EXPTIME Tableaux for the Coalgebraic μ-Calculus*

Corina Cîrstea[1], Clemens Kupke[2], and Dirk Pattinson[2]

[1] School of Electronics and Computer Science, University of Southampton
[2] Department of Computing, Imperial College London

Abstract. The coalgebraic approach to modal logic provides a uniform framework that captures the semantics of a large class of structurally different modal logics, including e.g. graded and probabilistic modal logics and coalition logic. In this paper, we introduce the coalgebraic μ-calculus, an extension of the general (coalgebraic) framework with fixpoint operators. Our main results are completeness of the associated tableau calculus and EXPTIME decidability. Technically, this is achieved by reducing satisfiability to the existence of non-wellfounded tableaux, which is in turn equivalent to the existence of winning strategies in parity games. Our results are parametric in the underlying class of models and yield, as concrete applications, previously unknown complexity bounds for the probabilistic μ-calculus and for an extension of coalition logic with fixpoints.

1 Introduction

The extension of a modal logic with operators for least and greatest fixpoints leads to a dramatic increase in expressive power [1]. The paradigmatic example is of course the modal μ-calculus [10]. In the same way that the μ-calculus extends the modal logic K, one can freely add fixpoint operators to any propositional modal logic, as long as modal operators are monotone. Semantically, this poses no problems, and the interpretation of fixpoint formulas can be defined in a standard way in terms of the semantics of the underlying modal logic.

This apparent simplicity is lost once we move from semantics to syntax: completeness and complexity even of the modal μ-calculus are all but trivial [20, 4], and μ-calculi arising from other monotone modal logics are largely unstudied, with the notable exception of the graded μ-calculus [12]. Here, we improve on this situation, not by providing a new complexity result for a specific fixpoint logic, but by providing a generic and uniform treatment of modal fixpoint logics on the basis of *coalgebraic semantics*. This allows for a generic and uniform treatment of a large class of modal logics and replaces the investigation of a concretely given logic with the study of *coherence conditions* that mediate between the axiomatisation and the (coalgebraic) semantics. The use of coalgebras conveniently abstracts the details of a concretely given class of models, which is replaced by the class of coalgebras for a(n unspecified) endofunctor on sets. Specific choices for this endofunctor then yield specific model classes, such as the class of all Kripke frames or probabilistic transition systems. A property such as completeness or complexity of a specific logic is then automatic once the coherence conditions are satisfied. As it turns out, even *the same* coherence conditions that guarantee completeness

* Partially supported by grant EP/F031173/1 from the UK EPSRC.

and decidability of the underlying modal logic entail the same properties of the ensuing μ-calculus. This immediately provides us with a number of concrete examples: as instances of the generic framework, we obtain not only the known EXPTIME bounds, both for the modal and the graded μ-calculus [4, 12], but also previously unknown EXPTIME bounds for the probabilistic and monotone μ-calculus, and for an extension of coalition logic [15] with fixpoint operators.

Our main technical results are a syntactical characterisation of satisfiability in terms of (non-)existence of closed tableaux and a game-theoretic characterisation of satisfiability that yields an EXPTIME upper bound for the satisfiability problem. Along the way, we establish a small model theorem. We start by describing a parity game that characterizes model checking for the coalgebraic μ-calculus. As in the model-checking game for the modal μ-calculus (see e.g. [18]), we allow greatest and least fixpoints to be unfolded ad libitum. Truth of a formula in a particular state of a model then follows, if only greatest fixpoints are unfolded infinitely often on the top level along infinite paths. This condition can be captured by a parity condition. The same technique is employed in the construction of tableaux, which we conceptualise as finite directed graphs: closed tableaux witness unsatisfiability of the root formula, provided that along any infinite tableau path one can construct an infinite sequence of formulas (a "trace") that violates the parity condition. In particular, closed tableaux are finitely represented proofs of the unsatisfiability of the root formula. Soundness of the tableau calculus is established by showing that a winning strategy in the model checking game precludes existence of a closed tableau. An EXPTIME upper bound for decidability is then established with the help of tableau games, where the adversary chooses a tableau rule, and the player claiming satisfiability chooses one conclusion which effectively constructs a path in a tableau. In order to turn this tableau game into a parity game we combine the game board with the transition function of a deterministic parity word automaton. This automaton checks that on any given play, i.e., on any tableau path, there exists no trace that violates the parity condition. We prove adequacy of the tableau game by constructing a satisfying model from a winning strategy in the tableau game, which makes crucial use of the coherence conditions between the axiomatisation and the coalgebraic semantics. This allows us to determine satisfiability of a fixpoint formula by deciding the associated (parity) tableau game, and the announced EXPTIME upper bound follows once we can ensure that legality of moves in the tableau game can be decided in exponential time.

Related Work. Our treatment is inspired by [14, 19, 17], but we note some important differences. In contrast to [14], we use parity games that directly correspond to tableaux, together with parity automata to detect bad traces. Moreover, owing to the generality of the coalgebraic framework, the model construction here needs to superimpose a coalgebra structure on the relation induced by a winning strategy. This construction is necessarily different from [17], since we cannot argue in terms of modal rank in the presence of fixpoints. Coalgebraic fixpoint logics are also treated in [19], where an automata theoretic characterisation of satisfiability is presented. We add to this picture by providing complexity results and a complete tableau calculus. Moreover, we use standard syntax for modal operators, which allows us to subsume for instance the graded μ-calculus that cannot be expressed in terms of the ∇-operator used in *op.cit.*

2 The Coalgebraic μ-Calculus

To keep our treatment fully parametric in the underlying (modal) logic, we define the syntax of the coalgebraic μ-calculus relative to a (fixed) modal similarity type, that is, a set Λ of modal operators with associated arities. Throughout, we fix a denumerable set V of propositional variables. We will only deal with formulas in negation normal form and abbreviate $\overline{\Lambda} = \{\overline{\heartsuit} \mid \heartsuit \in \Lambda\}$ and $\overline{\mathsf{V}} = \{\overline{p} \mid p \in \mathsf{V}\}$. The arity of $\overline{\heartsuit} \in \overline{\Lambda}$ is the same as that of $\heartsuit$. The set $\mathcal{F}(\Lambda)$ of Λ-formulas is given by the grammar

$$A, B ::= p \mid \overline{p} \mid A \vee B \mid A \wedge B \mid \heartsuit(A_1, \dots, A_n) \mid \mu p.A \mid \nu p.A$$

where $p \in \mathsf{V}$, $\heartsuit \in \Lambda \cup \overline{\Lambda}$ is n-ary and $\overline{p}$ does not occur in A in the last two clauses. The sets of free and bound variables of a formula are defined as usual, in particular p is bound in $\mu p.A$ and $\nu p.A$. Negation $\overline{\cdot} : \mathcal{F}(\Lambda) \to \mathcal{F}(\Lambda)$ is given inductively by $\overline{\overline{p}} = p$, $\overline{A \wedge B} = \overline{A} \vee \overline{B}$, $\overline{\heartsuit(A_1, \dots, A_n)} = \overline{\heartsuit}(\overline{A_1}, \dots, \overline{A_n})$ and $\overline{\mu p.A} = \nu p.\overline{A}[\overline{p} := p]$ and the dual clauses for $\vee$ and ν. If S is a set of formulas, then the collection of formulas that arises by prefixing elements of S by one layer of modalities is denoted by $(\Lambda \cup \overline{\Lambda})(S) = \{\heartsuit(S_1, \dots, S_n) \mid \heartsuit \in \Lambda \cup \overline{\Lambda}\ n\text{-ary}, S_1, \dots, S_n \in S\}$. A *substitution* is a mapping $\sigma : V \to \mathcal{F}(\Lambda)$ and $A\sigma$ is the result of replacing all free occurrences of $p \in V$ in A by $\sigma(p)$.

On the semantical side, parametricity is achieved by adopting coalgebraic semantics: formulas are interpreted over T-coalgebras, where T is an (unspecified) endofunctor on sets, and we recover the semantics of a large number of logics in the form of specific choices for T. To interpret the modal operators $\heartsuit \in \Lambda$, we require that T extends to a *Λ-structure* and comes with a predicate lifting, that is, a natural transformation of type $[\![\heartsuit]\!] : 2^n \to 2 \circ T^{\mathrm{op}}$ for every n-ary modality $\heartsuit \in \Lambda$, where $2 : \mathsf{Set} \to \mathsf{Set}^{\mathrm{op}}$ is the contravariant powerset functor. In elementary terms, this amounts to assigning a set-indexed family of functions $([\![\heartsuit]\!]_X : \mathcal{P}(X)^n \to \mathcal{P}(TX))_{X \in \mathsf{Set}}$ to every n-ary modal operator $\heartsuit \in \Lambda$ such that $(Tf)^{-1} \circ [\![\heartsuit]\!]_X(A_1, \dots, A_n) = [\![\heartsuit]\!]_Y(f^{-1}(A_1), \dots, f^{-1}(A_n))$ for all functions $f : Y \to X$. If $\heartsuit \in \Lambda$ is n-ary, we put $[\![\overline{\heartsuit}]\!]_X(A_1, \dots, A_n) = (TX) \setminus [\![\heartsuit]\!]_X(X \setminus A_1, \dots, X \setminus A_n)$. We usually denote a structure just by the endofunctor T and leave the definition of the predicate liftings implicit. A Λ-structure is *monotone* if, for all sets X we have that $[\![\heartsuit]\!]_X(A_1, \dots, A_n) \subseteq [\![\heartsuit]\!]_X(B_1, \dots, B_n)$ whenever $A_i \subseteq B_i$ for all $i = 1, \dots, n$.

In the coalgebraic approach, the role of frames is played by *T-coalgebras*, i.e. pairs (C, γ) where C is a (state) set and $\gamma : C \to TC$ is a (transition) function. A *T-model* is a triple (C, γ, σ) where (C, γ) is a T-coalgebra and $\sigma : V \to \mathcal{P}(C)$ is a valuation (we put $\sigma(\overline{p}) = C \setminus \sigma(p)$). For a monotone T structure and a T-model $M = (C, \gamma, \sigma)$, the *truth set* $[\![A]\!]_M$ of a formula $A \in \mathcal{F}(\Lambda)$ w.r.t. M is given inductively by

$$[\![p]\!]_M = \sigma(p) \quad [\![\overline{p}]\!]_M = C \setminus \sigma(p) \quad [\![\mu p.A]\!]_M = \mathsf{LFP}(A_p^M) \quad [\![\nu p.A]\!]_M = \mathsf{GFP}(A_p^M)$$

$$[\![\heartsuit(A_1, \dots, A_n)]\!]_M = \gamma^{-1} \circ [\![\heartsuit]\!]_C([\![A_1]\!]_M, \dots, [\![A_n]\!]_M)$$

where $\mathsf{LFP}(A_p^M)$ and $\mathsf{GFP}(A_p^M)$ are the least and greatest fixpoint of the monotone mapping $A_p^M : \mathcal{P}(C) \to \mathcal{P}(C)$ defined by $A_p^M(X) = [\![A]\!]_{(C,\gamma,\sigma')}$ with $\sigma'(q) = \sigma(q)$ for $q \neq p$ and $\sigma'(p) = X$. We write $M, c \models A$ if $c \in [\![A]\!]_M$ to denote that A is satisfied at c. A formula $A \in \mathcal{F}(\Lambda)$ is *satisfiable* w.r.t. a given Λ-structure T if there exists a T-model M such that $[\![A]\!]_M \neq \emptyset$. The mappings A_p^M are indeed monotone in case of a monotone Λ-structure, which guarantees the existence of fixpoints.

Example 1. 1. T-coalgebras $(C, \gamma : C \to \mathcal{P}(C))$ for $TX = \mathcal{P}(X)$ are Kripke frames. If $\Lambda = \{\Box\}$ for $\Box$ unary and $\overline{\Box} = \Diamond$, $\mathcal{F}(\Lambda)$ are the formulas of the modal μ-calculus [10], and the structure $[\![\Box]\!]_X(A) = \{B \in \mathcal{P}(X) \mid B \subseteq A\}$ gives its semantics.

2. The syntax of the graded μ-calculus [12] is given (modulo an index shift) by the similarity type $\Lambda = \{\langle n \rangle \mid n \geq 0\}$ where $\overline{\langle n \rangle} = [n]$, and $\langle n \rangle A$ reads as "A holds in more than n successors". In contrast to *op. cit.* we interpret the graded μ-calculus over multigraphs, i.e. coalgebras for the functor B (below) that extends to a structure via

$$\mathsf{B}(X) = \{f : X \to \mathbb{N} \mid \mathsf{supp}(f) \text{ finite}\} \quad [\![\langle n \rangle]\!]_X(A) = \{f \in \mathsf{B}(X) \mid \sum_{x \in X} f(x) > n\}.$$

where $\mathsf{supp}(f) = \{x \in X \mid f(x) \neq 0\}$ is the support of f. Note that this semantics differs from the Kripke semantics for both graded modal logic [7] and the graded μ-calculus, but both types of semantics induce the same satisfiability problem: Kripke frames are multigraphs where each edge has multiplicity one, and the unravelling of a multigraph can be turned into a Kripke frame by inserting the appropriate number of copies of each state. Both transformations preserve satisfiability.

3. The probabilistic μ-calculus arises from the similarity type $\Lambda = \{\langle p \rangle \mid p \in [0,1] \cap \mathbb{Q}\}$ where $\overline{\langle p \rangle} = [p]$ and $\langle p \rangle \phi$ reads as "ϕ holds with probability at least p in the next state". The semantics of the probabilistic μ-calculus is given by the structure

$$\mathsf{D}(X) = \{\mu : X \to_f [0,1] \mid \sum_{x \in X} \mu(x) = 1\} \quad [\![\langle p \rangle]\!]_X(A) = \{\mu \in \mathsf{D}(X) \mid \sum_{x \in A} \mu(x) \geq p\}$$

where $\to_f$ indicates maps with finite support. Coalgebras for D are precisely image-finite Markov chains, and the finite model property of the coalgebraic μ-calculus that we establish later ensures that satisfiability is independent of image-finite semantics.

4. Formulas of coalition logic over a finite set N of agents [15] arise via $\Lambda = \{[C] \mid C \subseteq N\}$, and are interpreted over game frames, i.e. coalgebras for the functor

$$\mathsf{G}(X) = \{(f, (S_i)_{i \in N}) \mid \prod_{i \in N} S_i \neq \emptyset, f : \prod_{i \in N} S_i \to X\}$$

which is a class-valued functor, which however fits with the subsequent development. We think of S_i as the set of strategies for agent i and f is an outcome function. We read $[C]A$ reads as "coalition C can achieve A", which is captured by the lifting

$$[\![[C]]\!]_X(A) = \{(f, (S_i)_{i \in N}) \in \mathsf{G}(X) \mid \exists (s_i)_{i \in C} \forall (s_i)_{i \in N \setminus C} (f((s_i)_{i \in N}) \in A\}$$

that induces the standard semantics of coalition logic.

5. Finally, the similarity type $\Lambda = \{\Box\}$ of monotone modal logic [2] has a single unary $\Box$ (we write $\overline{\Box} = \Diamond$) and interpret the ensuing language over monotone neighbourhood frames, that is, coalgebras for the functor / structure

$$\mathcal{M}(X) = \{Y \subseteq \mathcal{P}(\mathcal{P}(X)) \mid Y \text{ upwards closed}\} \quad [\![\Box]\!]_X(A) = \{Y \in \mathcal{M}(X) \mid A \in Y\}$$

which recovers the standard semantics in a coalgebraic setting [8].
It is readily verified that all structures above are indeed monotone.

3 The Model-Checking Game

We start by describing a characterisation of model checking in terms of parity games that generalises [18, Theorem 1, Chapter 6] to the coalgebraic setting. The model-checking game is a variant of the one from [3]. A *parity game* played by $\exists$ (Éloise) and $\forall$ (Abelard) is a tuple $\mathcal{G} = (B_\exists, B_\forall, E, \Omega)$ where $B = B_\exists \cup B_\forall$ is the disjoint union of *positions* owned by $\exists$ and $\forall$, respectively, $E \subseteq B \times B$ indicates the allowed moves, and $\Omega : B \to \omega$ is a (parity) map with finite range. An infinite sequence $(b_0, b_1, b_2, \dots)$ of positions is called *bad* if $\max\{k \mid k = \Omega(b_i)$ for infinitely many $i \in \omega\}$ is odd.

A *play* in $\mathcal{G}$ is a finite or infinite sequence of positions $(b_0, b_1, \dots)$ with the property that $(b_i, b_{i+1}) \in E$ for all i, i.e. all moves are legal, and b_0 is the *initial position* of the play. A *full play* is either infinite, or a finite play ending in a position b_n where $E[b_n] = \{b \in B \mid (b_n, b) \in E\} = \emptyset$, i.e. no more moves are possible. A finite play is lost by the player who cannot move, and an infinite play $(b_0, b_1, \dots)$ is lost by $\exists$ if $(b_0, b_1, \dots)$ is bad.

A *strategy* in $\mathcal{G}$ for a player $P \in \{\exists, \forall\}$ is a function s that maps plays that end in a position $b \in B_P$ of P to a position $b' \in B$ such that $(b, b') \in E$ whenever $E[b] \neq \emptyset$. Intuitively, a strategy determines a player's next move, depending on the history of the play, whenever the player has a move available. A strategy for a player $P \in \{\exists, \forall\}$ is called *history-free* if it only depends on the last position of a play. Formally, a *history-free strategy* for player $P \in \{\exists, \forall\}$ is a function $s : B_P \to B$ such that $(b, s(b)) \in E$ for all $b \in B_P$ with $E[b] \neq \emptyset$. A play $(b_0, b_1, \dots)$ is *played according to some strategy* s if $b_{i+1} = s(b_0 \dots b_i)$ for all i with $b_i \in B_P$. Similarly a play $(b_0, b_1, \dots)$ is played according to some *history-free strategy* s if $b_{i+1} = s(b_i)$ for all i with $b_i \in B_P$. Finally, we say s is a *winning strategy* from position $b \in B$ if P wins all plays with initial position b that are played according to s.

We will use the fact that parity games are history-free determined [5, 13] and that winning regions can be decided in UP $\cap$ co-UP [9].

Theorem 2. *At every position $b \in B_\exists \cup B_\forall$ in a parity game $\mathcal{G} = (B_\exists, B_\forall, E, \Omega)$ one of the players has a history-free winning strategy. Furthermore, for every $b \in B_\exists \cup B_\forall$, it can be determined in time $O\left(d \cdot m \cdot \left(\frac{n}{\lfloor d/2 \rfloor}\right)^{\lfloor d/2 \rfloor}\right)$ which player has a winning strategy from position b, where n, m and d are the size of B, E and the range of Ω, respectively.*

The model checking game is played on both states and formulas, and only the closure of the initial formula, which is assumed to be clean and guarded, is relevant:

Definition 3. *A set $\Gamma \subseteq \mathcal{F}(\Lambda)$ of formulas is* closed *if $B \in \Gamma$ whenever B is a subformula of some $A \in \Gamma$ and $A[p := \eta p.A] \in \Gamma$ if $\eta p.A \in \Gamma$, where $\eta \in \{\mu, \nu\}$. The* closure *of Γ is the smallest closed set $\mathrm{Cl}(\Gamma)$ for which $\Gamma \subseteq \mathrm{Cl}(\Gamma)$.*

A formula $A \in \mathcal{F}(\Lambda)$ is guarded *if, for all subformulas $\eta p.B$ of A, p only appears in the scope of a modal operator within B, and A is* clean *if every variable is bound at most once in A. A set of formulas is clean/guarded if this applies to every element.*

In the model checking game, the unfolding of fixpoint formulas gives rise to infinite plays, and we have to ensure that all infinite plays that cycle on an outermost μ-variable are lost by $\exists$ (who claims that the formula(s) under consideration are satisfied), as this would correspond to the infinite unfolding of a least fixpoint. This is achieved by the parity function.

Definition 4. *A* parity map *for a finite, clean set of formulas Γ is a function $\Omega : \mathrm{Cl}(\Gamma) \to \omega$ with finite range for which $\Omega(A) = 0$ unless A is of the form $\eta p.B$, $\eta \in \{\mu, \nu\}$, $\Omega(A)$ is odd (even) iff A is of the form $\mu p.B$ ($\nu p.B$), and $\Omega(\eta_1 p_1.B_1) \leq \Omega(\eta_2 p_2.B_2)$ whenever $\eta_1 p_1.B_1$ is a subformula of $\eta_2 p_2.B_2$, where $\eta_1, \eta_2 \in \{\mu, \nu\}$.*

It is easy to see that every clean set of formulas admits a parity function.

Lemma 5. *If $\Gamma \subseteq \mathcal{F}(\Lambda)$ is finite and clean, then Γ admits a parity function whose range is bounded by the cardinality of* $\mathrm{Cl}(\Gamma)$.

A parity function for Γ defines the following game:

Definition 6. *Suppose that $M = (C, \gamma, \sigma)$ is a T-model, $\Gamma \subseteq \mathcal{F}(\Lambda)$ is finite, clean and guarded, and Ω is a parity map for Γ. The* model checking game *$\mathcal{MG}_\Gamma(M)$ is the parity game whose positions and admissible moves are given in the following table,*

Position: b	*Player*	*Admissible moves: $E[b]$*
$(p, c), c \in \sigma(p)$	$\forall$	$\emptyset$
$(p, c), c \notin \sigma(p)$	$\exists$	$\emptyset$
$(\eta p.A(p), c)$ *for* $\eta \in \{\mu, \nu\}$	$\exists$	$\{(A[p := \eta p.A(p)], c)\}$
$(A_1 \vee A_2, c)$	$\exists$	$\{(A_1, c), (A_2, c)\}$
$(A_1 \wedge A_2, c)$	$\forall$	$\{(A_1, c), (A_2, c)\}$
$(\heartsuit(A_1, \dots, A_n), c)$	$\exists$	$\{(\heartsuit(A_1, \dots, A_n), (U_1, \dots, U_n)) \mid$ $U_1, \dots, U_n \subseteq C, \gamma(c) \in [\![\heartsuit]\!]_C(U_1, \dots, U_n)\}$
$(\heartsuit(A_1, \dots, A_n), (U_1, \dots, U_n))$	$\forall$	$\{(A_i, c) \mid 1 \leq i \leq n, c \in U_i\}$

where $p \in V \cup \overline{V}$, $\heartsuit \in \Lambda \cup \overline{\Lambda}$, $A, A_1, \dots, A_n \in \mathrm{Cl}(\Lambda)$ are Λ-formulas, $c \in C$ are states and $U_i \subseteq C$ are state sets. The parity function of $\mathcal{MG}_\Gamma(M)$ is given by $\Omega'(A, c) = \Omega(A)$ for $A \in \mathrm{Cl}(\Gamma)$ and $c \in C$, and $\Omega'(\_) = 0$ otherwise.

As any two parity functions for a given set of formulas induce the same winning region for both players, we speak of *the* model checking game given by a set of formulas. The announced generalisation of [18, Theorem 1, Chapter 6] now takes the following form:

Theorem 7. *For Γ finite, clean and guarded, a T-model $M = (C, \gamma, \sigma)$, $A \in \mathrm{Cl}(\Gamma)$ and $c \in C$, $\exists$ has a winning strategy in $\mathcal{MG}_\Gamma(M)$ from position (A, c) iff $M, c \models A$.*

The model checking game is used to show completeness of associated tableau calculi.

4 Tableaux for the coalgebraic μ-calculus

The construction of tableaux for the coalgebraic μ-calculus relies on a set of rules that provides the glue between syntax and semantics. As we do not commit to a particular semantics, we exhibit coherence conditions that ensure soundness and completeness.

Definition 8. *A* monotone one-step tableau rule *for a similarity type Λ is of the form*

$$\frac{\Gamma_0}{\Gamma_1 \quad \dots \quad \Gamma_n}$$

where $\Gamma_0 \in (\Lambda \cup \overline{\Lambda})(V)$ and $\Gamma_1, \dots, \Gamma_n \subseteq V$, every propositional variable occurs at most once in Γ_0 and all variables occurring in one of the Γ_i's ($i > 0$) also occur in Γ_0.

Monotone tableau rules do not contain negated propositional variables, which are not needed to axiomatise (the class of models induced by) monotone Λ structures. The restriction on occurrences of propositional variables is unproblematic, as variables that occur in a conclusion but not in the premise and multiple occurrences of variables in the premise can always be eliminated. The set of one-step tableau rules is a (the only) parameter in the construction of tableaux for coalgebraic fixpoint logics. Example rules are most conveniently presented if we identify a linear inequality $\sum_i a_i p_i < k$ where $p_i \in V \cup \overline{V}$ and $a_i, k \in \mathbb{Q}$ with the set of prime implicants of the boolean function $(x_i) \mapsto 1$ iff $\sum a_i \mathsf{sg}(p_i) < k$ where $\mathsf{sg}(p) = 1$ and $\mathsf{sg}(\overline{p}) = 0$ for $p \in V$, and prime implicants are represented by sets of propositional variables.

Example 9. The following rule sets are used for the logics introduced in Example 1,

$$(K)\frac{\Diamond p_0; \Box p_1; \dots; \Box p_n}{p_0; p_1; \dots; p_n} \qquad (M)\frac{\Box p; \Diamond q}{p; q}$$

$$(G)\frac{\langle k_1 \rangle p_1; \dots; \langle k_n \rangle p_n; [l_1] q_1; \dots; [l_m] q_m}{\sum_{j=1}^m s_j \overline{q_j} - \sum_{i=1}^n r_i p_i < 0}$$

$$(P)\frac{\langle a_1 \rangle p_1; \dots; \langle a_n \rangle p_n; [b_1] q_1; \dots; [b_m] q_m}{\sum_{j=1}^m s_j \overline{q}_j - \sum_{i=1}^n r_i p_i < k}$$

$$(C_1)\frac{[C_1] p_1; \dots; [C_n] p_n}{p_1; \dots; p_n} \qquad (C_2)\frac{[C_1] p_1; \dots; [C_n] p_n; \overline{[D]} q; \overline{[N]} r_1; \dots; \overline{[N]} r_m}{p_1; \dots; p_n; q; r_1; \dots; r_m}$$

where $n, m \in \mathbb{N}$ and sets are represented by ;-separated lists. For the modal and monotone μ-calculus, we have all instances of (K) and (M), respectively. The graded μ-calculus uses all instances of (G) for which $r_i, s_j \in \mathbb{N} \setminus \{0\}$ and $\sum_{i=1}^n r_i(k_i + 1) \geq 1 + \sum_{j=1}^m s_j l_j$. The probabilistic μ-calculus is axiomatised by instances of (P) where $r_i, s_j \in \mathbb{N} \setminus \{0\}$ and $\sum_{i=1}^n r_i a_i - \sum_{j=1}^m s_j b_j \leq k$ if $n > 0$ and $-\sum_{j=1}^m s_j b_j < k$ if $n = 0$. Finally, we associate all instances of $(C_1), (C_2)$ for which the C_i are disjoint and moreover $C_i \subseteq D$ in the case of (C_2). We note that all rules above are monotone.

Tableaux themselves are formulated in terms of sequents:

Definition 10. *A Λ-*tableau sequent*, or just* sequent*, is a finite set of Λ-formulas. We write $\mathsf{S}(\Lambda)$ for the set of Λ-sequents. If $\Gamma \in \mathsf{S}(\Lambda)$ we write $\mathsf{S}(\Gamma) = \{\Delta \in \mathsf{S}(\Lambda) \mid \Delta \subseteq \mathrm{Cl}(\Gamma)\}$ for the set of sequents over the closure of Γ. We identify a formula $A \in \mathcal{F}(\Lambda)$ with the singleton set $\{A\}$, and write $\Gamma; \Delta = \Gamma \cup \Delta$ for the union of $\Gamma, \Delta \in \mathsf{S}(\Lambda)$ as before. Substitution extends to sequents via $\Gamma\sigma = \{A\sigma \mid A \in \Gamma\}$.*

The set TR of tableau rules *induced by a set R of one-step rules contains the propositional and fixpoint rules, the modal rules (m) and the axiom (rule) below:*

$$(\wedge)\frac{\Gamma; A \wedge B}{\Gamma; A; B} \quad (\vee)\frac{\Gamma; A \vee B}{\Gamma; A \quad \Gamma; B} \quad (\mathsf{f})\frac{\Gamma; \eta p.A}{\Gamma; A[p := \eta p.A]} \quad (\mathsf{m})\frac{\Gamma_0\sigma, \Delta}{\Gamma_1\sigma \ldots \Gamma_n\sigma} \quad (\mathsf{Ax})\frac{\Gamma, A, \overline{A}}{}$$

Here, $\Gamma, \Delta \in \mathsf{S}(\Lambda)$ range over sequents and $A, B \in \mathcal{F}(\Lambda)$ over formulas. In (m), $\Gamma_0/\Gamma_1 \ldots \Gamma_n \in \mathsf{R}$ and $\sigma : V \to \mathcal{F}(\Lambda)$ is so that $A\sigma = B\sigma$ only if $A = B$ for $A, B \in \Gamma_0$. An axiom *is a premise of (Ax). The sequent Δ is called a* context *of the modal rule $\Gamma_0\sigma, \Delta/\Gamma_1\sigma \ldots \Gamma_n\sigma$, and the context of a non-modal rule is always empty.*

We only allow substitutions that do not duplicate literals in the premise of modal rules to ensure decidability, and we require that the set of one-step tableau rules is closed under contraction later. Since fixpoint rules generate infinite paths, we formalise tableaux as finite, rooted graphs. As a consequence, *closed* tableaux are finitely represented proofs of the unsatisfiability of the root formula.

Definition 11. *A* tableau *for a clean, guarded sequent $\Gamma \in \mathsf{S}(\Lambda)$ is a finite, directed, rooted and labelled graph (N, K, R, ℓ) where N is the set of nodes, $K \subseteq N \times N$ is the set of edges, R is the root node and $\ell : N \to \mathsf{S}(\Gamma)$ is a labelling function such that $\ell(R) = \Gamma$ and, if $K(n) = \{n' \mid (n, n') \in K\}$:*

- *if $\ell(n)$ is not the premise of a rule in TR, then $K(n) = \emptyset$.*
- *if $\ell(n)$ is a premise of a rule in TR, then $\ell(n)/\{\ell(n') \mid n' \in K(n)\} \in \mathsf{TR}$.*

An annotation *of a tableau is a mapping $\alpha : N \to \mathsf{S}(\Lambda)$ such that $\alpha(n)$ is a context of the rule $\ell(n)/\{\ell(n') \mid n' \in K(n)\}$ whenever $K(n) \neq \emptyset$.*

In other words, tableaux are sequent-labelled graphs where a rule has to be applied at a node if the node label matches a rule premise, no rule may be applied otherwise. The purpose of annotations is to record the weakening steps immediately prior to the applications of modal rules, which is needed for the definition of traces later.

Our goal is to show that a formula $A \in \mathcal{F}(\Lambda)$ is satisfiable iff no tableau for A ever closes. In a setting without fixpoints, a tableau is closed iff all leaves are labelled with axioms. Here we also need to consider infinite paths, and ensure that only greatest fixpoints are unfolded infinitely often at the top level of an infinite path. As in [14], this necessitates to consider the set of *traces* through a given tableau.

Definition 12. *The set of* directional rule names *is given by $\mathsf{N} = \{\vee_l, \vee_r, \wedge, \mathsf{f}, \mathsf{m}\}$, and an* instance *of $\flat \in \mathsf{N}$ is an instance of the $\vee$-rule if $\flat = \vee_l$ or $\flat = \vee_r$ and an instance of a fixpoint rule/modal rule if $\flat = \mathsf{f}/\mathsf{m}$. A* trace tile *is a triple $t = (\Delta, \flat, \Delta')$ for $\Delta, \Delta' \in \mathsf{S}(\Lambda)$ and $\flat \in \mathsf{N}$. The trace tile t is* consistent *if there exists an instance of $\flat$ with empty context that has Γ as a premise and Δ as one of its conclusions, where Δ has to be the left (right) conclusion of the $\vee$-rule in case $\flat = \vee_l$ ($\flat = \vee_r$). A* path *through* *a tableau* $\mathsf{T} = (N, K, R, \ell)$ *is a finite or infinite sequence of nodes and directional rule names*

$$\pi = n_0 \xrightarrow{\flat_0} n_1 \xrightarrow{\flat_1} n_2 \xrightarrow{\flat_2} n_3 \ldots$$

such that $n_{i+1} \in K(n_i)$, $\ell(n_i)/\{\ell(n') \mid n' \in K(n)\}$ *is an instance of* $\flat$, *and* $(\ell(n_i) \setminus \alpha(n_i), \flat_i, \ell(n_{i+1}))$ *is a consistent trace tile. A finite path* π *is* of maximal length *if* $K(n) = \emptyset$ *for the end node* n *of* π.

If α *is an annotation for* T, *then an* α-trace *through* π *is a finite or infinite sequence of formulas* $(A_0, A_1, \ldots)$ *such that* $A_i \in \ell(n_i)$ *and* $(A_i, A_{i+1}) \in \mathsf{Tr}(\ell(n_i) \setminus \alpha(n_i), \flat_i, \ell(n_{i+1}))$ *where the relations* $\mathsf{Tr}(\Gamma, \flat, \Delta)$ *induced by trace tiles are given by*

- $\mathsf{Tr}((\Gamma, A \vee B), \vee_l, (\Gamma, A)) = \{(A \vee B, A)\} \cup \mathsf{Diag}(\Gamma)$
- $\mathsf{Tr}((\Gamma, A \vee B), \vee_r, (\Gamma, B)) = \{(A \vee B, B)\} \cup \mathsf{Diag}(\Gamma)$
- $\mathsf{Tr}((\Gamma, A \wedge B), \wedge, (\Gamma, A, B)) = \{(A \wedge B, A), (A \wedge B, B)\} \cup \mathsf{Diag}(\Gamma)$
- $\mathsf{Tr}(\Gamma, \mathsf{m}, \Delta) = \{(\heartsuit(B_1, \ldots, B_k), B_j) \mid \heartsuit(B_1, \ldots, B_k) \in \Gamma, B_j \in \Delta, j \leq k\}$
- $\mathsf{Tr}((\Gamma, \eta p.B), \mathsf{f}, (\Gamma, B[p := \eta p.B]) = \{\eta p.B, B[p := \eta p.B])\} \cup \mathsf{Diag}(\Gamma)$
- $\mathsf{Tr}(\Gamma, \flat, \Delta) = \emptyset$ *otherwise.*

where $\mathsf{Diag}(X) = \{(x, x) \mid x \in X\}$ *is the diagonal on any set* X. *Finally, a tableau* T *is* closed, *if there exists an annotation* α *such that the end node of any finite path through* T *of maximal length that starts in the root node is labelled with a tableau axiom and every infinite path starting in the root node carries at least one bad* α*-trace with respect to a parity function* Ω *for* Γ.

Consistent trace tiles record the premise of a rule and one of its conclusions, together with the directional rule name. Here we require empty context, so that the rule that witnesses consistency of a trace tile is necessarily a substituted one-step rule. This implies that all traces on a tableau path ending in the context of a rule premise terminate.

Example 13. If nodes are represented by their labels, then the path

$$A \vee \mu p.B; C \xrightarrow{\vee_r} \mu p.\Diamond B; C \xrightarrow{\mathsf{f}} \Diamond B[p := \mu p.B]; C \ldots$$

supports the traces $(A \vee \mu p.\Diamond B, \mu p.\Diamond B, \Diamond B[p := \mu p.B], \ldots)$ and $(C, C, C, \ldots)$. Note that there is no trace on this path that starts with A.

Our goal is to show that non-existence of a closed tableau is equivalent to the satisfiability of the root formula. This is where we need coherence conditions between the axiomatisation and the (coalgebraic) semantics. In essence, we require that one-step tableau rules characterise satisfiability of a set of modalised formulas of the form $\heartsuit(A_1, \ldots, A_n)$ purely in terms of the Λ-structure.

Definition 14. *The* interpretation *of a propositional sequent* $\Gamma \subseteq V \cup \overline{V}$ *with respect to a set* X *and a valuation* $\tau : V \to \mathcal{P}(X)$ *is given by* $[\![\Gamma]\!]_{X,\tau} = \bigcap\{\tau(p) \mid p \in \Gamma\}$, *and the interpretation* $[\![\Gamma]\!]_{TX,\tau} \subseteq TX$ *of a modalised sequent* $\Gamma \subseteq (\Lambda \cup \overline{\Lambda})(V)$ *is*

$$[\![\Gamma]\!]_{TX,\tau} = \cap\{[\![\heartsuit]\!]_X(\tau(p_1), \ldots, \tau(p_n)) \mid \heartsuit(p_1, \ldots, p_n) \in \Gamma\}.$$

If T *is a* Λ*-structure, then a set* R *of monotone tableau rules for* Λ *is* one-step tableau complete *with respect to* T *if* $[\![\Gamma]\!]_{TX,\tau} \neq \emptyset$ *iff for all* $\Gamma_0/\Gamma_1, \ldots, \Gamma_n \in \mathsf{R}$ *and all* $\sigma : V \to V$ *with* $\Gamma_0\sigma \subseteq \Gamma$, *there exists* $1 \leq i \leq n$ *such that* $[\![\Gamma_i\sigma]\!]_{X,\tau} \neq \emptyset$, *whenever* $\Gamma \subseteq (\Lambda \cup \overline{\Lambda})(V)$ *and* $\tau : V \to \mathcal{P}(X)$.

Informally speaking, a set R of one-step tableau rules is one-step tableau complete if a modalised sequent Γ is satisfiable iff any rule that matches Γ has a satisfiable conclusion.

An adaptation of [16, Theorem 17] to the setting of monotone tableau rules establishes existence of a tableau complete set of monotone rules for monotone Λ-structures.

Proposition 15. *Every monotone Λ-structure admits a one-step tableau complete set of monotone tableau rules.*

In our examples, the situation is as follows:

Proposition 16. *The rule sets introduced in Example 9 are one-step tableau complete with respect to the corresponding structures defined in Example 1.*

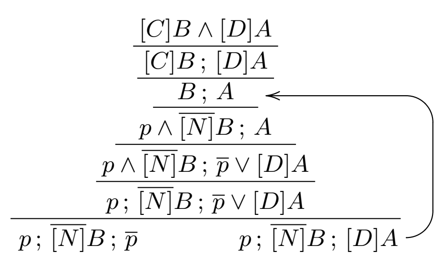
With the help of Theorem 7 we can now show that satisfiability precludes the existence of closed tableaux, as a winning strategy for $\exists$ in the model checking game induces a path through any tableau that contradicts closedness.

Theorem 17. *Let R be a one-step tableau complete set of monotone rules for the modal similarity type Λ, and let $\Gamma \in \mathsf{S}(\Lambda)$ be clean and guarded. If Γ is satisfiable in some model $M = (C, \gamma, \sigma)$ at state $c \in C$, then no closed tableau for Γ exists.*

Example 18. Consider the following formula of the coalitional μ-calculus

$$[C]\nu X.(p \wedge \overline{[N]}X) \wedge [D]\mu Y.(\overline{p} \vee [D]Y)$$

stating that "coalition C can achieve that, from the next stage onwards, p holds irrespective of the strategies used by other agents, and coalition D can ensure (through suitable strategies used in the long term) that $\overline{p}$ holds after some finite number of steps". Here, we assume that $C, D \subseteq N$ are such that $C \cap D = \emptyset$. Define a parity map Ω for the above formula by $\Omega(\nu X.(p \wedge \overline{[N]}X)) = 2$, $\Omega(\mu Y.(\overline{p} \vee [D]Y)) = 1$ and $\Omega(A) = 0$ otherwise. The unsatisfiability of this formula is witnessed by the following closed tableau:



where $B = \nu X.(p \wedge \overline{[N]}X)$ and $A = \mu Y.(\overline{p} \vee [D]Y)$. Any finite path through this tableau ends in an axiom, and the only infinite path contains the trace

$$[C]B \wedge [D]A,\ [D]A,\ A,\ \overline{A,\ \overline{p} \vee [D]A,\ \overline{p} \vee [D]A,\ [D]A,\ A}$$

where the overlined sequence is repeated ad infinitum. This trace is bad with respect to Ω, as $\Omega(A) = 1$ and A is the only fixpoint formula that occurrs infinitely often.

5 The Tableau Game

We now introduce the tableau game associated to a clean and guarded sequent Γ, and use it to characterise the (non-)existence of closed tableaux in terms of winning strategies in the tableau game. For the entire section, we fix a modal similarity type Λ and a one-step tableau complete set R of monotone tableau rules. The idea underlying the tableau game is that $\forall$ intends to construct a closed tableau for a given set of formulas Γ, while $\exists$ wants to demonstrate that any tableau constructed by $\forall$ contains a path π that violates the closedness condition. As (certain) infinite plays of the tableau game correspond to paths through a tableau, an infinite play should be won by $\exists$ if it does not carry a bad trace, with bad traces being detected with the help of parity word automata.

Definition 19. *Let Σ be a finite alphabet. A* non-deterministic parity Σ-word automaton *is a quadruple $\mathbb{A} = (Q, a_I, \delta : Q \times \Sigma \to \mathcal{P}(Q), \Omega)$ where Q is the set of states of $\mathbb{A}$, $a_I \in Q$ is the initial state, δ is the transition function, and $\Omega : Q \to \omega$ is a parity function. Given an infinite word $\gamma = c_0c_1c_2c_3\dots$ over Σ, a run of $\mathbb{A}$ on γ is a sequence $\rho = a_0a_1a_2\dots \in Q^\omega$ such that $a_0 = a_I$ and for all $i \in \omega$ we have $a_{i+1} \in \delta(a_i, c_i)$. A run ρ is* accepting *if ρ is not a bad sequence with respect to Ω. We say that $\mathbb{A}$* accepts *an infinite Σ-word γ if there exists an accepting run ρ of $\mathbb{A}$ on γ. Finally we call $\mathbb{A}$* deterministic *if $\delta(a, c)$ has exactly one element for all $(a, c) \in Q \times \Sigma$. In other words, $\mathbb{A}$ is deterministic if δ is a function of type $Q \times \Sigma \to Q$.*

The tableau game uses an automaton over trace tiles (cf. Definition 12) to detect the existence of bad traces through infinite plays.

Lemma and Definition 20. Let $\Gamma \in \mathsf{S}(\Lambda)$ be a clean, guarded sequent, and let Σ_Γ denote the set of trace tiles $(\Delta, \flat, \Delta')$ with $\Delta, \Delta' \in \mathsf{S}(\Gamma)$. There exists a deterministic parity Σ_Γ-word automaton $\mathbb{A}_\Gamma = (Q_\Gamma, a_\Gamma, \delta_\Gamma, \Omega')$ such that $\mathbb{A}$ accepts an infinite sequence $(t_0, t_1, \dots) \in \Sigma_\Gamma^\infty$ of trace tiles iff there is no sequence of formulas $(A_0, A_1, \dots)$ with $(A_i, A_{i+1}) \in \mathsf{Tr}(t_i)$ which is a bad trace with respect to a parity function for Γ. Moreover, the index of $\mathbb{A}$ and the cardinality of Q are bounded by $p(|\mathrm{Cl}(\Gamma)|)$ and $2^{p(|\mathrm{Cl}(\Gamma)|)}$ for a polynomial p, respectively. Such an automaton $\mathbb{A}$ is called a *Γ-parity automaton*.

Definition 21. *Let $\Gamma \in \mathsf{S}(\Lambda)$ be clean and guarded, and let $\mathbb{A} = (Q, a_\Gamma, \delta, \Omega)$ be a Γ-parity automaton. We denote the set of tableau rules $\Gamma_0/\Gamma_1, \dots, \Gamma_n \in \mathsf{TR}$ for which $\Gamma_i \in \mathsf{S}(\Gamma)$ by TR_Γ. The Γ-*tableau game *is the parity game $\mathcal{G}_\Gamma = (B_\exists, B_\forall, E, \Omega')$ where $B_\forall = \mathsf{S}(\Gamma) \times Q$, $B_\exists = \mathsf{TR}_\Gamma \times \mathsf{S}(\Gamma) \times Q$, and $(b_1, b_2) \in E$ if either*

- *$b_1 = (\Delta, a) \in B_\forall$ and $b_2 = (r, \Sigma, a) \in B_\exists$ where $r \in \mathsf{TR}_\Gamma$ has premise Δ and $\Sigma \subseteq \Delta$ is a context of r, or*
- *$b_1 = (r, \Sigma, a) \in B_\exists$, $b_2 = (\Delta, a') \in B_\forall$ and there exists $\flat \in \mathsf{N}$ such that r is an instance of $\flat$, Δ is a conclusion of r, the trace tile $t = (\Gamma \setminus \Sigma, \flat, \Delta)$ is consistent where Γ is the premise of r, and $a' = \delta(a, t)$.*

The parity function $\Omega' : (B_\exists \cup B_\forall) \to \omega$ of $\mathcal{G}_\Gamma$ is given by $\Omega'(\Delta, a) = \Omega(a)$ if $(\Delta, a) \in B_\forall$ and $\Omega'(r, \Sigma, a) = 0$.

If not explicitly stated otherwise, we will only consider $\mathcal{G}_\Gamma$-plays that start at (Γ, a_Γ) where a_Γ is the initial state of the automaton $\mathbb{A}$. In particular, we say that a player has a winning strategy in $\mathcal{G}_\Gamma$ if she/he has a winning strategy in $\mathcal{G}_\Gamma$ at position (Γ, a_Γ).

The easier part of the correspondence between satisfiability and winning strategies in $\mathcal{G}_\Gamma$ is proved by constructing a closed tableau based on a winning strategy for $\forall$. The notion of trace through a $\mathcal{G}_\Gamma$-play is used to show closedness.

Definition 22. *For a $\mathcal{G}_\Gamma$-play*

$$\pi = (\Gamma^0, a_0)(r_0, \Sigma_0, a_0)(\Gamma^1, a_1)(r_1, \Sigma_1, a_1)\dots(\Gamma^l, a_l)(r_l, \Sigma_l, a_l)\dots$$

a sequence $\pi' = \Gamma^0\flat_0\Gamma^1\flat_1\dots\Gamma^l\flat_l\dots$ of sequents and directional rule names is an underlying path *of π if r_i is a $\flat_i$-rule, $t_i = (\Gamma^i \setminus \Sigma_i, \flat_i, \Gamma^{i+1})$ is a consistent trace tile, and $\delta(a_i, t_i) = a_{i+1}$ for all $i \in \mathbb{N}$. A sequence of formulas $\alpha = A_0A_1A_2\dots \in \mathcal{F}(\Lambda)^\infty$ is a* trace *through π if there exists an underlying path $\pi' = \Gamma^0\flat_0\Gamma^1\flat_1\Gamma^2\dots$ of π such that $(A_i, A_{i+1}) \in \mathsf{Tr}(t_i)$ for all $i \in \mathbb{N}$.*

An underlying path of a $\mathcal{G}_\Gamma$-play assigns directional rule names to the rules used in $\forall$'s moves, so that only consistent trace tiles are considered when defining traces.

Theorem 23. *Let $\Gamma \in \mathsf{S}(\Lambda)$ be clean and guarded. If $\forall$ has a winning strategy in $\mathcal{G}_\Gamma$, then Γ has a closed* TR*-tableau.*

The converse of the above theorem is established later as Theorem 26. The challenge there is to construct a model for Γ based on a winning strategy for $\exists$ in the Γ-tableau game. This crucially requires the set of tableau rules to be closed under contraction.

Definition 24. *A set* R *of monotone one-step rules is* closed under contraction*, if for all $\Gamma_0/\Gamma_1, \dots, \Gamma_n \in \mathsf{R}$ and all $\sigma : \mathsf{V} \to \mathsf{V}$, there exists a rule $\Delta_0/\Delta_1, \dots, \Delta_k \in \mathsf{R}$ and a renaming $\tau : V \to V$ such that $A\tau = B\tau$ for $A, B \in \Delta_0$ implies that $A = B$, $\Delta_0\tau \subseteq \Gamma_0\sigma$ and, for each $1 \le i \le n$, there exists $1 \le j \le k$ such that $\Gamma_i\sigma \subseteq \Delta_j\tau$.*

In other words, instances of one-step rules which duplicate literals in the premise may be replaced by instances for which this is not the case. Under this condition, we prove:

Theorem 25. *Suppose that $\Gamma \in \mathsf{S}(\Lambda)$ is clean and guarded and* R *is one-step tableau complete and contraction closed. If $\exists$ has a winning strategy in $\mathcal{G}_\Gamma$, then Γ is satisfiable in a model of size $\mathcal{O}(2^{p(n)})$ where n is the cardinality of $\mathrm{Cl}(\Gamma)$ and p is a polynomial.*

The proof of Theorem 25 constructs a model for Γ out of the game board of $\mathcal{G}_\Gamma$ using a winning strategy f for $\exists$ in $\mathcal{G}_\Gamma$. We use one-step tableau completeness to impose a T-coalgebra structure on the $\mathcal{G}_\Gamma$-positions reachable through an f-conform $\mathcal{G}_\Gamma$-play, in such a way that the truth lemma is satisfied. We subsequently equip this T-coalgebra with a valuation that makes Γ satisfiable in the resulting model. While our construction shares some similarities with the shallow model construction of [17], it is by no means a simple adaptation of loc. cit., as we are dealing with fixpoint formulas and thus cannot employ induction over the modal rank of formulas to construct satisfying models. Our proof of satisfiability is also substantially different from the corresponding proof for the modal μ-calculus (cf. [14]) – we show satisfiability by directly deriving a winning strategy for $\exists$ in the model-checking game from a winning strategy of $\exists$ in the tableau game.

Putting everything together, we obtain a complete characterisation of satisfiability in the coalgebraic μ-calculus.

Theorem 26. *Suppose that $\Gamma \in \mathsf{S}(\Lambda)$ is a clean, guarded sequent and R is one-step tableau complete and contraction closed. Then Γ is satisfiable iff no tableau for Γ is closed iff $\exists$ has a winning strategy in the tableau game $\mathcal{G}_\Gamma$.*

As a by-product, we obtain the following small model property.

Corollary 27. *A satisfiable, clean and guarded formula A is satisfiable in a model of size $\mathcal{O}(2^{p(n)})$ where n is the cardinality of $\mathrm{Cl}(A)$ and p is a polynomial.*

Proof. The statement follows immediately from Theorems 17, 23 and 25 together with the determinacy of two player parity games.

6 Complexity

We now show that – subject to a mild condition on the rule set – the satisfiability problem of the coalgebraic μ-calculus is decidable in exponential time. By Theorem 26, the satisfiability problem is reducible to the existence of winning strategies in parity games.

To measure the size of a formula, we assume that the underlying similarity type Λ is equipped with a size measure $s : \Lambda \to \mathbb{N}$ and measure the size of a formula A in terms of the number of subformulas counted with multiplicities, adding $s(\heartsuit)$ for every occurrence of a modal operator $\heartsuit \in \Lambda$ in A. In the examples, we code numbers in binary, that is, $s(\langle k \rangle) = s([k]) = \lceil \log_2 k \rceil$ for the graded μ-calculus and $\langle p/q \rangle = [p/q] = \lceil \log_2 p \rceil + \lceil \log_2 q \rceil$ for the probabilistic μ-calculus, and $s([a_1, \ldots, a_k]) = 1$ for coalition logic. The definition of size is extended to sequents by $\mathsf{size}(\Gamma) = \sum_{A \in \Gamma} \mathsf{size}(A)$ for $\Gamma \in \mathsf{S}(\Lambda)$ and $\mathsf{size}(\{\Gamma_1, \ldots, \Gamma_n\}) = \sum_{i=1}^n \mathsf{size}(\Gamma_i)$ for sets of sequents. To apply Theorem 26 we need to assume that the formula that we seek to satisfy is both clean and guarded, but this can be achieved in linear time.

Lemma 28. *For every formula $A \in \mathcal{F}(\Lambda)$ we can find an equivalent clean and guarded formula in linear time.*

The proof is a straightforward generalisation of a similar result [11, Theorem 2.1]. To ensure that the size of the game board remains exponential, we encode the set of positions of the game board by strings of polynomial length, measured in the size of the initial sequent, and the rules need to be computationally well behaved. We require:

Definition 29. *A set R of tableau rules is* exponentially tractable*, if there exists an alphabet Σ and two functions $f : \mathsf{S}(\Lambda) \to \mathcal{P}(\Sigma^*)$ and $g : \Sigma^* \to \mathcal{P}(\mathsf{S}(\Lambda))$ together with a polynomial p such that $|x| \le p(\mathsf{size}(\Gamma))$ for all $x \in f(\Gamma)$, $\mathsf{size}(\Delta) \le p(|y|)$ for all $\Delta \in g(y)$, so that, for $\Gamma \in \mathsf{S}(\Lambda)$,*

$$\{g(x) \mid x \in f(\Gamma_0)\} = \{\{\Gamma_1, \ldots, \Gamma_n\} \mid \Gamma_0/\Gamma_1, \ldots, \Gamma_n \in R\}$$

and both relations $x \in f(\Gamma)$ and $\Gamma \in g(x)$ are decidable in EXPTIME.

Tractability of the set TR of tableau rules follows from tractability of the substitution instances of rules in R, as the non-modal rules can be encoded easily.

Lemma 30. *Suppose* R *is a set of monotone one-step rules. Then* TR *is exponentially tractable iff the set* $\{\Gamma_0\sigma/\Gamma_1\sigma, \dots, \Gamma_n\sigma \mid \Gamma_0/\Gamma_1, \dots, \Gamma_n \in \mathsf{R}, \forall A, B \in \Gamma_0(A\sigma = B\sigma \implies A = B)\}$ *of substituted one-step rules is exponentially tractable.*

Exponential tractability bounds the board of the tableau game and the complexity of both the parity function and the relation determining legal moves.

Lemma 31. *Suppose that* R *is exponentially tractable. Then every position in the tableau game* $\mathsf{G}_\Gamma = (B_\exists, B_\forall, E, \Omega)$ *of* $\Gamma \in \mathsf{S}(\Lambda)$ *can be represented by a string of polynomial length in* $\mathsf{size}(\Gamma)$*. Under this coding, the relation* $(b, b') \in E$ *is decidable in exponential time and the parity function* Ω *can be computed in exponential time.*

Together with Lemma 28, we now obtain an EXPTIME upper bound for satisfiability.

Corollary 32. *Suppose* T *is a monotone* Λ*-structure and* R *is exponentially tractable, contraction closed and one-step tableau complete for* T*. Then the problem of deciding whether* $\exists$ *has a winning strategy in the tableau game for a clean, guarded sequent* $\Gamma \in \mathsf{S}(\Lambda)$ *is in EXPTIME. As a consequence, the same holds for satisfiability of* $A \in \mathcal{F}(\Lambda)$*.*

Proof. The first assertion follows from Lemma 31 as the problem of deciding the winner in a parity game is exponential only in the size of the parity function of the game (Theorem 2) which is polynomial in the size of Γ (Lemma 20). The second statement now follows with the help of Theorem 26.

Example 33. It is easy to see that the rule sets for the modal μ-calculus, the coalitional μ-calculus and the monotone μ-calculus are exponentially tractable, as the number of conclusions of each one-step rule is bounded. To establish exponential tractability for the rule sets for the graded and probabilistic μ-calculus, we argue as in [17] where tractability of the (dual) proof rules has been established. We encode the conclusion $\sum_{i=1}^{n} r_i a_i < k$ as $(r_1, a_1, \dots, r_n, a_n, k)$ and Lemma 6.16 of *op. cit.* provides a polynomial bound on the size of the solutions for the linear inequalities that combine conclusion and side condition of both the (G) and (P)-rule. This allows us to guess the set of prime implicants of the conclusion in nondeterministic polynomial time, which shows that both rule sets are exponentially tractable. In all cases, contraction closure is immediate.

7 Conclusions

We have introduced the coalgebraic μ-calculus, a generic and uniform framework for modal fixpoint logics. Our main results are soundness and completeness of tableaux calculi, and an EXPTIME upper bound for the satisfiability problem. Concrete instances of the generic approach directly

- reproduce the complexity bound for the modal μ-calculus [6], together with the completeness of a slight variant of the tableau calculus presented in [14]
- lead to a new proof of the known EXPTIME bound for the graded μ-calculus [12]
- establish previously unknown EXPTIME bounds for the probabilistic μ-calculus, for coalition logic with fixpoints and the monotone μ-calculus.

We note that these bounds are tight for all logics except possibly the monotone μ-calculus, as the modal μ-calculus can be encoded into all other logics.

References

1. J. C. Bradfield. On the expressivity of the modal mu-calculus. In C. Puech and R. Reischuk, editors, *Proc. STACS 1996*, volume 1046 of *Lecture Notes in Computer Science*, pages 479–490. Springer, 1996.
2. B. Chellas. *Modal Logic*. Cambridge, 1980.
3. C. Cîrstea and M. Sadrzadeh. Modular Games for Coalgebraic Fixed Point Logics. In J. Adámek and C. Kupke, editors, *Coalgebraic Methods in Computer Science (CMCS'2008)*, volume 203 of *ENTCS*, 2008.
4. E. Emerson and C. Jutla. The complexity of tree automata and logics of programs. In *Proc. FOCS 1988*, pages 328–337. IEEE, 1988.
5. E. Emerson and C. 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.
6. E. A. Emerson and C. S. Jutla. The complexity of tree automata and logics of programs. *SIAM J. Comput.*, 29(1):132–158, 1999.
7. K. Fine. In so many possible worlds. *Notre Dame J. Formal Logic*, 13:516–520, 1972.
8. H. H. Hansen and C. Kupke. A coalgebraic perspective on monotone modal logic. In J. Adámek and S. Milius, editors, *Coalgebraic Methods in Computer Science*, volume 106 of *ENTCS*, pages 121–143. Elsevier, 2004.
9. M. Jurdziński. Small Progress Measures for Solving Parity Games. In *Proceedings of the 17th Annual Symposium on Theoretical Aspects of Computer Science, STACS*, volume 1770 of *LNCS*, pages 290–301, 2000.
10. D. Kozen. Results on the propositional μ-calculus. *Theoret. Comput. Sci.*, 27:333–354, 1983.
11. O. Kupferman, M. Vardi, and P. Wolper. An automata-theoretic approach to branching-time model checking. *Journal of the ACM*, 47(2):312–360, 2000.
12. U. S. O. Kupferman and M. Y. Vardi. The complexity of the graded mu-calculus. In *Proc. CADE 2002*, volume 2392 of *LNCS*, pages 423–437. Springer, 2002.
13. A. Mostowski. Games with forbidden positions. Technical Report 78, Instytut Matematyki, Uniwersytet Gdański, Poland, 1991.
14. D. Niwinski and I. Walukiewicz. Games for the mu-calculus. *Theor. Comput. Sci.*, 163(1&2):99–116, 1996.
15. M. Pauly. A modal logic for coalitional power in games. *J. Logic Comput.*, 12(1):149–166, 2002.
16. L. Schröder. A finite model construction for coalgebraic modal logic. In L. Aceto and A. Ingólfsdóttir, editors, *Foundations Of Software Science And Computation Structures*, volume 3921 of *LNCS*, pages 157–171. Springer, 2006.
17. L. Schröder and D. Pattinson. PSPACE bounds for rank-1 modal logics. *ACM Trans. Compl Log.*, 2(10), 2008. to appear.
18. C. Stirling. *Modal and Temporal Properties of Processes*. Texts in Computer Science. Springer, 2001.
19. Y. Venema. Automata and fixed point logics: a coalgebraic perspective. *Inform. Comput.*, 204(4):637–678, 2006.
20. I. Walukiewicz. Completeness of Kozen's axiomatisation of the propositional μ-calculus. *Inf. Comput.*, 157(1-2):142–182, 2000.