S$, 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. Note that any match of the game is over as soon as the responsible player gets stuck, in the sense that (s)he reaches a position in which no moves are admissible. This happens for instance in a position (a,s) such that $\Delta(a,s) = \varnothing$; in this case $\exists$ gets stuck and immediately looses the match. Likewise, if $\exists$ can choose $\varnothing \in \Delta(a,s)$ then she wins the match since $\forall$ will get stuck in the next move.

For future use as a measure of the complexity of an F-automaton, we now define the *index* of an automaton, and some auxiliary notions. Recall that for an element $\xi \in \mathsf{F}A$, where A is finite, we define $Base(\xi)$ as the smallest subset $A_0 \subseteq A$ such that $\xi \in \mathsf{F}A_0$.

Definition 4.5 Given an R-standard set functor F, let $\mathbb{A} = (A, a_I, \Delta, \Omega)$ be some F-automaton. The relation $\to_{\mathbb{A}}$ is defined by putting $a \to_{\mathbb{A}} b$ if $b \in Base(\xi)$ for some $\xi \in \mathsf{F}A$ that occurs in $\Delta(a)$. $\lhd$

It will be intuitively clear that in an acceptance game $\mathcal{G}(\mathbb{A},\mathbb{S})$, at a position of the form $(\xi,\sigma(s)) \in \mathsf{F}A \times \mathsf{F}S$, $\exists$ will choose the relation $Z \subseteq A \times S$ as small as possible. But the set $Base(\xi)$ has the property that if $(\xi,\sigma(s)) \in \overline{\mathsf{F}}Z$, then $(\xi,\sigma(s))$ already belongs to the set $\overline{\mathsf{F}}(Z \cap (Base(\xi) \times S))$. Hence we may always assume without loss of generality that $Z \subseteq Base(\xi) \times S$. From this we may conclude that two subsequent basic positions (a,s) and (b,t) in any acceptance game will be such that $a \to_{\mathbb{A}} b$.

Definition 4.6 Let F be some R-standard set functor, and consider an F-automaton $\mathbb{A} = (A, a_I, \Delta, \Omega)$. We will call a subset C of A *strongly connected* if for every c and d in C there is a path $c = c_0 \to_{\mathbb{A}} c_1 \to_{\mathbb{A}} \ldots \to_{\mathbb{A}} c_n = d$ in C. Now the *index* of $\mathbb{A}$ is defined as

$$ind(\mathbb{A}) := \max\{\#(\Omega[C]) \mid C \text{ a strongly connected component of } \mathbb{A}\},$$

that is, $ind(\mathbb{A})$ is the maximal number of distinct parities reached by the elements of a strongly connected component of $\mathbb{A}$. $\lhd$

4.3 Some basic results

In order to obtain some familiarity with the notion of an F-automaton, let us start with discussing two basic results. First, we prove that acceptance is bisimilation invariant.

Proposition 4.7 *Let* F *be some* R*-standard endofunctor on* Set*, let* $\mathbb{A}$ *be some* F*-automaton, and let* $\mathbb{S} = (S, \sigma)$ *and* $\mathbb{S}' = (S', \sigma')$ *be two* F*-coalgebras. Then for any pair of bisimilar states* s *in* $\mathbb{S}$ *and* s' *in* $\mathbb{S}'$*,* $\mathbb{A}$ *accepts* $(\mathbb{S}, s)$ *iff it accepts* $(\mathbb{S}', s')$*.*

Proof. Let $B \subseteq S \times S'$ be a bisimulation containing the pair $(s_0, s'_0) \in S \times S'$, and assume that $\mathbb{A} = (A, a_I, \Delta, \Omega)$ accepts $(\mathbb{S}, s_0)$. We will prove that $\mathbb{A}$ then also accepts $(\mathbb{S}', s'_0)$.

To see why this is so, consider a match of the game $\mathcal{G}' = \mathcal{G}(\mathbb{A}, \mathbb{S}')$ that starts at the position (a_I, s'_0). Since by assumption, the position (a_I, s_0) in the game $\mathcal{G} = \mathcal{G}(\mathbb{A}, \mathbb{S})$ is winning for $\exists$, it holds in particular that $\Delta(a) \neq \varnothing$ — otherwise, $\exists$ would have lost immediately. Suppose that in $\mathcal{G}$, $\exists$ would choose the position $(\Xi, s) \in \mathcal{P}(\mathsf{F}A) \times S$, then in $\mathcal{G}'$ her choice at position (a_I, s'_0) will be (Ξ, s'_0).

Now if we have $\Xi = \varnothing$, then she wins the match immediately, in which case we are done. So suppose otherwise, then $\forall$ chooses an element $\xi \in \Xi$, so that the next position in the $\mathcal{G}'$-match is (ξ, s'_0). Now suppose that in the corresponding $\mathcal{G}$-position (ξ, s_0), $\exists$'s move would be some relation $Z \subseteq A \times S$. We claim that in the corresponding $\mathcal{G}'$-match, the relation $Z' := Z \circ B$ is a legitimate move for $\exists$ at (ξ, σ'_0), that is, we have $(\xi, \sigma'(s'_0)) \in \overline{\mathsf{F}}Z'$. To see why this is so, observe that it follows from the legitimacy of Z as a move in $\mathcal{G}$ at (ξ, s_0), that $(\xi, \sigma(s_0)) \in \overline{\mathsf{F}}Z$, and from the fact that B is a bisimulation with $(s_0, s'_0) \in B$, that $(\sigma(s_0), \sigma'(s'_0)) \in \overline{\mathsf{F}}B$. But then we have $(\xi, \sigma'_0) \in \overline{\mathsf{F}}Z \circ \overline{\mathsf{F}}B = \overline{\mathsf{F}}(Z \circ B) = \overline{\mathsf{F}}Z'$, as required.

To finish the first round of the game, suppose that at the position Z' of $\mathcal{G}'$, $\forall$ picks a pair $(a_1, s'_1) \in Z'$. Then by definition, there is a point s_1 in $\mathbb{S}$ such that $(a_1, s_1) \in Z$ and $(s_1, s'_1) \in B$.

Continuing in this fashion, we see that $\exists$ obtains her strategy in $\mathcal{G}'$ from playing a *shadow match* of $\mathcal{G}$. The relation between any match β' of $\mathcal{G}'$ in which $\exists$ follows this strategy, and the corresponding shadow match β of $\mathcal{G}$, is that at every stage k of the match, the k-th positions p_k of $\mathcal{G}$ and p'_k of $\mathcal{G}'$ satisfy one of the following conditions:

1. p_k is of the form (x, s) and p'_k is of the form (x, s'), with $(s, s') \in B$ (and x belonging to either A, $\mathcal{P}(\mathsf{F}a)$ or $\mathsf{F}A$), or

2. p_k is some relation $Z \subseteq A \times S$, and p'_k is the relation $Z \circ B$.

But then it follows immediately from the definitions that the winners of the two matches must be identical. And so, from our assumption that $\mathbb{A}$ accepts $(\mathbb{S}, s_0)$, it follows that it also accepts $(\mathbb{S}', s'_0)$. QED

In fact, we may prove something of a converse to this result, at least for *finite* coalgebras. (Our result could be extended to arbitrary coalgebras if we would allow automata to be infinite.) That is, with every finite pointed F-coalgebra $(\mathbb{S}, s)$ we may associate an automaton $\mathbb{A}_{\mathbb{S},s}$ that *characterizes* $(\mathbb{S}, s)$ modulo bisimulation. This automaton is in fact the structure $(\mathbb{S}, s)$ itself, dressed up as a deterministic F-automaton (cf. the discussion following Definition 4.4).

Definition 4.8 Let F be some R-standard endofunctor on Set. Given a finite F-coalgebra $\mathbb{S} = (S, \sigma)$, and a point s in $\mathbb{S}$, define the F-automaton $\mathbb{A}_{\mathbb{S},s}$ as the structure $\mathbb{A}_{\mathbb{S},s} = (S, s, \Delta_\sigma, \Omega_0)$, where Δ_σ and Ω_0 are given by putting $\Delta_\sigma(t) := \{\{\sigma(t)\}\}$, and $\Omega_0(t) := 0$, for every point $t \in S$. ◁

Proposition 4.9 *Let* F *be some R-standard endofunctor on* Set*, and let* $(\mathbb{S}, s)$ *be some finite pointed* F*-coalgebra. Then for any pointed coalgebra* $(\mathbb{S}', s')$*, it holds that*

$$\mathbb{A}_{\mathbb{S},s} \textit{ accepts } (\mathbb{S}', s') \textit{ iff } \mathbb{S}, s \leftrightarrows \mathbb{S}', s'.$$

Proof. It is immediate from the definitions that for every pointed coalgebra $(\mathbb{S}', s')$, the evaluation game $\mathcal{G}(\mathbb{A}_{\mathbb{S},s}, \mathbb{S}')$, initialized at (s, s'), is essentially the same as Baltag's bisimulation game, see Example 4.3. From this, the proposition follows immediately. QED

4.4 Variation: chromatic F-automata

Familiar automata, such as the ones discussed in section 3, operate on coalgebras that are *colored* by some set C, and have a transition function Δ taking input from the set $A \times C$ (with A the state set of the automaton). Now obviously, our definition 4.1, when applied to a functor of the form F_C, does provide automata that will scan C-colored F-coalgebras, but the reader may worry that their transition function $\Delta : A \to \mathcal{P}\mathcal{P}(C \times \mathsf{F}A)$ has the wrong shape since it seems to take input only from A. In this section we will show that the notion of an F-automaton is flexible enough to encode the technicalities involving colors, so that we may work in the simpler framework without making concessions to its scope of applicability.

Let us first introduce a coalgebraic generalization of the notion of automaton that seems to be more in line with standard usage in automata theory. As before, we represent F_C-coalgebras as triples of the form $\mathbb{S} = (S, \gamma, \sigma)$ with $\gamma : S \to C$ and $\sigma : S \to \mathsf{F}S$.

Definition 4.10 Let F be an R-standard endofunctor on Set. A *chromatic* F*-automaton over* C is a quintuple $\mathbb{A} = (A, a_I, C, \Delta, \Omega)$ such that $\Delta : A \times C \to \mathcal{P}\mathcal{P}\mathsf{F}A$ (and A, a_I, and Ω are as before).

Given such an automaton and an F_C-coalgebra $\mathbb{S} = (S, \gamma, \sigma)$, 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: $E[b]$	$\overline{\Omega}(b)$
$(a, s) \in A \times S$	$\exists$	$\{(\Xi, s) \in \mathcal{P}(\mathsf{F}A) \times S \mid \Xi \in \Delta(a, \gamma(s))\}$	$\Omega(a)$
$(\Xi, s) \in \mathcal{P}(\mathsf{F}A) \times S$	$\forall$	$\{(\xi, \tau) \in \mathsf{F}A \times \mathsf{F}S \mid \xi \in \Xi \text{ and } \tau = \sigma(s)\}$	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

◁

We will now show that the differences between the two kinds of automata for recognizing C-colored F-coalgebras are only superficial. That is, we will provide very simple constructions for transforming F_C-automata into equivalent chromatic F-automata over C, and vice versa.

Definition 4.11 Let $\mathbb{A} = (A, a_I, C, \Delta, \Omega)$ be a chromatic F-automaton over C. We define its F_C-*companion* $\mathbb{A}^C$ as the automaton $(A, a_I, \Delta^C, \Omega)$, with $\Delta^C : A \to \mathcal{P}\mathcal{P}(C \times \mathsf{F}A)$ given by

$$\Delta^C(a) := \{\{c\} \times \Xi \mid \Xi \in \Delta(q, c), c \in C\}.$$

(Note that if $\Xi = \{\xi_1, \ldots, \xi_n\}$, then $\{c\} \times \Xi$ is the set $\{(c, \xi_1), \ldots, (c, \xi_n)\}$).

Conversely, given an F_C-automaton $\mathbb{B} = (B, b_I, \Theta, \Psi)$, the structure $(B, b_I, C, \Theta_C, \Psi)$ with

$$\Theta_C(b, c) := \{\Xi \in \mathcal{P}\mathsf{F}B \mid \{c\} \times \Xi \in \Theta(b)\}.$$

is called the *chromatic* F-*companion* of $\mathbb{B}$, notation: $\mathbb{B}_C$. ◁

The following claim shows that the two kinds of automata for F_C-coalgebras are merely variants of one another.

Proposition 4.12 *Let $\mathbb{A}$ be a chromatic F-automaton over C, and $\mathbb{B}$ an F_C-automaton. Then for any pointed F_C-coalgebra $(\mathbb{S}, s)$ it holds that*

$$\mathbb{A} \textit{ accepts } (\mathbb{S}, s) \quad \textit{iff} \quad \mathbb{A}^C \textit{ accepts } (\mathbb{S}, s), \tag{3}$$

$$\mathbb{B} \textit{ accepts } (\mathbb{S}, s) \quad \textit{iff} \quad \mathbb{B}_C \textit{ accepts } (\mathbb{S}, s). \tag{4}$$

Proof. We confine our attention to (4). Fix $\mathbb{B}$ and $\mathbb{S}$. We will show that

$$Win_\exists(\mathcal{G}(\mathbb{B}, \mathbb{S})) = Win_\exists(\mathcal{G}_C(\mathbb{B}_C, \mathbb{S})), \tag{5}$$

which clearly suffices to prove the equivalence of $\mathbb{B}$ and $\mathbb{B}_C$.

For the inclusion $\supseteq$ of (5), note that by Fact 2.5 we may assume that in $\mathcal{G}_C(\mathbb{B}_C, \mathbb{S})$, $\exists$ has a history free strategy f which is winning from every position in $Win_\exists(\mathcal{G}_C(\mathbb{B}_C, \mathbb{S}))$. Now define the following map f' on $\exists$'s positions in the other game, $\mathcal{G}(\mathbb{B}, \mathbb{S})$. For $(b, s) \in B \times S$, let $f^-(b, s)$ denote the unique element $\Xi \in \mathsf{F}A$ such that $f(b, s) = (\Xi, s)$, and put

$$\begin{aligned} f'(b, s) &:= \big(\{g(s)\} \times f^-(b, s), s\big), \\ f'(\xi, \tau) &:= f(\xi, \tau). \end{aligned}$$

We first show that f' is a legitimate strategy for $\exists$ on each of her positions in the set $Win_\exists(\mathcal{G}_C(\mathbb{B}_C, \mathbb{S}))$. This is obvious for positions of the form $(\xi, \tau) \in \mathsf{F}B \times \mathsf{F}S$. For a position $(b, s) \in B \times S$, from the fact that $f(b, s) \in \Theta_C(b, \gamma(s))$ and the definition of Θ_C, it is immediate that $f'(b, s)$ belongs to $\Theta(b)$, as required.

Now consider an arbitrary match β of $\mathcal{G}(\mathbb{B}, \mathbb{S})$, initiated at a position p that we know to be winning in the game $\mathcal{G}_C(\mathbb{B}_C, \mathbb{S})$, and assume that $\exists$ plays according to her strategy f'. It is not hard to see that with β we may associate a *shadow match* β' of $\mathcal{G}_C(\mathbb{B}_C, \mathbb{S})$ in which $\exists$ plays according to her winning strategy f, and such that β and β' pass through exactly the same basic positions (i.e., in $A \times S$). This immediately implies that $\exists$ wins β, and so her strategy f' must be winning for her in $\mathcal{G}(\mathbb{B}, \mathbb{S})$. This proves the inclusion $\supseteq$ of (5).

For the other inclusion, let f be a history free winning strategy for $\exists$ in $\mathcal{G}(\mathbb{B}, \mathbb{S})$. The key observation is that for any position $(b, s) \in Win_\exists(\mathcal{G}(\mathbb{B}, \mathbb{S}))$, $f(b, s)$ must be of the form

$(\{c\} \times \Xi, s)$ for some $c \in C$ and $\Xi \in \mathsf{F}B$. From this observation it is completely straightforward to define a strategy for $\exists$ in the other game, $\mathcal{G}_C(\mathbb{B}_C, \mathbb{S})$, and to prove, in analogy of the proof just given, that this strategy is winning from every position in $Win_\exists(\mathcal{G}(\mathbb{B}, \mathbb{S}))$.

In order to prove the key observation, assume that $f(b, s) = (\Pi, s) \in \mathcal{P}(C \times \mathsf{F}B) \times S$, and suppose for contradiction that Π contains elements (c_1, ξ_1) and (c_2, ξ_2) with $c_1 \neq c_2$. The point is that this would always enable $\forall$ to choose, in the next move, a pair $((c_i, \xi_i), s)$ such that c_i is *distinct* from $\gamma(s)$, and thus provide him with an immediate win of the match. Hence we arrive at the desired contradiction, since we assumed that f was a winning strategy on (b, s).

This justifies our key observation, and hence, we are done with the proof of (5). QED

4.5 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 $\mathsf{F}A$. This suggests the following generalization. Recall that, given a set X, $\mathcal{DL}(X)$ denotes the set of lattice expressions over X.

Definition 4.13 Let F be an R-standard endofunctor on Set. A *logical* F*-automaton* is a quadruple $\mathbb{A} = (A, a_I, \Delta, \Omega)$ with A, a_I and Ω 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 $\alpha \in \mathsf{F}A$. Play then continues as in the game for the standard automaton.

Position: b	$P(b)$	Admissible moves: $E[b]$	$\overline{\Omega}(b)$
$(a, s) \in A \times S$	–	$\{\Delta(a)\}$	$\Omega(a)$
$(\bigvee P, s) \in \mathcal{DL}(\mathsf{F}A) \times S$	$\exists$	$\{(p, s) \in \mathcal{DL}(A \cup \mathsf{F}A) \times S \mid p \in P\}$	0
$(\bigwedge P, s) \in \mathcal{DL}(\mathsf{F}A) \times S$	$\forall$	$\{(p, s) \in \mathcal{DL}(A \cup \mathsf{F}A) \times S \mid p \in P\}$	0
$(\xi, s) \in \mathsf{F}A \times S$	$\exists$	$\{Z \in \mathcal{P}(A \times S) \mid (\xi, \sigma(s)) \in \overline{\mathsf{F}}Z\}$	0
$Z \in \mathcal{P}(A \times S)$	$\forall$	Z	0

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

Proposition 4.14 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, see for instance MULLER & SCHUPP [19, Appendix C]. (Basically, it just involves applying the distributive laws of disjunction over conjunction, and vice versa).

4.6 Variation: delayed F-automata

The automata of Definition 4.1 all have the property that in the acceptance game, the play basically switches from positions in $A \times S$ to ones in $\mathsf{F}A \times \mathsf{F}S$, perhaps with some alternation between $\bigvee/\exists$ and $\bigwedge/\forall$. For many purposes this is rather restrictive; it would be more convenient to allow moves from a position $(a, s) \in A \times S$ to another position $(b, s) \in A \times S$ without making a coalgebraic move in between. That is, while the automaton switches state, it would stay in the same point of the coalgebra. We will call such automata *delayed* because the move to successors of the coalgebra point is delayed. Just as in the case of ordinary automata, we can define the transition function of a delayed automaton as a map of the form $\Delta : A \to \mathcal{PP}(A \cup \mathsf{F}A)$, or we can choose the equivalent 'logical' format. For our purposes, the latter formulation is more convenient.

Definition 4.15 Let F be an R-standard endofunctor on Set. A *delayed* F*-automaton* is a quadruple $\mathbb{A} = (A, a_I, \Delta, \Omega)$, with A, a_I and Ω as usual, and transition function $\Delta : A \to \mathcal{DL}(A \cup \mathsf{F}A)$. $\lhd$

The *acceptance game* $\mathcal{G}(\mathbb{A}, \mathbb{S})$ associated with $\mathbb{A}$ and an F-coalgebra $\mathbb{S}$ is the parity graph game given by the following table. (For the sake of a smoother notation, positions of the form $(\xi, \sigma(s)) \in \mathsf{F}A \times \mathsf{F}S$ are given as $(\xi, s) \in \mathsf{F}A \times S$.)

Position: b	$P(b)$	Admissible moves: $E[b]$	$\overline{\Omega}(b)$
$(a, s) \in A \times S$	–	$\{\Delta(a)\}$	$\Omega(a)$
$(\bigvee P, s) \in \mathcal{DL}(A \cup \mathsf{F}A) \times S$	$\exists$	$\{(p, s) \in \mathcal{DL}(A \cup \mathsf{F}A) \times S \mid p \in P\}$	0
$(\bigwedge P, s) \in \mathcal{DL}(A \cup \mathsf{F}A) \times S$	$\forall$	$\{(p, s) \in \mathcal{DL}(A \cup \mathsf{F}A) \times S \mid p \in P\}$	0
$(\xi, s) \in \mathsf{F}A \times S$	$\exists$	$\{Z \in \mathcal{P}(A \times S) \mid (\xi, \sigma(s)) \in \overline{\mathsf{F}}Z\}$	0
$Z \in \mathcal{P}(A \times S)$	$\forall$	Z	0

Once more, it can be shown that the new type of F-automata has the same expressive power as the old one.

Proposition 4.16 *Delayed* F*-automata recognize the same classes of coalgebras as ordinary* F*-automata.*

In order to prove this proposition, we first need some definitions.

Definition 4.17 Let $\mathbb{A} = (A, a_I, \Delta, \Omega)$ be a delayed F-automaton. We define the relation $\lhd \subseteq A \times A$ by putting $b \lhd a$ if and only if b occurs in $\Delta(a)$ (that is, b is an atomic subterm of $\Delta(a)$). We call a state $a \in A$ *semi-guarded* if $\Omega(b) > \Omega(a)$ whenever $b \lhd a$, and guarded if there is no b such that $b \lhd a$. An automaton is called *semi-guarded* (*guarded*, respectively) if each of its states is semi-guarded (guarded). $\lhd$

We will now first prove that any delayed automaton can be turned into an equivalent semi-guarded one, and then show that any semi-guarded automaton can be replaced with an equivalent guarded one. This suffices to prove the proposition, because clearly, the ordinary F-automata can be identified with the guarded delayed ones.

Proposition 4.18 *Every delayed* F*-automata* $\mathbb{A}$ *is equivalent to a semiguarded* F*-automaton.*

Proof. Fix the delayed automaton $\mathbb{A} = (A, a_I, \Delta, \Omega)$. Without loss of generality, we may assume that Ω is injective. By induction we will show that for all $i \geq -1$ we may find an automaton $\mathbb{A}_i = (A, a_I, \Delta_i, \Omega)$ which is equivalent to $\mathbb{A}$ and satisfies

$$\Omega(b) > \min(\Omega(a), i) \text{ for all } a \in A \text{ and for all } b \lhd a. \tag{6}$$

Clearly then the proposition is proved once i takes the value of the maximum parity of all states in A.

Since the base case of the induction ($i = -1$) is immediate by the definition of parity functions, we move on to the inductive case, for $i+1$. By the inductive hypothesis then, we have that $\Omega(b) > \min(\Omega(a), i)$ for all $a \in A$ and for all $b \lhd a$. Now distinguish cases.

If there is *no* $a \in A$ such that $\Omega(a) = i+1$, we simply put $\mathbb{A}_{i+1} := \mathbb{A}_i$. In this case $\mathbb{A}_{i+1}$ and $\mathbb{A}_i$ are trivially equivalent, so by the induction hypothesis, $\mathbb{A}_{i+1}$ is equivalent to $\mathbb{A}_i$. It remains to be proved that $\mathbb{A}_{i+1}$ meets the required constraint. Take states a and b in A such that $b \lhd a$, and distinguish further cases. In case $\Omega(a) < i+1$, then we have $\min(\Omega(a), i) = \Omega(a)$, so we may read the induction hypothesis as stating that $\Omega(b) > \Omega(a)$; hence, we see that $\Omega(b) > \min(\Omega(a), i+1)$, as required. In the case that $\Omega(a) \geq i+1$, it follows from the assumption that in fact, $\Omega(a) > i+1$. Now the induction hypothesis implies that $\Omega(b) > i$, and since $\Omega(b) \neq i+1$, this gives that $\Omega(b) > i+1 = \min(\Omega(a), i+1)$, again, as required.

Now suppose that, on the other hand, $i+1$ *does* belong to the range of Ω. We only consider the case that $i+1$ is odd — the case that it is even can be treated in a similar fashion. By our assumption on Ω there is in fact a *unique* state $c \in A$ such that $\Omega(c) = i+1$. Define, for any natural number j, $A_j := \{a \in A \mid \Omega(a) \geq j\}$, then it easily follows from the induction hypothesis that $\Delta_i(c) \in \mathcal{DL}(A_{i+1} \cup \mathsf{F}A)$. Due to the validity of the distributive laws in this context, without loss of generality we may assume that $\Delta_i(c)$ is of the form $(c \vee \delta_1) \wedge \delta_2$, where c does not appear in δ_1 or δ_2.

Let $\theta := \delta_1 \wedge \delta_2$, and let, for any $\varphi \in \mathcal{DL}(A \cup \mathsf{F}A)$, $\varphi[\theta/c]$ denote the result of uniformly substituting θ for c in the lattice term φ (elements from $\mathsf{F}A$ remain untouched under this operation). Now define Δ_{i+1} as follows:

$$\Delta_{i+1}(a) := \begin{cases} \Delta_i(a) & \text{if } \Omega(a) < i+1, \\ \theta & \text{if } \Omega(a) = i+1 \text{ (i.e., } a = c), \\ \Delta_i(a)[\theta/c] & \text{if } \Omega(a) > i+1. \end{cases}$$

From the fact that $\Delta_i(c) \in \mathcal{DL}(A_{i+1} \cup \mathsf{F}A)$, and the assumption that c does not occur in δ_1 and δ_2, it follows that $\theta \in \mathcal{DL}(A_{i+2} \cup \mathsf{F}A)$. Using the induction hypothesis, it is straighforward to derive from this that Δ_{i+1} satisfies (6) for $i+1$, whence $\mathbb{A}_{i+1}$ is at least of the right format.

It is thus left to prove that $\mathbb{A}_{i+1}$ is equivalent to $\mathbb{A}$, so by the induction hypothesis it suffices to show that $\mathbb{A}_{i+1}$ is equivalent to $\mathbb{A}_i$. Fix some F-coalgebra $\mathbb{S} = (S, \sigma)$, then we must show, for all points $s_0 \in S$, that

$$(a_I, s_0) \in \mathit{Win}_\exists(\mathcal{G}(\mathbb{A}_i, \mathbb{S})) \text{ iff } (a_I, s_0) \in \mathit{Win}_\exists(\mathcal{G}(\mathbb{A}_{i+1}, \mathbb{S})). \tag{7}$$

For the direction ($\Leftarrow$) of (7), let f be a history free winning strategy for $\exists$ in $\mathcal{G}_{i+1} := \mathcal{G}(\mathbb{A}_{i+1}, \mathbb{S})$. In order to define a strategy f' for her in $\mathcal{G}_i := \mathcal{G}(\mathbb{A}_i, \mathbb{S})$, suppose that play of $\mathcal{G}_i$ has arrived at a position (φ, s) with $\varphi \in \mathcal{DL}(A \cup \mathsf{F}A)$. Let (a, s) be the last basic position that we passed in this match, and distinguish cases:

If $\Omega(a) \leq i + 1$ and $\varphi \neq c \vee \delta_1$, then it is easy to see that (φ, s) must be a position of the game $\mathcal{G}_{i+1}$ as well, so that we may simply define $f'(\varphi, s) := f(\varphi, s)$. In this case we say that (φ, s) is its own corresponding position.

If $\Omega(a) = i + 1$ and $\varphi = c \vee \delta_1$, then $\exists$ chooses $f'(c \vee \delta_1, s) := (\delta_1, s)$. In this case, (φ, s) has no corresponding position.

If $\Omega(a) > i + 1$, then the pair $(\varphi[\theta/c], s)$ must be a position of $\mathcal{G}_{i+1}$; since we are dealing with positions for $\exists$, φ must be a disjunction. Now define $f'(\varphi, s) := (\psi[\theta/c], s)$, where ψ is the disjunct of φ given by $f(\varphi, s) = (\psi, s)$. Also, call $(\varphi[\theta/c], s)$ the corresponding position, in $\mathcal{G}_{i+1}$, of (φ, s).

Now consider an arbitrary match β of $\mathcal{G}_i$, starting from (a_I, s_0), and such that $\exists$ plays according to the strategy described above. It is easy to see that if we (i) replace every $\mathcal{G}_i$-position in β with its corresponding $\mathcal{G}_{i+1}$-position, (ii) erase all positions of the form $(c \vee \delta_1, s)$, and (iii) leave positions of the form $Z \subseteq A \times S$ untouched, then we obtain an f-conform match β' of $\mathcal{G}_{i+1}$. It follows by the assumption on f that β' is won by $\exists$.

Now let k be the highest parity occurring infinitely often in β. If $k < i + 1$, then from a certain moment on, the matches β and β' are *identical*; clearly then, β is also won by $\exists$. If $k > i+1$, then it is not hard to see that k must also be the highest priority occurring infinitely often in β', so that again, $\exists$ is the winner of β. Now suppose for contradiction that $k = i + 1$. Since $\exists$ never chooses c in a position of the form $(c \vee \delta_1, s)$, we may infer that positions of the form (c, s) occur infinitely often in β because of being chosen by $\forall$ as elements of a position $Z_k \subseteq A \times S$, and that among all positions for which this holds, these are the ones with the highest parity. But then by definition this must apply to the match β' as well. From this it is easy to derive that $i + 1$ is the highest parity occurring infinitely often in β'. This provides the desired contradiction with the assumption on f, and thus proves that $k \neq i + 1$.

It follows that in all cases, $\exists$ wins β, and since β was arbitrary, we have proved that (a_I, s_0) is a winning position for $\exists$. This proves the direction ($\Leftarrow$) of (7); we omit the proof for the other direction, which is similar. QED

Proposition 4.19 *Every semiguarded* F*-automata is equivalent to a guarded* F*-automaton.*

Proof. Let $\mathbb{A} = (A, a_I, \Delta, \Omega)$ be a delayed F-automaton that is semiguarded. For the definition of its guarded equivalent $\mathbb{A}'$, we need some preparations.

For each state $a \in A$, we will construct, in finitely many steps, a tree $T(a)$, together with a *labelling* and a (partial) *marking* of the nodes of the tree. To set up the construction, we start with the construction tree $C(a)$ of the $\mathcal{DL}(A \cup \mathsf{F}A)$-term $\Delta(a)$. The inner nodes of this tree are *labelled* with a connective ($\bigvee$ or $\bigwedge$), and the leaves with an *atomic* term of the form $b \in A$ or $\xi \in \mathsf{F}A$. (Nodes labelled with terms $\bigvee$ or $\bigwedge$ will be considered as inner nodes also if they have no children.) Furthermore, the root of this tree is *marked* 'a', while no other node of the initial tree is marked. Now recursively, replace each leaf labelled $b \in A$ with its construction tree $C(b)$, and use 'b' to mark the new inner node of the tree. Repeat the process until no leaves are left that are labelled with elements of A.

It is not difficult to see that this process must terminate after finitely many steps. The key observation here is that for any two states $a, b \in A$, we find b as the label of some leaf of $C(A)$ if and only if $b \lhd a$. And since $\mathbb{A}$ is semiguarded we have $\Omega(b) > \Omega(a)$ if $b \lhd a$. From this it follows that there are no infinite sequences $a_0 \rhd a_1 \rhd a_2 \rhd \ldots$, and so the algorithm must terminate.

We define $T(a)$ as the tree that is constructed by the algorithm that we just described. Clearly, $T(a)$ can be seen as the construction tree of some $\mathcal{DL}(\mathsf{F}A)$ term $\delta(a)$. Let $Sq_{\lhd}(\mathbb{A})$ be the set of all nonempty sequences $a_0 \ldots a_n$ such that $a_n \lhd a_{n-1} \lhd \ldots \lhd a_0$, then with each leaf l of the final tree we may associate, apart from its label $\xi_l \in \mathsf{F}A$, also a unique sequence $\alpha_l \in Sq_{\lhd}(\mathbb{A})$, consisting of the sequence of marks encountered on the path leading from the root to the leaf l.

We are ready for the definition of $\mathbb{A}'$. To start with, for its state set we take the set $A' := Sq_{\lhd}(\mathbb{A}) \times A$. It follows from earlier observations by König's Lemma that $Sq_{\lhd}(\mathbb{A})$ is finite, and so A' is finite as well. (We could obtain a smaller guarded equivalent for $\mathbb{A}$, but size considerations are not our worry here.) For any $\alpha \in Sq_{\lhd}(\mathbb{A})$, we let $\rho_\alpha : A \to A'$ denote the map given by $\rho_\alpha(a) := (\alpha, a)$. Now let $T'(a)$ be the tree obtained from $T(a)$ by replacing, for each leaf l, the label $\xi_l \in \mathsf{F}A$ with the object $(\mathsf{F}\rho_{\alpha_l})(\xi_l) \in \mathsf{F}A'$, and let $\delta(a)$ denote the $\mathcal{DL}(\mathsf{F}A')$ term associated with $T'(a)$. Then we define the transition map Δ' of $\mathbb{A}'$ by

$$\Delta'(\alpha, a) := \delta(a),$$

while its parity map Ω' is given by

$$\Omega'(a_1 \ldots a_n, a) := \max\{\Omega(a_i) \mid 1 \leq i \leq n\}.$$

(In fact, it follows from the definitions that $\Omega'(a_1 \ldots a_n, a) = \Omega(a_n)$, but this does not play a role in the sequel.) Finally, we define

$$\mathbb{A}' := (A', (a_I, a_I), \Delta', \Omega').$$

It follows immediately from the definitions that $\mathbb{A}'$ is a guarded automaton, so it is left to show that $\mathbb{A}$ and $\mathbb{A}'$ are equivalent. For this purpose, fix an arbitrary F-coalgebra $\mathbb{S} = (S, \sigma)$. We will prove that

$$\text{for all } s \in S,\ \mathbb{A} \text{ accepts } (\mathbb{S}, s) \text{ only if } \mathbb{A}' \text{ accepts } (\mathbb{S}, s). \tag{8}$$

(We omit the similar proof that, conversely, $\mathbb{A}$ accepts every pointed coalgebras that $\mathbb{A}'$ accepts.) For simplicity we assume that all games are infinite; games that might end after finitely many rounds can be taken care of by a suitable case distinction.

Before we turn to the proof of (8), first consider a basic position $((\alpha, a), s) \in A' \times S$ of the game $\mathcal{G}' = \mathcal{G}(\mathbb{A}', \mathbb{S})$. Define the *static* game $\mathcal{G}'_{((\alpha,a),s)}$ as the (part of) the game $\mathcal{G}'$ that starts at $((\alpha, a), s)$ and ends when a position $(\xi', s) \in \mathsf{F}A' \times S$ is reached. A similar definition applies to the static game $\mathcal{G}_{(a,s)}$. The crucial observation concerning these games is that the earlier defined tree $T(a)$ represents the game tree of *both* these static games, in the sense that the nodes of $T(a)$ labelled with a disjunction (conjunction, respectively) represent choice nodes of $\exists$ ($\forall$, respectively) and that leaves of $T(a)$ represent positions that mark the end of the static games, etc. Now a *strategy* of $\exists$ in either $\mathcal{G}'_{((\alpha,a),s)}$ or $\mathcal{G}_{(a,s)}$ corresponds to a partial function that maps each disjunctive node of $T(a)$ to one of its children, and a similar correspondence applies to strategies of $\forall$. This provides an obvious and direct way to mimick any strategy, for either player, and in either static game, by a strategy for the same player, in the other static game.

Now we consider one round of the game $\mathcal{G}'$, starting from the basic position $((\alpha, a), s) \in A' \times S$. Let f be an arbitrary history free strategy for $\exists$ in $\mathcal{G}$, and let f' be the strategy in $\mathcal{G}'_{((\alpha,a),s)}$ that mimicks (the relevant part of) f. Then with any match of $\mathcal{G}'_{((\alpha,a),s)}$ that is conform f', we may associate a *shadow match* of $\mathcal{G}_{(s,a)}$ that is conform f. The point is that if the $\mathcal{G}'_{((\alpha,a),s)}$ match ends at a position $((\mathsf{F}\rho_{aa_1\ldots a_n})(\xi), s) \in \mathsf{F}A' \times S$, then the shadow match ends at the position $(\xi, s) \in \mathsf{F}A \times S$. (In the tree $T(a)$, both matches correspond to a path passing the inner nodes marked a, a_1, …, a_n.)

Suppose that in $\mathcal{G}$, $\exists$'s move at the position (ξ, s) is the binary relation $Z \subseteq A \times S$. It follows from the legitimacy of Z that $(\xi, \sigma(s)) \in \overline{\mathsf{F}}Z$. Now define the relation $Z' \subseteq A' \times S$ as follows:

$$Z' := \{((\beta, b), t) \in A' \times S \mid (b, t) \in Z\},$$

where β abbreviates the sequence $\beta = aa_1 \ldots a_n$. Putting it differently, we have $Z' = (Gr(\rho_\beta))^{\smile} \circ Z$. Using Fact 2.1, we may derive that $\overline{\mathsf{F}}Z' = \overline{\mathsf{F}}(Gr(\rho_\beta))^{\smile} \circ \overline{\mathsf{F}}Z = (Gr(\mathsf{F}\rho_\beta))^{\smile} \circ \overline{\mathsf{F}}Z$. In particular, it follows from $(\xi, \sigma(s)) \in \overline{\mathsf{F}}Z$ that $((\mathsf{F}\rho_\beta)(\xi), \sigma(s)) \in \overline{\mathsf{F}}Z'$. In other words, Z' is a *legitimate* move for $\exists$ at the position $((\mathsf{F}\rho_\beta)(\xi), s)$. If she chooses, indeed, Z' as her next move, $\exists$ guarantees that any element of Z' that $\forall$ picks must be of the form $((aa_1 \ldots a_n, b), t)$. That is, the history $a, a_1, \ldots, a_n$ has been coded up in the next basic position $((aa_1 \ldots a_n, b), t)$.

Summarizing, where one round of the $\mathcal{G}'$-match moves from one basic position $((\alpha, a), s)$ to a next one of the form $((aa_1 \ldots a_n, b), t)$, the shadow match in $\mathcal{G}$ started from the basic position (a, s), passed the basic positions (a_1, s), …, (a_n, s), and arrived at the basic position (b, t).

Now assume that $\exists$ plays an entire match of $\mathcal{G}'$, according to the strategy f' just described. Start from the basic position $((a_{0,0}, a_{0,0}), s_0)$ — here we write $a_{0,0}$ for the initial state a_I of $\mathbb{A}$. Suppose that

$$((a_{0,0}, a_{0,0}), s_0)((a_{0,0}a_{0,1} \ldots a_{0,n_0}, a_{1,0}), s_1)((a_{1,0}a_{1,1} \ldots a_{1,n_1}, a_{2,0}), s_2) \ldots \tag{9}$$

is the sequence of basic positions in an arbitrary match in which $\exists$ plays f'. Then it is not hard to see that

$$(a_{0,0}, s_0)(a_{0,1}, s_0) \ldots (a_{0,n_0}, s_0)(a_{1,0}, s_1) \ldots (a_{1,n_1}, s_1)(a_{2,0}, s_2) \ldots \tag{10}$$

is the sequence of basic positions in the associated shadow match in $\mathcal{G}$ starting from $(a_{0,0}, s_0)$. We claim that the two matches are won by the same player.

To see why this is so, let Q' be the set of states from A' that occur infinitely often in (9), and likewise with Q and (10). It is straightforward to verify that Q' can be represented as

$$Q' = \{(b_{i,0}b_{i,1} \ldots b_{i,n_i}, b_{j,0}) \mid i, j \in I\}$$

for some finite set I, so that it is not so hard to derive that

$$Q = \{b_{i,k} \mid i \in I, 0 \leq k \leq n_i\}.$$

Now by definition of Ω' it follows immediately from these characterizations that $\max\{\Omega'(a') \mid a' \in Q'\} = \max\{\Omega(a) \mid a \in Q\}$, which establishes our claim.

This shows that if $\mathbb{A}$ accepts $(\mathbb{S}, s_0)$, then so does $\mathbb{A}'$, and thus finishes the proof of (the direction from right to left of) (8). QED

5 Coalgebraic fixed point logic

We now turn to the second main topic of the paper, coalgebraic fixed point logic. The formalism that we are about to define can be seen as a straightforward extension with fixed point operators of a natural, finitary, version of Moss' coalgebraic logic [17].

5.1 Syntax

Definition 5.1 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 $\mu\mathcal{L}^{\mathsf{F}}_n(X)$ which contains the formula $\nabla\pi$ for each π that belongs to $\mathsf{F}Q$ for some finite $Q \subseteq \mu\mathcal{L}^{\mathsf{F}}_n(X)$ and 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. $\lhd$

The set X in $\mu\mathcal{L}^{\mathsf{F}}(X)$ refers to *all* variables that may occur in the formulas, not just the free ones (to be defined later). 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 5.2 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 [12], 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 $\bigwedge\{\langle a \rangle p \mid p \in \Phi\} \wedge [a] \bigvee\{p \mid p \in \Phi\}$, cf. the ∇-operator from (2). This is of course quite familiar in coalgebraic logic, and it is not difficult to show that the language of Janin & Walukiewicz is in fact expressively equivalent to our coalgebraic fixed point logic for this functor, see Example 5.16 for more details.

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 has 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 $Base(\pi) \subseteq \mu\mathcal{L}_n^{\mathsf{F}}$ such that $\pi \in \mathsf{F}Q'$ (we already mentioned that 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 5.3 We will write $q \trianglelefteq p$ if q is a *subformula* of p. Inductively we define the set $Sfor(p)$ of subformulas of p as follows:

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

where $Base(\pi)$ denotes the smallest set Q such that $\pi \in \mathsf{F}Q$; the elements of $Base(\pi)$ 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 5.4 *Every formula $p \in \mu\mathcal{L}^{\mathsf{F}}$ has finitely many subformulas.*

Definition 5.5 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 fairly standard, and so are the definitions of the sets $FVar(p)$ and $BVar(p)$ of *free* and *bound* variables, respectively, of a formula $p \in \mu\mathcal{L}^{\mathsf{F}}$. (For instance, the inductive clause for ∇ reads $FVar(\nabla\pi) := \bigcup\{FVar(p) \mid p \in Base(\pi)\}$.) 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 $\eta_x x.p_x$, 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. Let $\leq_p \subseteq BVar(p) \times BVar(p)$ denote the relation given by

$$x \leq_p y \text{ if } q_x \trianglelefteq q_y.$$

Clearly, $\leq_p$ is a partial order on $BVar(p)$; it is called the *subformula order* of p. ◁

5.2 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 5.6 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. $\lhd$

Of course, it would be more in style with the coalgebraic paradigm to present an F-model (S, σ, V) over X as a coalgebra for the functor $\mathsf{F}_{\mathcal{P}(X)}$ (cf. Definition 3.1). We follow the present approach because it seems to lend itself better towards the treatment of fixed point operators.

Definition 5.7 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:

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

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

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 $\mathbb{M}$. $\lhd$

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 [17]) 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 guideline 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. Fortunately, since any such π belongs to the set $F(Base(\pi))$, it follows from Proposition 2.4 that the two definitions are equivalent.

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

Definition 5.8 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 $X = \varphi(X)$. $\lhd$

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 $\lambda X \in \mathcal{P}(S).[\![p]\!]_{\mathbb{M}[x\mapsto X]}$, while $[\![\nu x.p]\!]_{\mathbb{M}}$ is the union of the collection of all post-fixed points of this map.

5.3 Basic semantic results

Before we can do anything interesting, there are 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.

Proposition 5.9 (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*

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

Proof. The proof is by induction on the complexity of p. All cases are completely standard, with the possible exception of the case that $p = \nabla\pi$. Inductively we assume that $\Vdash^{V}\upharpoonright_{Base(\pi)} = \Vdash^{V'}\upharpoonright_{Base(\pi)}$, so that $\overline{\mathsf{F}}(\Vdash^{V}\upharpoonright_{Base(\pi)}) = \overline{\mathsf{F}}(\Vdash^{V'}\upharpoonright_{Base(\pi)})$. From this it is immediate by the definition that $S, \sigma, V \Vdash \nabla\pi$ iff $S, \sigma, V' \Vdash \nabla\pi$. QED

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 5.10 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). $\lhd$

Next we turn to the Monotonicity Lemma.

Proposition 5.11 (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). QED

Remark 5.12 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 5.13 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. This yields 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 S 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 formulas in S, then both $\bigwedge_{\alpha<\beta} p_\alpha$ and $\bigvee_{\alpha<\beta} p_\alpha$ belong to S, 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:

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

Using these formulas, one puts

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

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:

$$[\![p]\!]_{\mathbb{M}} = [\![t^\alpha p]\!]_{\mathbb{M}}, \text{ for any } \mathsf{F}\text{-model } \mathbb{M} = (S, \sigma, V) \text{ and any ordinal } \alpha > |S|^+. \tag{11}$$

Note however, that in general, the 'unfolding ordinal' α of (11) 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 5.14 *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 5.13, and the easily established fact that truth of $\mathcal{L}^{\mathsf{F}}_{\infty}$-sentences is a bisimulation invariant property. QED

We are now ready to state our last basic semantic result.

Proposition 5.15 (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'.*

Proof. It is easy to rewrite an arbitrary $\mu\mathcal{L}^{\mathsf{F}}$-formula into a clean equivalent, by consistently renaming bound variables.

The second part of the proposition is proved by a completely standard induction on the complexity of formulas. We confine ourselves to a proof sketch for the case that p is of the form $\mu x.q$.

By the inductive hypothesis we may assume that q is guarded. Hence, if we replace every fixed point subformula $\eta y.r(x, y)$ of q with its unfolding $r(x, \eta y.r(x, y))$, we obtain an equivalent q' of q, in which the only *unguarded* occurrences of x are *outside* the scope of fixed point operators. Then, using laws of classical propositional logic, it is not hard to rewrite $q'(x)$ in an equivalent form $q''(x) = (x \vee r(x)) \wedge s(x)$, where all occurrences of x in $r(x)$ and $s(x)$ are guarded. It can subsequently be shown that $\mu x.(x \vee r(x)) \wedge s(x)$ is equivalent to the formula $\mu x.r(x) \wedge s(x)$. QED

Now that we have explained the syntax and semantics of coalgebraic fixed point logic, we briefly describe how it generalizes the modal μ-calculus.

Example 5.16 Fix a set Prop of atomic propositions and a set Act of atomic actions. We will show that the language of the modal μ-calculus, in the formulation of Janin & Walukiewicz [12] (see Example 5.2) is in fact expressively equivalent to our coalgebraic fixed point logic for the functor $\mathsf{F}S = \mathcal{P}(\mathsf{Prop}) \times \mathcal{P}(S)^{\mathsf{Act}}$.

First, if we consider an arbitrary $\mu\mathcal{L}^{\mathsf{F}}$-formula of the form $\nabla\pi$, it is easy to see that π can be represented as a pair $(P, \{\Phi_a \mid a \in \mathsf{Act}\})$, with $P \subseteq \mathsf{Prop}$ a set of atomic propositions, and each Φ_a a finite set of formulas. Then we may rephrase the formula $\nabla\pi$ in the language of Janin and Walukiewicz as

$$\nabla\pi \;\equiv\; \bigwedge_{q \in P} q \wedge \bigwedge_{q \notin P} \neg q \wedge \bigwedge_{a \in \mathsf{Act}} (a \to \Phi_a).$$

On the basis of this it is straightforward to define a truth invariant translation from $\mu\mathcal{L}^{\mathsf{F}}$-formulas to formulas in the modal μ-calculus.

Conversely, one may show that every coalgebraic fixed point formula for the functor F defined above has an equivalent modal fixed point formula. To see why this is so, first observe that for any action a, the formulas $a \to \varnothing$ and $a \to \{\top\}$ are equivalent to $[a]\bot$ and $\langle a\rangle\top$, respectively. Hence, the formula $\top$ can be represented as the disjunction $(a \to \varnothing) \vee (a \to \{\top\})$. Now for simplicity we assume that Act consists of two elements, a_1 and a_2. It follows that, for instance, the formula $a_1 \to \Phi$ can be rewritten as

$$(a_1 \to \Phi) \;\equiv\; \bigvee_{P \subseteq \mathsf{Prop}} \nabla(P, \{(a_1, \Phi), (a_2, \varnothing)\}) \vee \nabla(P, \{(a_1, \Phi), (a_2, \{\top\})\}).$$

Also, using the fact that any proposition letter q is equivalent to $\bigvee_{q \in P \subseteq \mathsf{Prop}} (\bigwedge_{p \in P} p \wedge \bigwedge_{p \notin P} \neg p)$, it can be shown that

$$q \;\equiv\; \bigvee_{q \in P \subseteq \mathsf{Prop}} \; \bigvee_{\Phi_1 \in \{\varnothing, \{\top\}\}} \; \bigvee_{\Phi_2 \in \{\varnothing, \{\top\}\}} \nabla(P, \{(a_1, \Phi_1), (a_2, \Phi_2)\}).$$

From these two observations it is again completely straightforward to obtain a function mapping modal μ-formulas to equivalent coalgebraic fixed point formulas.

6 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.

6.1 Evaluation games

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

Definition 6.1 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* $\mathcal{E} = \mathcal{E}(q, \mathbb{M})$.
The board of $\mathcal{E}$ is given as the set

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

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. $\lhd$

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.

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,

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

Figure 1: Admissible moves in the evaluation game

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, which only comes out in infinite matches, can be put in the following slogan: all fixed points mean unfolding, and least fixed points mean finite unfolding. In order to make this more precise, we need the following observation.

Proposition 6.2 *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 $\mathcal{E}(q, \mathbb{M})$, the set of variables that are unfolded infinitely often during β contains a maximal member (in the subformula order).*

Proof. Let U be the set of variables that are unfolded infinitely often during β. Since β is an infinite game, and q has only finitely many subformulas, U is non-empty. We claim that U is in fact directed (with respect to the subformula order $\leq_q$). The claim of the Proposition is then immediate by the fact that U is finite.

Suppose for contradiction that x and y are in U while x and y are incomparable with respect to $\leq_q$, that is, neither $q_x \trianglelefteq q_y$ nor $q_y \trianglelefteq q_x$. Since both x and y get unfolded infinitely often during β, the match can never go into one of the formulas, say, q_x, and stay there. But then the only way to get back, from a position inside q_x, to a position where y can be unfolded, is through unfolding a variable z such that both q_x and q_y are subformulas of q_z. Since this must happen infinitely often, one such variable z must be in U. Hence U is directed. QED

Definition 6.3 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* $\mathcal{E} = \mathcal{E}(q, \mathbb{M})$.

Let β be a full match played on the arena of $\mathcal{E}$.

- If β is finite then it is lost by the player who got stuck (and thus, won by their adversary).

- If β is infinite, let x be the highest ranking fixed point variable that got unfolded infinitely often during β. Now β is won by $\exists$ if x is a ν-variable, and by $\forall$ if x is a ν-variable.

$\lhd$

6.2 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 clean* $\mu\mathcal{L}^{\mathsf{F}}$*-formula* q*, any* F*-model* $\mathbb{M} = (S, \sigma, V)$ *and any state* $s \in S$ *it holds that*

$$\mathbb{M}, s \Vdash q \text{ iff } (q, s) \in \mathit{Win}_{\exists}(\mathcal{E}(q, \mathbb{M})). \tag{12}$$

Proof. The proof of this theorem proceeds by induction on the complexity of the formula q. We leave the base step ($p \in \{\top, \bot\}$ or p is a variable), the boolean cases, and the greatest fixed point case of the inductive step as exercises for the reader.

We do treat the inductive case where q is of the form $q = \mu x.q'$. First consider the direction ($\Rightarrow$) of (12). Let W be the set of states $w \in S$ such that (w, q) is a winning position for $\exists$ in $\mathcal{E} = \mathcal{E}(q, \mathbb{M})$. In order to show that $[\![q]\!]_{\mathbb{M}} \subseteq W$ it suffices to prove that W is a prefixed point of the map $\lambda X.[\![q']\!]_{\mathbb{M}[x \mapsto X]}$. Abbreviate $V' := V[x \mapsto X]$ and $\mathbb{M}' := (S, \sigma, V')$, and let $t \in S$ be an arbitrary state in $[\![q']\!]_{\mathbb{M}'}$, that is, $\mathbb{M}', t \Vdash q'$. It suffices to show that $t \in W$; in other words, we have to provide $\exists$ with a winning strategy in $\mathcal{E}(q, \mathbb{M})$ starting from position (q, t).

First, note that it follows inductively from $\mathbb{M}', t \Vdash q'$ that $\exists$ has a winning strategy f' from position (q', t) in the evaluation game $\mathcal{E}' = \mathcal{E}(q', \mathbb{M}')$. Now observe that $\mathcal{E}$ and $\mathcal{E}'$ are in fact *very* similar games: apart from the fact that $\mathcal{E}'$ has no positions of the form (q, u), the only difference between the two games concerns positions of the form (x, u). In $\mathcal{E}'$, x is a *free* variable, so in a position (x, u), the match is over, and the winner of such a match depends on whether u belongs to $V'(x) = W$ or not. In $\mathcal{E}$ on the other hand, x is a *bound* variable, so at a state (x, u), the variable x will get unfolded.

Second, observe that by definition of W, for every state $w \in W$, $\exists$ has a winning strategy f_w for the game $\mathcal{E}$ initialized at (w, q). Note that in this initialized game, the second position invariably will be (w, q'). So $\exists$ could not have spoiled her chances in this first round, and hence f_w is winning for $\exists$ in $\mathcal{E}$ at (w, q') as well.

Now suppose that $\exists$ plays $\mathcal{E}$ from (q, t) according to the following strategy g:

- after the initial move, the pebble is in position (q', t);
- $\exists$ first plays her strategy f';
- as soon as a position (x, u) is reached, distinguish the following two cases:
  1. if $u \in W$ then $\exists$ continues with f_u;
  2. if $u \notin W$ then $\exists$ continues with a random strategy.

We now claim that this strategy g is in fact a winning strategy for $\exists$ in the game $\mathcal{E}$ initialized at (q, t). To see why this must be so, make the following case distinction concerning an arbitrary full play β which is consistent with g:

No state (x, u) is ever reached. This means that β doubles as an $\mathcal{E}$-match and an $\mathcal{E}'$-match. As an $\mathcal{E}'$-match, β is won by $\exists$. Since $\mathcal{E}$ and $\mathcal{E}'$ only differ when it comes to x, this means that β is also a win for $\exists$ in $\mathcal{E}$. Note that it does not matter here whether β is finite or infinite.

At some stage a position (x, u) is reached. In the $\mathcal{E}'$-perspective on π, the match would have reached a final position here. Since f' was a winning strategy for $\exists$, this can only happen if $u \in V'(x) = W$. (In other words, case 2 mentioned above will never occur.) So $\exists$ consequently plays according to f_u; the first position after (x, u) is (q', u). We know that f_u is a winning strategy for $\exists$ in the game $\mathcal{E}$ initialized at (q', u). It is then easy to see that any continuation of the match in which $\exists$ plays f_u, is won by $\exists$.

Altogether this shows that indeed, g is a winning strategy for $\exists$.

We now consider, still for the inductive case in which $q = \mu x.q'$, the direction $(\Leftarrow)$ of (12). Assume that $\exists$ has a winning strategy f in the game $\mathcal{E} = \mathcal{E}(\mu x.q', \mathbb{M})$, and suppose for contradiction that $\mathbb{M}, s \not\Vdash \mu x.q'$. Abbreviating $Q := [\![\mu x.q']\!]_{\mathbb{M}}$, this means that $s \notin Q$.

First consider an arbitrary point $t \notin Q$. It follows from $\mathbb{S}, t \not\Vdash \mu x.q'$ that there is a prefixed point $U \subseteq S$ of the map $\lambda X \subseteq S.[\![q']\!]_{\mathbb{M}[x \mapsto X]}$ to which t does *not* belong. That is, $t \notin U$ while $[\![q']\!]_{\mathbb{M}[x \mapsto U]} \subseteq U$. It follows that $t \notin [\![q']\!]_{\mathbb{M}[x \mapsto U]}$, or, equivalently, that $\mathbb{M}[x \mapsto U], t \not\Vdash q'$. By the inductive hypothesis then, $\exists$ does not have a winning strategy in $\mathcal{E}(q', \mathbb{M}[x \mapsto U])$ from (q', t). But since $Q \subseteq U$ (because U is a prefixed point of the map $\lambda X \subseteq S.[\![q']\!]_{\mathbb{M}[x \mapsto X]}$, and Q is the intersection of all such prefixed points), it easily follows from the rules of the game that $\exists$ does not have a winning strategy in the game $\mathcal{E}' := \mathcal{E}(q', \mathbb{M}[x \mapsto Q])$ from (q', t) either. That is, for each strategy g of $\exists$ starting at (q', t), $\forall$ has a counter strategy $\overline{g}_t$ such that the match of $\mathcal{E}'$ determined by g and $\overline{g}_t$ is won by $\forall$.

Furthermore, observe that because of the resemblance between the games $\mathcal{E}$ and $\mathcal{E}'$, any strategy g of $\exists$ in $\mathcal{E}$, as a map restricted to partial $\mathcal{E}'$-matches, uniquely determines a strategy in $\mathcal{E}'$; this strategy will be denoted as g as well.

Now consider the matches of $\mathcal{E}$, starting at $(\mu x.q', s)$, in which $\exists$ plays according to her supposedly winning strategy f. Suppose that $\forall$ counters the strategy f as follows:

- $\forall$ starts with the strategy $\overline{f}_s$;
- from that moment on, $\forall$ sticks to the current strategy, unless a position (x, u) is reached; now distinguish cases:
  1. if $u \in Q$ then $\forall$ continues with a random strategy;
  2. if $u \notin Q$ then $\forall$ plays as follows. Let β be the match this far (including (x, u)), and let f_β denote the strategy of $\exists$ for the $\mathcal{E}$-game starting at (q', u) given by $f_\beta(\gamma) = f(\beta\gamma)$. Then by our earlier discussion, f_β can be seen as an $\mathcal{E}'$-strategy for matches starting at (q', u), and so $\forall$ may adopt his counter strategy $\overline{(f_\beta)}_u$ from this moment on.

Consider the $\mathcal{E}$-match β starting at $(\mu x.q', s)$ determined by $\exists$ playing her strategy f and $\forall$ using the strategy defined above. First observe that β can pass through positions of the form (x, u) only finitely many times, for otherwise, the μ-variable x would be the highest fixed point variable unfolded infinitely often, contradicting the assumption that f is winning for $\exists$. Second, note that the first case of passing a state (x, u) will never occur; since arriving at a position (x, u) with $u \in Q$ would mean that, contrary to our earlier conclusion, $\exists$ would have a successful strategy in $\mathcal{E}'$ at a point $v \notin Q$ after all.

This means, however, that after a certain initial partial play β, ending in a position (x, u) with $u \notin Q$, $\forall$ will stick to his strategy $\overline{(f_\beta)}_u$, while no further position (v, x) is ever reached. It follows from our assumptions on $\overline{(f_\beta)}_u$ that the match γ resulting from $\exists$ playing f_β against $\forall$ playing $\overline{(f_\beta)}_u$ is winning for $\forall$ in $\mathcal{E}'$, and from this it is not hard to derive that the $\mathcal{E}$-match $\delta = \beta\gamma$ is won by $\forall$. This provides the desired contradiction, since it shows that the strategy f is not winning for $\exists$ after all.

This finishes the inductive case where $q = \mu x.q'$, so we now turn to the case where $q = \nabla\pi$. In order to prove the equivalence of (12), first assume that $\mathbb{M}, s \Vdash \nabla\pi$, and consider the game $\mathcal{E} = \mathcal{E}(q, \mathbb{M})$. Let $\exists$, in the game $\mathcal{E}$ initialized at $(\nabla\pi, s)$, choose the set $Z := \Vdash\restriction_{Base(\pi)}$ as her first move. Note that it follows by definition from $\mathbb{M}, s \Vdash \nabla\pi$ that $(\sigma(s), \pi) \in \Vdash$, so this is a legitimate move. Now suppose that $\forall$ moves at position Z, choosing $(p, t) \in Z$ as the next position. It follows from $(p, t) \in Z$ and the inductive hypothesis that in the game $\mathcal{E}(p, \mathbb{M})$, $\exists$ has a winning strategy starting from (p, t). But then it is easy to see that this strategy will also guarantee her winning $\mathcal{E}(\nabla\pi, \mathbb{M})$ from (p, t). All in all we have provided her with a strategy winning $\mathcal{E}$ from $(\nabla\pi, s)$.

For the other direction of (12), suppose that $\exists$ wins the game $\mathcal{E} = \mathcal{E}(q, \mathbb{M})$ starting at position $(\nabla\pi, s)$. Let's say that her choice at position $(\nabla\pi, s)$ is the set $Z \subseteq Base(\pi) \times S$. Since, at position Z, $\forall$ may choose any $(r, t) \in Z$, we may assume that each such (r, t) is winning for $\exists$ in $\mathcal{E}(r, \mathbb{M})$. It thus follows from the inductive hypothesis that $Z \subseteq \Vdash$, whence we see that $Z \subseteq \Vdash\restriction_{Base(\pi)}$. Hence by monotonicity of $\overline{\mathsf{F}}$ we obtain that $\overline{\mathsf{F}}(Z) \subseteq \overline{\mathsf{F}}(\Vdash\restriction_{Base(\pi)})$. But we know that $(\pi, \sigma(s)) \in Z$, for if not, then Z would have been illegitimate. So we find that $(\pi, \sigma(s)) \in \overline{\mathsf{F}}(\Vdash\restriction_{Base(\pi)})$, precisely what is needed to show that $\mathbb{M}, s \Vdash \nabla\pi$. QED

7 Automata and fixed point formulas

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 theorems below, which show that the F-automata have the same expressive strength as the logical formalism $\mu\mathcal{L}^{\mathsf{F}}$ when it comes to describing pointed F-coalgebras.

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 a parity F-automaton $\mathbb{A}_q$ such that for any pointed F-coalgebra $(\mathbb{S}, s)$:*

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

Proof. By Proposition 5.15 we can assume without loss of generality that p is clean, and by Proposition 4.16, it suffices to construct a logical F-automaton $\mathbb{A}_p$. The structure of $\mathbb{A}_p$ will closely resemble that of the set $\mathit{Sfor}(p)$ of subformulas of p.

In fact, we can *identify* the states of $\mathbb{A}_p$ with the subformulas of p, and let $p := a_I$ be the initial state of $\mathbb{A}_p$. Now define the following transition function Δ on $\mathit{Sfor}(p)$ (where in order to avoid confusion, we use $\sqcup$ and $\sqcap$ to denote the disjunction and conjunction for the automaton):

$$\begin{array}{rcl} \Delta(\bot) & := & \bigvee \varnothing \\ \Delta(\top) & := & \bigwedge \varnothing \\ \Delta(q \vee q') & := & q \sqcup q' \\ \Delta(q \wedge q') & := & q \sqcap q' \\ \Delta(\eta x.q_x) & := & q_x \\ \Delta(\nabla \pi) & := & \pi \\ \Delta(x) & := & q_x. \end{array}$$

With this definition, we have established that for any F-coalgebra $\mathbb{S}$, the *boards* of the acceptance game $\mathcal{G} = \mathcal{G}(\mathbb{A}_p, \mathbb{S})$ and of the evaluation game $\mathcal{E} = \mathcal{E}(p, \mathbb{S})$ are in fact *identical.* Hence in particular, the matches of the two games coincide.

The only thing left is to define a parity function on A that takes proper care of the winning conditions of the evaluation game $\mathcal{E}$. Using the construction tree of the formula p, it is easy to define a function $\Omega : \mathit{Sfor}(p) \to \omega$ such that

- $\Omega(q) = 0$ if $q \notin \mathit{BVar}(p)$,
- $\Omega(x)$ is odd if x is a μ-variable, and even if x is a ν-variable,
- $\Omega(x) \leq \Omega(y)$ if $x \leq_p y$ (i.e., if $\eta_x x.p_x \trianglelefteq \eta_y y.p_y$).

It is then straightforward to verify that $\exists$ is the winner of a match β in $\mathcal{G}$ if and only if she is the winner of β, seen as a match of $\mathcal{E}$. From this it is immediate that $\mathit{Win}_\exists(\mathcal{G}) = \mathit{Win}_\exists(\mathcal{E})$, and hence the theorem follows by the Adequacy Theorem of the game semantics of $\mu\mathcal{L}^{\mathsf{F}}$. QED

Conversely, one can show that, given a parity F-automaton $\mathbb{A}$, one can construct a $\mu\mathcal{L}^{\mathsf{F}}$-formula $q_{\mathbb{A}}$ that holds precisely at those pointed F-coalgebras that are accepted by $\mathbb{A}$.

Theorem 3 (Automata are formulas) *Let F be an R-standard endofunctor on Set. Then any parity F-automaton $\mathbb{A}$ can be transformed into a $\mu\mathcal{L}^{\mathsf{F}}$-sentence $q_{\mathbb{A}}$ such that for any pointed F-coalgebra $(\mathbb{S}, s)$:*

$$\mathbb{A} \textit{ accepts } (\mathbb{S}, s) \textit{ iff } \mathbb{S}, s \Vdash q_{\mathbb{A}}.$$

Proof. Since this result is rather standard (see for instance [7, Theorem 11.6]), we confine ourselves to a sketch of its proof.

As an auxiliary notion we need to adapt the concept of an F-automaton to a device that operates on F-*models.* Given a set X of variables, a *(logical)* F, X*-automaton* is a quadruple

$\mathbb{A} = (A, a_I, \Delta, \Omega)$, where A, a_I and Ω are as before, while the transition map is now a function $\Delta : A \to \mathcal{DL}(X \cup \mathsf{F}A)$. The acceptance game of such a device is like that for ordinary logical automata, with the proviso that a position of the form (x, s) marks an immediate end to the match, the winner being $\exists$ if x is true at s, and $\forall$ if x is false at s.

Observe that this concept generalizes that of ordinary logical F-automata: these can be seen as specimens of the new device over the *empty* set of variables. This means that by Proposition 4.14, we may prove the theorem by establishing the following claim.

Claim For every set X and for every F, X-automaton $\mathbb{A}$, there is a $\mu\mathcal{L}^{\mathsf{F}}$-formula $q_{\mathbb{A}}$ with $FVar(p_{\mathbb{A}}) \subseteq X$, such that for all F, X-models $(\mathbb{S}, V)$, and all points s in $\mathbb{S}$:

$$\mathbb{A} \text{ accepts } (\mathbb{S}, V, s) \text{ iff } \mathbb{S}, V, s \Vdash q_{\mathbb{A}}. \tag{13}$$

The *proof* of this claim proceeds by induction on the *index* of $\mathbb{A}$ (this notion is defined for F, X-automata just like for F-automata). Without loss of generality we may assume that states in $\mathbb{A}$ of maximum parity must belong to some strongly connected component C of $\mathbb{A}$ such that $ind(\mathbb{A}) = \#(\Omega[C])$.

If $ind(\mathbb{A}) = 0$, then there are no infinite $\to_{\mathbb{A}}$-paths. Define the *height* $h(a)$ of a state $a \in A$ as the length k of the longest $\to_{\mathbb{A}}$-path $a \to_{\mathbb{A}} a_1 \to_{\mathbb{A}} \ldots \to_{\mathbb{A}} a_k$ starting at a, and put, for $n \geq 0$, $A_n := \{a \in A \mid h(a) \leq n\}$. Furthermore, for $a \in A$, let $\mathbb{A}_a$ be the automaton (A, a, Δ, Ω), that is, the automaton $\mathbb{A}$ but with a as its initial state.

By a subinduction on the height of a we then prove that there is a (fixed point free) formula p_a of the right shape that characterizes the pointed F, X-models that are accepted by $\mathbb{A}_a$. In the base case of this subinduction, we are dealing with a state a of height zero. It is easy to see that for such a, we have that $\Delta(a) \in \mathcal{DL}(X)$, so that we may put $p_a := \Delta(a)$.

In the induction step of the subinduction, we have $h(a) = n + 1$. Then by the inductive hypothesis we may assume the existence of a total map $p_n : A_n \to \mu\mathcal{L}^{\mathsf{F}}$ assigning to each state $b \in A_n$ its associated formula $p(b)$. Also note that each $\xi \in \mathsf{F}A$ occurring in $\Delta(a)$ must actually belong to $\mathsf{F}A_n$. It is then straightforward to verify that for each such ξ, the object $\nabla(\mathsf{F}p_n)(\xi)$ is a formula in $\mu\mathcal{L}^{\mathsf{F}}(X)$. Now define p_a as the formula we obtain by replacing every $\xi \in \mathsf{F}A$ occurring in the $\mathcal{DL}(X \cup \mathsf{F}A_n)$-term $\Delta(a)$ with the formula $\nabla(\mathsf{F}p_n)(\xi)$. Clearly then, p_a has the right format. It is in fact also straightforward to prove that p_a characterizes the F, X-models that are accepted by $\mathbb{A}_a$; details are left to the reader.

For the inductive case (of the main induction), assume that $ind(\mathbb{A}) > 0$. Let m be the maximum parity of the states of A, and define $M = \{a_1, \ldots, a_k\}$ as the set of states of $\mathbb{A}$ that actually have parity m. Now consider the $\mathsf{F}, X \cup M$-automaton

$$\mathbb{A}_M := (A \setminus M, a_I, \Delta\restriction_{A \setminus M}, \Omega\restriction_{A \setminus M}).$$

That is, we have turned the states of M into (new) *variables*. Furthermore, for $i \in \{1, \ldots, k\}$, let $\mathbb{A}_i = (A \setminus M, a_i, \Delta\restriction_{A \setminus M}, \Omega\restriction_{A \setminus M})$ be the version of $\mathbb{A}_M$ which has a_i as its initial position. It follows from our assumptions on $\mathbb{A}$ and M, that each of these automata has a *smaller* index than $\mathbb{A}$. We may thus apply the inductive hypothesis, which provides fixed point formulas p_M, p_1, ..., p_k, all taking free variables from the set $X \cup M$, and such that for any $\mathsf{F}, X \cup M$-model $(\mathbb{S}, V)$ and any point s in $\mathbb{S}$, we have that $\mathbb{A}_M$ accepts $(\mathbb{S}, V, s)$ iff $\mathbb{S}, V, s \Vdash p_M$, and, for each i, $\mathbb{A}_i$ accepts $(\mathbb{S}, V, s)$ iff $\mathbb{S}, V, s \Vdash p_i$.

Clearly then, for any $\mathsf{F}, X \cup M$-model $(\mathbb{S}, V)$, the k-tuple $\overline{p}$ determines a monotone map $[\![\overline{p}]\!]_{\mathbb{S},V} : (\mathcal{P}(S))^k \to (\mathcal{P}(S))^k$ given by

$$[\![\overline{p}]\!]_{\mathbb{S},V}(T_1, \ldots, T_k) := ([\![p_1]\!]_{\mathbb{S},V[\overline{a} \mapsto \overline{T}]}, \ldots, [\![p_1]\!]_{\mathbb{S},V[\overline{a} \mapsto \overline{T}]}).$$

Here $V[\overline{a} \mapsto \overline{T}]$ denotes the valuation given by $V[\overline{a} \mapsto \overline{T}](a_i) = T_i$ while $V[\overline{a} \mapsto \overline{T}](x) = V(x)$ for all variables $x \in X \setminus M$. It follows from standard fixed point theory (see for instance [7, Theorem 20.12]), that the least and greatest fixed points of this map are given by $\mu\mathcal{L}^{\mathsf{F}}$-formulas. More precisely, there are formulas $p_1^\mu, \ldots, p_k^\mu$ and $p_1^\nu, \ldots, p_k^\nu$, all with free variables in X, such that

$$([\![p_1^\mu]\!]_{\mathbb{S},V}, \ldots, [\![p_k^\mu]\!]_{\mathbb{S},V}) \text{ is the } \textit{least} \text{ fixed point of } [\![\overline{p}]\!]_{\mathbb{S},V}$$

for every F, X-model $(\mathbb{S}, V)$, and likewise for the greatest fixed point.

Now let $q_{\mathbb{A}}$ be the formula

$$q_{\mathbb{A}} := p_M[\overline{p}^\eta / \overline{a}].$$

That is, we uniformly substitute, in p_M, each a_i with the formula p_i^η, where η denotes μ if m is odd, and ν if m is even. The proof that this formula $q_{\mathbb{A}}$ indeed satisfies (13) is fairly similar to the proof of the Adequacy Theorem, whence we omit further details. QED

As a corollary to this Theorem, we mention a result that was first observed by Alexandru Baltag (personal communication).

Corollary 7.1 *Let F be an R-standard set functor, and let $(\mathbb{S}, s)$ be some finite pointed F-coalgebra. Then there is a $\mu\mathcal{L}^{\mathsf{F}}$-formula $q_{\mathbb{S},s}$ such that for any pointed F-coalgebra $(\mathbb{S}', s')$*

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

Proof. Define $q_{\mathbb{S},s}$ as the formula obtained by applying (the algorithm in the proof of) Theorem 3 to the automaton $\mathbb{A}_{\mathbb{S},s}$ of Definition 4.8. Then the result is immediate by Proposition 4.9 and Theorem 3. QED

8 Further research

We believe that our F-automata provide a good notion of an automaton for classifying pointed F-coalgebras. Not only do F-automata generalize the familiar devices operating on words, trees and graphs, but the notion is also stable under a number of natural variations, and it is equivalent to a natural coalgebraic fixed point logic. It therefore seems interesting and useful to develop the theory of F-automata further, 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 we just mention the following.

1. In our opinion, the most interesting line of research is to take a coalgebraic perspective on the study of the recognizing power of automata. The point here is that many familiar theorems concerning the expressivity of automata as mechanisms for recognizing structures, can now be parametrized by the coalgebraic functor type. It is thus a natural problem to find out whether (the analogs of) these theorems hold for coalgebras of arbitrary type F. If so, it might be of interest to find a uniform, coalgebraic proof for the result, and if not, we have arrived at an interesting *property* that an endofunctor on Set could or could not have.

   To be a bit more specific, recall that some of the most important results in automata theory concern the following kinds of questions:

   simplification Given an automaton of a certain type (say, a nondeterministic automaton), can it be transformed into a equivalent automaton of a simpler kind (say, a deterministic one)?

   closure properties Call a class C of pointed F-coalgebras A-*recognizable*, where A is a class of F-automata, if there is some automaton $\mathbb{A}$ in A such that C is the class of pointed F-coalgebras that are accepted by $\mathbb{A}$. Is the collection of A-recognizable classes closed under natural operations such as union, intersection, complementation, projection?

   Such questions can now be formulated as questions about the functor F. The first results in this direction are promising. In [14], Clemens Kupke and I prove that for any R-standard functor F, the class of languages that are recognizable by an arbitrary F-automaton is closed under taking unions, intersections and projections. Our main technical result concerns a construction which transforms a given alternating F-automaton into an equivalent non-deterministic one.

   Many interesting problems remain, however. To mention one specific question: for which functors F can we prove a *Complementation Lemma*? That is, for which functors F can we always find, given a (non-deterministic) F-automaton $\mathbb{A}$, another (non-deterministic) F-automaton $\overline{\mathbb{A}}$, with the property that a pointed F-coalgebra is accepted by $\overline{\mathbb{A}}$ iff it is rejected by $\mathbb{A}$?

2. Our parity F-automata have a coalgebraic shape themselves: the automaton $\mathbb{A} = (A, a_I, \Delta, \Omega)$ can, at least object-wise, 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 [24].

3. 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, one would like to understand what happens if we add negation to the language $\mu\mathcal{L}^{\mathsf{F}}$ discussed in section 5. But also, the relation between our system and fixed point extensions of the coalgebraic modal logics developed in Pattinson [21] would be an interesting object of study.

4. As already mentioned in the introduction, there are earlier studies of automata that are based on categories and functors, see for instance ADÁMEK & TRNKOVÁ [2]. This connection clearly has to be investigated further.

References

[1] S. Abramsky. A Cook's Tour of the Finitary Non-Well-Founded Sets. Invited Lecture at BCTCS 1988.

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

[3] 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.

[4] J. Barwise and L. Moss. *Vicious Circles*, volume 60 of *CSLI Lecture Notes.* CSLI Publications, 1996.

[5] 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.

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

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

[8] E.A. Emerson and C.S. Jutla. Tree automata, mu-calculus and determinacy. In *Proceedings of the 32nd IEEE Symposium on Foundations of Computer Science (FoCS'91)*, pages 368–377. IEEE Computer Society Press, 1991.

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

[10] 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.

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

[12] 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.

[13] D. Janin and I. Walukiewicz. On the expressive completeness of the propositional mu-calculus with respect to monadic second order logic. In *CONCUR'96: Concurrency Theory, 7th International Conference*, volume 1119 of *LNCS*, pages 263–277. Springer, 1996.

[14] C. Kupke and Y. Venema. Closure properties of coalgebra automata. In *Proceedings of the 20th Annual IEEE Symposium on Logic in Computer Science (LICS 2005)*, 2005. To appear.

[15] 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.

[16] D. Martin. Borel determinacy. *Annals of Mathematics*, 102:363–371, 1975.

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

[18] A. Mostowski. Games with forbidden positions. Technical Report 78, Instytut Matematyki, Uniwersytet Gdański, Poland, 1991.

[19] D. Muller and P. Schupp. Simulating alternating tree automata by nondeterministic ones: New results and new proofs of the theorems of Rabin, McNaughton and Safra. *Theoretical Computer Science*, 141:69–107, 1995.

[20] D. Niwiński. Fixed point characterization of infinite behavior of finite-state systems. *Theoretical Computer Science*, 189:1–69, 1997.

[21] 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.

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

[23] 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.

[24] 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.

[25] 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.

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

[27] V. Trnková. Relational automata in a category and theory of languages. In M. Karpinski, editor, *Fundamentals of computation theory: proceedings of the 1977 international FCT-conference*, volume 56 of *LNCS*, pages 340–355. Springer, 1977.

[28] Y. Venema. Automata and fixed point logics for coalgebras. *Electronic Notes in Theoretical Computer Science*, 106:355–375, 2004.