The decidability of guarded fixed point logic

Erich Grädel

Abstract

Guarded fixed point logics are obtained by adding least and greatest fixed points to the guarded fragments of first-order logic that were recently introduced by Andréka, van Benthem and Németi. Guarded fixed point logics can also be viewed as the natural common extensions of the modal μ-calculus and the guarded fragments. In a joint paper with Igor Walukiewicz, we proved recently that the satisfiability problems for guarded fixed point logics are decidable and complete for deterministic double exponential time. That proof relies on alternating automata on trees and on a forgetful determinacy theorem for games on graphs with unbounded branching. We present here an elementary proof for the decidability of guarded fixed point logics which is based on guarded bisimulations, a tree model property for guarded logics and an interpretation into the monadic second-order theory of countable trees.

Contents

1 Introduction **2**

2 Guarded fixed point logics **4**

3 The tree model property **7**

4 Reduction to the monadic theory of trees **10**

5 Variations **13**

1 Introduction

Modal logics are widely used in a number of areas in computer science, in particular for the specification and verification of hardware and software systems, for knowledge representation, in databases, and in artificial intelligence. The most important reason for the successful applications of these logics is that they provide a good balance between expressive power and computational complexity. A great number of logical formalisms have been successfully tailored in such a way that they are powerful enough to express interesting properties for a specific application but still admit reasonably efficient algorithms for their central computational problems, in particular for model checking and for satisfiability or validity tests. Many of these formalisms are essentially modal logics although this is not always apparent from their 'official' definitions (as for instance in the case of *description logics*).

The basic propositional (poly)modal logic ML (for a given set A of 'actions' or 'modalities') extends propositional logic by the possibility to construct formulae $\langle a\rangle\psi$ and $[a]\psi$ (where $a \in A$) with the meaning that ψ holds at some, respectively each, a-successor of the current state. Although ML is too weak for most of the really interesting applications, it can be extended by features like path quantification, transitive closure operators, counting quantifiers, least and greatest fixed points, and it has turned out that most of these extensions are still decidable and indeed of considerable practical importance.

Up to now, the reasons for these good algorithmic properties of modal logics have not been sufficiently understood. In [15] Vardi explicitly asked the question: *"Why is modal logic so robustly decidable?"*.

To discuss this question, it is useful to consider propositional modal logic as a fragment of first-order logic. Kripke structures which provide the semantics for modal logics, are relational structures with only unary and binary relations. Every formula $\psi \in$ ML can be translated into a first-order formula $\psi^*(x)$ with one free variable, which is equivalent in the sense that for every Kripke structure $\mathcal{K}$ with a distinguished node w we have that $\mathcal{K}, w \models \psi$ if and only if $\mathcal{K} \models \psi^*(w)$. This translation takes an atomic proposition P to the atom Px, it commutes with the Boolean connectives, and it translates the modal operators by quantifiers as follows:

$$\begin{aligned}\langle a\rangle\psi &\rightsquigarrow (\langle a\rangle\psi)^*(x) := \exists y(E_axy \wedge \psi^*(y))\\ [a]\psi &\rightsquigarrow ([a]\psi)^*(x) := \forall y(E_axy \rightarrow \psi^*(y)),\end{aligned}$$

where $\psi^*(y)$ is obtained from $\psi^*(x)$ by replacing all occurrences of x by y and vice versa and where E_a is the transition relation associated with the modality a.

The *modal fragment* of first-order logic is the image of propositional modal logic under this translation. It is properly contained in FO^2, relational first-order logic with only two variables. But although FO^2 is decidable and has the finite model property (see [12, 6]), it lacks the nice model-theoretic properties [1, 9] and, in particular, the robust decidability properties of modal logics. Indeed while the extensions of modal logic by path quantification, transitive closure operators, least fixed points etc. are still decidable, most of the corresponding extensions of FO^2 are highly undecidable (see [7, 8]). In particular this is the case for fixed-point logic with two variables, which is the natural common extension of FO^2 and the μ-calculus. The embedding of ML in FO^2 therefore does *not* give a satisfactory answer to Vardi's question.

An important property that is shared by modal logic and its extensions, but not by FO^2, and whose importance for the decidability of modal logics has been pointed out by Vardi [15] is the *tree model property*: Every satisfiable sentence has a model that is a tree of bounded branching. The tree model property is the basis for the use of automata theoretic techniques to decide the satisfiability problems for modal logics.

An alternative explanation for the good properties of modal logics has been proposed by Andréka, van Benthem and Németi [1]. Starting from the observation that in the translation of modal formulae into first-order formulae, the quantifiers are used only in a very restricted way, they defined the *guarded fragment* of first-order logic. They dropped the restriction to use only two variables and only monadic and binary predicates, but imposed that all quantifiers must be relativized by atomic formulae. This means that quantifiers appear only in the form

$$\exists \boldsymbol{y}(\alpha(\boldsymbol{x},\boldsymbol{y}) \wedge \psi(\boldsymbol{x},\boldsymbol{y})) \qquad \text{or} \qquad \forall \boldsymbol{y}(\alpha(\boldsymbol{x},\boldsymbol{y}) \rightarrow \psi(\boldsymbol{x},\boldsymbol{y})).$$

Thus quantifiers may range over a tuple $\boldsymbol{y}$ of variables, but are 'guarded' by an atom α that contains all the free variables of ψ.

The guarded fragment GF extends the modal fragment and turns out to have interesting properties [1, 5]: (1) The satisfiability problem for GF is decidable; (2) GF has the finite model property, i.e., every satisfiable formula in the guarded fragment has a finite model; (3) GF has (a generalized variant of) the tree model property; (4) Many important model theoretic properties which hold for first-order logic and modal logic, but not, say, for the bounded-variable fragments FO^k, do hold also for the guarded fragment; (5) The notion of equivalence under guarded formulae can be characterized by a straightforward generalization of bisimulation.

In a further paper, van Benthem [2] generalized the guarded fragment to the *loosely guarded fragment* (LGF) where quantifiers are guarded by conjunctions of atomic formulae of certain forms (details will be given in the next section.) Most of the properties of GF generalize to LGF. In [5] it is shown that the the satisfiability problems for GF and LGF are complete for 2EXPTIME, the class of problems solvable by a deterministic algorithm in time $2^{2^{p(n)}}$, for some polynomial $p(n)$.

Probably the most important extension of ML is the μ-calculus, introduced in [11], which extends propositional modal logic by least and greatest fixed points and subsumes most of the other program logics such as PDL, CTL and CTL* and also many description logics. It is known that the satisfiability problem for the μ-calculus is decidable and EXPTIME-complete [3]. Recently, Vardi has shown that also a stronger variant of the μ-calculus that includes backward modalities remains decidable in exponential time [16].

If it is indeed the case that, as suggested by Andréka, van Benthem and Németi, the guarded nature of quantification in modal logics is the main reason also for their good algorithmic properties, then we are naturally lead to the following question:

> If we extend the guarded fragments of first-order logic by least and greatest fixed points, do we still get a decidable logic?

In [10] we were able to give a positive answer to this question. We denote by μGF and μLGF the extensions of GF and LGF by least and greatest fixed points (precise definitions will be given in the next section). The model-theoretic and algorithmic methods that are available for the μ-calculus on one side, and the guarded fragments of first-order logic on the other side, can indeed be combined and generalized to provide positive results for the guarded fixed point logics. Using automata theoretic methods and a forgetful determinacy theorem for graph games we could in fact establish precise complexity bounds.

Theorem 1.1 (Grädel, Walukiewicz). *The satisfiability problems for* μGF *and* μLGF *are decidable and* 2EXPTIME*-complete.*

Hence, the guarded nature of quantification does indeed seem to provide a convincing explanation for the good algorithmic and model theoretic properties of modal logics.

In this paper, a somewhat simpler decidability proof for guarded fixed point logic is presented which replaces the automata theoretic machinery used in [10] by an interpretation argument into the monadic second-order theory of countable trees which by Rabin's famous result [13] is known to be decidable. However, note that while this gives a more elementary decidability proof, it does not give good complexity bounds. Indeed, even the first-order theory of countable trees is non-elementary, i.e. its time complexity exceeds every bounded number of iterations of the exponential function.

Here is the plan of this paper. In Sect. 2 we present the definitions and some basic properties of the guarded fragments of first-order logic, GF and LGF, and of the guarded fixed point logics μGF and μLGF. In Sect. 3 we explain the notions of a guarded bisimulation and of the unraveling of a structure and show that guarded fixed point logic satisfies a variant of the tree model property. Finally, in Sect. 4 we reduce the satisfiability problem for μLGF to the monadic second-order theory of countable trees and prove its decidability.

2 Guarded fixed point logics

Definition 2.1. The *guarded fragment* GF of first-order logic is defined inductively as follows:

(1) Every relational atomic formula belongs to GF.

(2) GF is closed under propositional connectives $\neg$, $\wedge$, $\vee$, $\rightarrow$ and $\leftrightarrow$.

(3) If $\boldsymbol{x}, \boldsymbol{y}$ are tuples of variables, $\alpha(\boldsymbol{x}, \boldsymbol{y})$ is a positive atomic formula and $\psi(\boldsymbol{x}, \boldsymbol{y})$ is a formula in GF such that $\mathrm{free}(\psi) \subseteq \mathrm{free}(\alpha) = \boldsymbol{x} \cup \boldsymbol{y}$, then the formulae

$$\exists \boldsymbol{y}(\alpha(\boldsymbol{x}, \boldsymbol{y}) \wedge \psi(\boldsymbol{x}, \boldsymbol{y}))$$
$$\forall \boldsymbol{y}(\alpha(\boldsymbol{x}, \boldsymbol{y}) \rightarrow \psi(\boldsymbol{x}, \boldsymbol{y}))$$

belong to GF.

Here $\mathrm{free}(\psi)$ means the set of free variables of ψ. An atom $\alpha(\boldsymbol{x}, \boldsymbol{y})$ that relativizes a quantifier as in rule (3) is the *guard* of the quantifier. Notice that the guard must contain *all* the free variables of the formula in the scope of the quantifier.

While the guarded fragment clearly contains the modal fragment of first-order logic, it seems not to be able to express all of temporal logic over $(\mathbb{N}, <)$. Indeed, the straightforward translation of $(\psi$ **until** $\varphi)$ into first-order logic

$$\exists y(x \leq y \wedge \varphi(y) \wedge \forall z((x \leq z \wedge z < y) \rightarrow \psi(z))$$

is not guarded in the sense of Definition 2.1. However, the quantifier $\forall z$ in this formula is guarded in a weaker sense, which lead van Benthem [2] to the following generalization of GF.

Definition 2.2. The *loosely guarded fragment* LGF is defined in the same way as GF, but the quantifier-rule is relaxed as follows:

$(3)'$ If $\psi(\boldsymbol{x}, \boldsymbol{y})$ is in LGF, and $\alpha(\boldsymbol{x}, \boldsymbol{y}) = \alpha_1 \wedge \cdots \wedge \alpha_m$ is a conjunction of atoms, then

$$\exists \boldsymbol{y}((\alpha_1 \wedge \cdots \wedge \alpha_m) \wedge \psi(\boldsymbol{x}, \boldsymbol{y}))$$
$$\forall \boldsymbol{y}((\alpha_1 \wedge \cdots \wedge \alpha_m) \rightarrow \psi(\boldsymbol{x}, \boldsymbol{y}))$$

belong to LGF, provided that $\mathrm{free}(\psi) \subseteq \mathrm{free}(\alpha) = \boldsymbol{x} \cup \boldsymbol{y}$ and for every quantified variable $y \in \boldsymbol{y}$ and every variable $z \in \{\boldsymbol{x}, \boldsymbol{y}\}$ there is at least one atom α_j that contains both y and z.

In the translation of $(\psi$ **until** $\varphi)$ described above, the quantifier $\forall z$ is loosely guarded by $(x \leq z \wedge z < y)$ since z coexists with both x and y in some conjunct of the guard. On the other side, the transitivity axiom $\forall xyz(Exy \wedge Eyz \rightarrow Exz)$ is not in LGF. The conjunction $Exy \wedge Eyz$ is not a proper guard of $\forall xyz$ since x and z do not coexist in any conjunct. Indeed, it has been shown in [5] that there is no way to express transitivity in LGF.

Notation. We will use the notation $(\exists \boldsymbol{y} \,.\, \alpha)$ and $(\forall \boldsymbol{y} \,.\, \alpha)$ for relativized quantifiers, i.e., we write guarded formulae in the form

$$(\exists \boldsymbol{y} \,.\, \alpha)\psi(\boldsymbol{x}, \boldsymbol{y}) \text{ and } (\forall \boldsymbol{y} \,.\, \alpha)\psi(\boldsymbol{x}, \boldsymbol{y}).$$

When this notation is used, then it is always understood that α is indeed a proper guard as specified by condition (3) or $(3)'$.

Definition 2.3. The guarded fixed point logics μGF and μLGF are obtained by adding to GF and LGF, respectively, the following rules for constructing fixed-point formulae:

Let W be a k-ary relation symbol, $\boldsymbol{x} = x_1, \dots, x_k$ a k-tuple of distinct variables and $\psi(W, \boldsymbol{x})$ be a guarded formula that contains only positive occurrences of W, no free variables other than $x_1, \dots, x_k$ and where W is not used in guards. Then we can build the formulae

$$[\text{LFP } W\boldsymbol{x} \,.\, \psi](\boldsymbol{x}) \qquad \text{and} \qquad [\text{GFP } W\boldsymbol{x} \,.\, \psi](\boldsymbol{x}).$$

The semantics of the fixed point formulae is the usual one: Given a structure $\mathfrak{A}$ providing interpretations for all free second-order variables in ψ, except W, the formula $\psi(W, \boldsymbol{x})$ defines an operator on k-ary relations $W \subseteq A^k$, namely

$$\psi^{\mathfrak{A}} : W \mapsto \psi^{\mathfrak{A}}(W) := \{\boldsymbol{a} \in A^k : \mathfrak{A} \models \psi(W, \boldsymbol{a})\}.$$

Since W occurs only positively in ψ, this operator is monotone (i.e., $W \subseteq W'$ implies $\psi^{\mathfrak{A}}(W) \subseteq \psi^{\mathfrak{A}}(W')$) and therefore has a least fixed point $\text{LFP}(\psi^{\mathfrak{A}})$ and a greatest fixed point $\text{GFP}(\psi^{\mathfrak{A}})$. Now, the semantics of least fixed point formulae is defined by

$$\mathfrak{A} \models [\text{LFP } W\boldsymbol{x} \,.\, \psi(W, \boldsymbol{x})](\boldsymbol{a}) \quad \text{iff} \quad \boldsymbol{a} \in \text{LFP}(\psi^{\mathfrak{A}})$$

and similarly for the greatest fixed points.

The Löwenheim-Skolem property. For future use, we recall that even the (unguarded) least fixed point logic (FO + LFP), and hence also its fragments μGF and μLGF, have the Löwenheim-Skolem property. This result is part of the folklore on fixed point logic, but it is hard to find a published proof. Our exposition follows the one in [4].

Theorem 2.4. *Every satisfiable sentence in (FO + LFP) has a model of countable cardinality.*

Proof. Consider a fixed-point formula of form $\psi(\boldsymbol{x}) := [\text{LFP } R\boldsymbol{x} \,.\, \varphi(R, \boldsymbol{x})](\boldsymbol{x})$, with first-order φ such that $\mathfrak{A} \models \psi(\boldsymbol{a})$ for some infinite model $\mathfrak{A}$.

For any ordinal α, let R^α be stage α of the least fixed point defined by φ on $\mathfrak{A}$, i.e $R^0 := \varnothing$, $R^\alpha := \varphi^{\mathfrak{A}}(\bigcup_{\beta<\alpha} R^\beta)$ for $\alpha > 0$. Then the least fixed point $\psi^{\mathfrak{A}}$ coincides with R^γ for some ordinal γ (whose cardinality is bounded by $|\mathfrak{A}|$); γ is called the *closure ordinal* of ψ on $\mathfrak{A}$.

Expand $\mathfrak{A}$ by a monadic relation U a binary relation $<$ and a $m+1$-ary relation S (where m is the arity of R) such that

(1) $(U, <)$ is a well-ordering of length $\gamma + 1$, and $<$ is empty outside U.

(2) S describes the stages of $\varphi^{\mathfrak{A}}$ in the following way

$$\begin{aligned} S \;\; := \;\; & \{(u, \boldsymbol{b}) : \text{ for some ordinal } \alpha \leq \gamma,\ u \text{ is the } \alpha\text{-th element of } (U, <), \\ & \text{and } \boldsymbol{b} \in R^\alpha\}. \end{aligned}$$

In the expanded structure $\mathfrak{A}^* := (\mathfrak{A}, U, <, S)$ the stages of the operator $\varphi^{\mathfrak{A}}$ are defined by the sentence

$$\eta := \forall u \forall \boldsymbol{x} \big(Su\boldsymbol{x} \leftrightarrow \exists z (z < u \land \varphi[R\boldsymbol{y}/\exists z(z < u \land Sz\boldsymbol{y})](\boldsymbol{x})) \big).$$

Here $\varphi[R\boldsymbol{y}/\exists z(z < u \land Sz\boldsymbol{y})](\boldsymbol{x}))$ is the formula obtained form $\varphi(R, \boldsymbol{x})$ by replacing all occurrences of subformula $R\boldsymbol{y}$ by $\exists z(z < u \land Sz\boldsymbol{y})$.

Let $\mathfrak{B}^* = (\mathfrak{B}, U^{\mathfrak{B}}, <^{\mathfrak{B}}, S^{\mathfrak{B}})$ be a countable elementary substructure of $\mathfrak{A}^*$, containing the tuple $\boldsymbol{a}$. Since $\mathfrak{A}^* \models \eta$, also $\mathfrak{B}^* \models \eta$ and therefore $S^{\mathfrak{B}}$ encodes the stages of $\varphi^{\mathfrak{B}}$. Since also $\mathfrak{B}^* \models \exists u Su\boldsymbol{a}$, it follows that $\boldsymbol{a}$ is contained in the least fixed point of $\varphi^{\mathfrak{B}}$, i.e., $\mathfrak{B} \models \psi(\boldsymbol{a})$.

A straightforward iteration of this argument gives the desired result for arbitrary nestings of fixed point operators, and hence for the entire fixed point logic FO + LFP. □

3 The tree model property

Tree width is an important notion in graph theory. Here we need a generalisation of this concept to arbitrary relational structures. For readers who are familiar with the notion of tree width in graph theory and the notion of the *Gaifman graph* of a structure we can simply say that the tree width of a structure is the tree width of its Gaifman graph. Here is a more detailed definition.

Definition 3.1. A structure $\mathfrak{B}$ (with universe B and arbitrary vocabulary τ) has *tree width* k if k is the minimal natural number satisfying the following condition. There exists a directed tree $T = (V, E)$ and a function

$$F : V \to \{X \subseteq B : |X| \leq k + 1\},$$

assigning to every node v of T a set $F(v)$ of at most $k+1$ elements of $\mathfrak{B}$, such that the following two conditions hold.

(i) For every τ-atom $\alpha(x_1, \dots, x_r)$ and every tuple $b_1, \dots, b_r$ such that $\mathfrak{B} \models \alpha(b_1, \dots, b_r)$ there exists a node v of T such that $\{b_1, \dots, b_r\} \subseteq F(v)$.

(ii) For every element b of $\mathfrak{B}$, the set of nodes $\{v \in V : b \in F(v)\}$ is connected (and hence induces a subtree of T).

For each node v of T, the set $F(v)$ induces a substructure $\mathfrak{F}(v) \subseteq \mathfrak{B}$ of cardinality at most $k+1$. (Since $F(v)$ may be empty, we also permit empty substructures.) $\langle T, (\mathfrak{F}(v)_{v \in T}) \rangle$ is called a *tree decomposition* of width k of $\mathfrak{B}$.

Definition 3.2. A set X of elements is *loosely k-guarded* in a structure $\mathfrak{B}$ if, for some $s \leq k$, there exists a tuple $b_1, \dots, b_s$ in $\mathfrak{B}$ such that $X \subseteq \{b_1, \dots, b_s\}$ and $\mathfrak{B} \models \alpha(b_1, \dots, b_s)$ where $\alpha(b_1, \dots, b_s) = \alpha_1 \land \cdots \land \alpha_m$ is a conjunction of atoms that guards $b_1, \dots, b_s$ in the sense of LGF (i.e. every pair b_i, b_j coexists in some conjunct of α).

A set is loosely guarded if it is loosely k-guarded for some k. A tuple $\boldsymbol{b} = (b_1, \dots, b_r)$ is loosely guarded in $\mathfrak{B}$ if the set $\{b_1, \dots, b_r\}$ is loosely guarded in $\mathfrak{B}$.

Lemma 3.3. *Let $\langle T, (\mathfrak{F}(v)_{v\in T})\rangle$ be a tree decomposition of $\mathfrak{B}$ and $X \subseteq B$ be a loosely guarded set in $\mathfrak{B}$. Then there exists a node v of T such that $X \subseteq F(v)$.*

Proof. Let $\alpha(b_1, \dots, b_s)$ be a guard of X in $\mathfrak{B}$. In the case that α is atomic the claim follows immediately from the definition of a tree decomposition. Otherwise $\alpha(\boldsymbol{b})$ may be a conjunction $\alpha_1 \wedge \cdots \wedge \alpha_m$ of atoms. For each component $b \in \boldsymbol{b}$, let V_b be the set of nodes v such that $b \in F(v)$. By the definition of a tree decomposition, each V_b induces a subtree of T. The fact that $\mathfrak{B} \models \alpha_1 \wedge \cdots \wedge \alpha_m$ and the occurrence conditions of variables in guards imply that for all $b, b' \in \boldsymbol{b}$ the intersection $V_b \cap V_{b'}$ is non-empty. It is a well-known result in graph theory that any collection of pairwise overlapping subtrees of a tree has a common node (see e.g. [14, p. 94]). Hence there is a node v of the T such that $F(v)$ contains all elements of $\boldsymbol{b}$ and therefore all elements of X. □

Guarded bisimulations. The notion of bisimulation from modal logic generalises in a straightforward way to various notions of guarded bisimulation that describe indistinguishability in guarded logics. We focus here on loosely guarded k-bisimulations, the appropriate notion for loosely guarded formulae of width at most k. The *width* of a formula is the maximal number of free variables in subformulae of ψ.

Definition 3.4. A loosely guarded k-bisimulation, or briefly, a *k-bisimulation* between two τ-structures $\mathfrak{A}$ and $\mathfrak{B}$ is a non-empty set I of finite partial isomorphisms $f : X \to Y$ from $\mathfrak{A}$ to $\mathfrak{B}$, where $X \subseteq A$ and $Y \subseteq B$ are loosely k-guarded sets, such that the following back and forth conditions are satisfied. For every $f : X \to Y$ in I,

forth: for every loosely k-guarded set $X' \subseteq A$ there exists a partial isomorphism $g : X' \to Y'$ in I such that f and g agree on $X \cap X'$.

back: for every loosely k-guarded set $Y' \subseteq B$ there exists a partial isomorphism $g : X' \mapsto Y'$ in I such that f^{-1} and g^{-1} agree on $Y \cap Y'$.

Two τ-structures $\mathfrak{A}$ and $\mathfrak{B}$ are *k-bisimilar* if there exists a k-bisimulation between them.

Definition 3.5. Let LGF^∞ be the infinitary variant of the loosely guarded fragment, extending LGF by the following rule for building new formulae: If $\Phi \subseteq \mathrm{LGF}^\infty$ is a set of formulae, then also $\bigvee \Phi$ and $\bigwedge \Phi$ are formulae of LGF^∞.

Adapting basic and well-known model-theoretic techniques to the present situation, one obtains the following result.

Theorem 3.6. *Let $\mathfrak{A}$ and $\mathfrak{B}$ be two τ-structures. The following are equivalent:*

(i) $\mathfrak{A}$ and $\mathfrak{B}$ are k-bisimilar.

(ii) For all sentences $\psi \in \mathrm{LGF}^\infty$ of width at most k, $\mathfrak{A} \models \psi \iff \mathfrak{B} \models \psi$.

The following simple observation relates μLGF to LGF^∞ and shows that k-bisimilar structures cannot be separated by guarded fixed point sentences of width k.

Proposition 3.7. *For each $\psi \in \mu$LGF of width k and each cardinal γ, there is a $\psi' \in \text{LGF}^\infty$, also of width k, which is equivalent to ψ on all structures up to cardinality γ.*

Proof. Consider a fixed point formula $[\text{LFP } R\boldsymbol{x} \,.\, \varphi(R, \boldsymbol{x})](\boldsymbol{x})$. For every ordinal α, there is a formula $\varphi_\alpha(\boldsymbol{x}) \in \text{LGF}^\infty$ that defines the stage α of the induction of φ. Indeed, let $\varphi_0(\boldsymbol{x}) = \mathit{false}$ and, for $\alpha > 0$, let $\varphi_\alpha(\boldsymbol{x}) := \varphi[R\boldsymbol{y}/\bigvee_{\beta<\alpha}\varphi_\beta(\boldsymbol{y})](\boldsymbol{x})$, that is, the formula that one obtains from $\varphi(R, \boldsymbol{x})$ if one replaces each atom $R\boldsymbol{y}$ (for any $\boldsymbol{y}$) by the formula $\bigvee_{\beta<\alpha}\varphi_\beta(\boldsymbol{y})$ which defines the stages prior to α. On structures of bounded cardinality, also the closure ordinal of any fixed-point formula is bounded. Hence for every cardinal γ there exists an ordinal α such that $[\text{LFP } R\boldsymbol{x} \,.\, \varphi(R, \boldsymbol{x})$ is equivalent to $\varphi_\alpha(\boldsymbol{x})$ on structures of cardinality at most γ. ☐

Unravelings of structures. The k-unraveling $\mathfrak{B}^{(k)}$ of a structure $\mathfrak{B}$ is defined inductively. We build a tree T, with functions F and G such that each $F(v)$ induces a loosely guarded substructure $\mathfrak{F}(v) \subseteq \mathfrak{B}$, each $G(v)$ induces a substructure $\mathfrak{G}(v) \subseteq \mathfrak{B}^{(k)}$ that is isomorphic to $\mathfrak{F}(v)$, and $\langle T, (\mathfrak{G}(v))_{v\in T}\rangle$ is a tree decomposition of $\mathfrak{B}^{(k)}$.

The root of T is λ, with $F(\lambda) = G(\lambda) = \varnothing$. Given a node v of T with $F(v) = \{a_1, \dots, a_r\}$ and $G(v) = \{a_1^*, \dots, a_r^*\}$ we create for every loosely k-guarded set $\{b_1, \dots, b_s\}$ in $\mathfrak{B}$ a successor node w of v such that $F(w) = \{b_1, \dots, b_s\}$ and $G(w)$ is a set $\{b_1^*, \dots, b_s^*\}$ which is defined as follows. For those i, such that $b_i = a_j \in F(v)$, put $b_i^* = a_j^*$ so that $G(w)$ has the same overlap with $G(v)$ as $F(w)$ has with $F(v)$. The other b_i^* in $G(w)$ are fresh elements.

Let $f_w : F(w) \to G(w)$ be the bijection taking b_i to b_i^* for $i = 1, \dots, s$. For $\mathfrak{F}(w)$ being the substructure of $\mathfrak{B}$ induced by $F(w)$, define $\mathfrak{G}(w)$ so that f_w is an isomorphism from $\mathfrak{F}(w)$ to $\mathfrak{G}(w)$. Finally $\mathfrak{B}^{(k)}$ is the structure with tree decomposition $\langle T, (\mathfrak{G}(v)_{v\in T})\rangle$.

Note that the k-unraveling of a structure has tree width at most $k - 1$.

Proposition 3.8. *$\mathfrak{B}$ and $\mathfrak{B}^{(k)}$ are k-bisimilar.*

Proof. Let I be the set of functions $f_v : F(v) \to G(v)$ for all nodes v of T. ☐

It follows that no sentence of width k in LGF^∞, and hence no sentence of width k in μLGF distinguishes between a structure and its k-unraveling. Since every satisfiable sentence in μLGF has a model of at most countable cardinality, and since the k-unraveling of a countable model is again countable we obtain the following tree model property for guarded fixed point logic.

Theorem 3.9 (Weak tree model property). *Every satisfiable sentence ψ in μLGF of width k has a countable model of tree width at most $k - 1$.*

Remark. Using more sophisticated arguments, one can establish a stronger variant of the tree model property with the additional condition that the branching of the underlying tree is bounded by $O(|\psi|^k)$ (see [10]).

4 Reduction to the monadic theory of trees

Let $\langle T, (\mathfrak{F}(v))_{v\in T}\rangle$ be a tree decomposition of width $k-1$ of a τ-structure $\mathfrak{D}$ with universe D. We want to describe $\mathfrak{D}$ by a tree with a finite set of labels. To this end, we fix a set K of $2k$ constants and choose a function $f : D \to K$ assigning to each element d of $\mathfrak{D}$ a constant $a_d \in K$ such that the following condition is satisfied. If v, w are adjacent nodes of T, then distinct elements of $\mathfrak{F}(v) \cup \mathfrak{F}(w)$ are always mapped to distinct constants of K.

For each constant $a \in K$, let $\mathcal{O}_a$ be the set of those nodes $v \in T$ at which the constant a occurs, i.e., for which there exists an element $d \in \mathfrak{F}(v)$ such that $f(d) = a$. Further, we introduce for each m-ary relation R of $\mathfrak{D}$ a tuple $\bar{R} := (R_{\boldsymbol{a}})_{\boldsymbol{a}\in K^m}$ of monadic relations on T with

$$R_{\boldsymbol{a}} := \{v \in T : \mathfrak{F}(v) \models Rd_1 \cdots d_m \text{ and } f(d_1) = a_1, \dots, f(d_m) = a_m\}.$$

The tree $T = (V, E)$ together with the monadic relations $\mathcal{O}_a$ and $\bar{R}$ (for $R \in \tau$) is called the tree structure $\mathcal{T}(\mathfrak{D})$ associated with $\mathfrak{D}$ (and, strictly speaking, with its tree decomposition and with K and f). Note that two occurrences of a constant $a \in K$ at nodes u, v of T represent the same element of $\mathfrak{D}$ if and only if a occurs in the label of *all* nodes on the link between u and v. (The link between two nodes u, v in a tree T is the smallest connected subgraph of T containing both u and v.)

An arbitrary tree $T = (V, E)$ with monadic relations $\mathcal{O}_a$ and $\bar{R}$ does define a tree decomposition of a structure $\mathfrak{D}$, provided that the following axioms are satisfied.

(1) At each node v, at most k of the predicates $\mathcal{O}_a$ are true.

(2) Neighbouring nodes agree on their common elements. For all m-ary relation symbols $R \in \tau$ we have the axiom

$$\mathit{consistent}(\bar{R}) := \bigwedge_{\boldsymbol{a}\in K^m} \forall x \forall y \Big(\Big(Exy \wedge \bigwedge_{a\in\boldsymbol{a}} (\mathcal{O}_a x \wedge \mathcal{O}_a y) \Big) \to (R_{\boldsymbol{a}} x \leftrightarrow R_{\boldsymbol{a}} y) \Big).$$

Note that these are first-order axioms over the vocabulary $\tau^* := \{E\} \cup \{\mathcal{O}_a : a \in K\} \cup \{\bar{R} : R \in \tau\}$. Given a tree structure $\mathcal{T}$ with underlying tree $T = (V, E)$ and monadic predicates $\mathcal{O}_a$ and $R_{\boldsymbol{a}}$ satisfying (1) and (2), we obtain a structure $\mathfrak{D}$ such that $\mathcal{T}(\mathfrak{D}) = \mathcal{T}$ as follows. For every constant $a \in K$, we call two nodes u, w of T a-equivalent if $\mathcal{T} \models \mathcal{O}_a v$ for all nodes v on the link between u and w. Clearly this is an equivalence relation on $\mathcal{O}_a^{\mathcal{T}}$. We write $[v]_a$ for the a-equivalence class of the node v. The universe of $\mathfrak{D}$ is the set of all a-equivalence classes of T for $a \in K$, i.e.,

$$D := \{[v]_a : v \in T,\ a \in K,\ \mathcal{T} \models \mathcal{O}_a v\}.$$

For every m-ary relation symbol R in τ, we define

$$\begin{aligned} R^{\mathfrak{D}} := \{([v_1]_{a_1}, \dots, [v_m]_{a_m}) :\ & \mathcal{T} \models R_{a_1 \dots a_m} v \text{ for some} \\ & \text{(and hence all) } v \in [v_1]_{a_1} \cap \cdots \cap [v_m]_{a_m}\}. \end{aligned}$$

Recall that on any directed tree $T = (V, E)$ we can express that U is the set of nodes on the link between x and y by a monadic second-order formula $\mathit{link}(U, x, y)$ constructed as follows. Let first

$$\begin{aligned} \mathit{connect}(U, x, y) := Ux \wedge Uy \wedge \exists r(Ur \wedge \forall z(Ezr \rightarrow \neg Uz) \\ \wedge \forall w \forall z(Ewz \wedge Uz \wedge z \neq r \rightarrow Uw)). \end{aligned}$$

This formula expresses that U contains the link from x to y. Then set

$$\mathit{link}(U, x, y) := \mathit{connect}(U, x, y) \wedge \forall Z(\mathit{connect}(Z, x, y) \rightarrow U \subseteq Z).$$

We now describe a translation from μLGF into monadic second-order logic. Given a formula $\varphi(x_1, \ldots, x_m) \in \mu$LGF we construct, for all tuples $\boldsymbol{a} = a_1, \ldots, a_m$ over K, monadic second-order formulae $\varphi_{\boldsymbol{a}}(z)$ with one free variable, which describe in the associated tree structure $\mathcal{T}(\mathfrak{D})$ the same properties of loosely guarded tuples as $\varphi(\boldsymbol{x})$ does in $\mathfrak{D}$. (We will make this statement more precise below). The translation is defined by induction as follows:

(1) If $\varphi(\boldsymbol{x})$ is an atom $Sx_{i_1} \cdots x_{i_m}$ then $\varphi_{\boldsymbol{a}}(z) := S_{\boldsymbol{b}}z$ where $\boldsymbol{b} = (a_{i_1}, \ldots, a_{i_m})$.

(2) If $\varphi = (x_i = x_j)$, let $\varphi_{\boldsymbol{a}}(z) = \mathit{true}$ if $a_i = a_j$ and $\varphi_{\boldsymbol{a}}(z) = \mathit{false}$ otherwise.

(3) If $\varphi = \eta \wedge \vartheta$, let $\varphi_{\boldsymbol{a}}(z) = \eta_{\boldsymbol{a}}(z) \wedge \vartheta_{\boldsymbol{a}}(z)$.

(4) If $\varphi = \neg\vartheta$, let $\varphi_{\boldsymbol{a}}(z) = \neg\vartheta_{\boldsymbol{a}}(z)$.

(5) If $\varphi = (\exists \boldsymbol{y} \,.\, \alpha(\boldsymbol{x}, \boldsymbol{y}))\eta(\boldsymbol{x}, \boldsymbol{y})$, let $\varphi_{\boldsymbol{a}}(z) :=$

$$\exists U \exists y \Big(\mathit{link}(U, y, z) \wedge \forall x(Ux \rightarrow \bigwedge_{a \in \boldsymbol{a}} \mathcal{O}_a x) \wedge \bigvee_{\boldsymbol{b}} \Big(\bigwedge_{b \in \boldsymbol{b}} \mathcal{O}_b y \wedge \alpha_{\boldsymbol{ab}}(y) \wedge \eta_{\boldsymbol{ab}}(y)\Big)\Big).$$

(6) If $\varphi = [\mathrm{LFP} \quad S\boldsymbol{x} \,.\, \eta(S, \boldsymbol{x})](\boldsymbol{x})$, let

$$\varphi_{\boldsymbol{a}}(z) := \forall \bar{S}\Big(\Big(\mathit{consistent}(\bar{S}) \wedge \bigwedge_{\boldsymbol{b}} \forall x(S_{\boldsymbol{b}}x \leftrightarrow \eta_{\boldsymbol{b}}(\bar{S}, x))\Big) \rightarrow S_{\boldsymbol{a}}z\Big).$$

Here $\bar{S}$ is a tuple $(S_{\boldsymbol{b}})_{\boldsymbol{b} \in K^m}$ of monadic predicates where m is the arity of S.

Theorem 4.1. *Let $\varphi(\boldsymbol{x})$ be a formula in μLGF and $\mathfrak{D}$ be a structure with tree decomposition $\langle T, (\mathfrak{F}(v))_{v \in T}\rangle$. For an appropriate set of constants K and a function $f : D \rightarrow K$, let $\mathcal{T}(\mathfrak{D})$ be the associated tree structure. Then, for every node v of T and every loosely guarded tuple $\boldsymbol{d} \subseteq \mathfrak{F}(v)$ with $f(\boldsymbol{d}) = \boldsymbol{a}$,*

$$\mathfrak{D} \models \varphi(\boldsymbol{d}) \iff \mathcal{T}(\mathfrak{D}) \models \varphi_{\boldsymbol{a}}(v).$$

Proof. We proceed by induction on φ. The only non-trivial cases are existential quantification and least fixed points.

Suppose that $\varphi(\boldsymbol{x}) = (\exists \boldsymbol{y} \,.\, \alpha(\boldsymbol{x}, \boldsymbol{y}))\eta(\boldsymbol{x}, \boldsymbol{y})$ and that $\mathfrak{D} \models \varphi(\boldsymbol{d})$. Then there exists a tuple $\boldsymbol{d}'$ such that $\mathfrak{D} \models \alpha(\boldsymbol{d}, \boldsymbol{d}') \wedge \eta(\boldsymbol{d}, \boldsymbol{d}')$.

Claim. *There exists a node w of T such that all components of $\boldsymbol{d} \cup \boldsymbol{d}'$ are contained in $\mathfrak{F}(w)$.*

By assumption $\boldsymbol{d}$ is loosely guarded, and α guards $\boldsymbol{d}'$ with respect to $\boldsymbol{d}$ (i.e. $\alpha(\boldsymbol{d}, \boldsymbol{d}')$ is a conjunction $\alpha_1 \wedge \cdots \wedge \alpha_m$ of atoms such that every $d' \in \boldsymbol{d}'$ coexists with every element from $\boldsymbol{d} \cup \boldsymbol{d}'$ in at least one conjunct of $\alpha(\boldsymbol{d}, \boldsymbol{d}')$. Hence the conjunction of α with any guard for $\boldsymbol{d}$ is a guard for $\boldsymbol{d}, \boldsymbol{d}'$. Thus $\boldsymbol{d}, \boldsymbol{d}'$ is loosely guarded and by Lemma 3.3, the claim follows.

Let $f(\boldsymbol{d}') = \boldsymbol{b}$. By induction hypothesis it follows that

$$\mathcal{T}(\mathfrak{D}) \models \bigwedge_{b \in \boldsymbol{b}} \mathcal{O}_b w \wedge \alpha_{\boldsymbol{ab}}(w) \wedge \eta_{\boldsymbol{ab}}(w).$$

Let U be the set of nodes on the link between v and w. Then the tuple $\boldsymbol{d}$ occurs in $\mathfrak{F}(u)$ for all nodes $u \in U$. It follows that

$$\mathcal{T}(\mathfrak{D}) \models \mathit{link}(U, v, w) \wedge \forall x (Ux \rightarrow \bigwedge_{a \in \boldsymbol{a}} \mathcal{O}_a x).$$

Hence $\mathcal{T}(\mathfrak{D}) \models \varphi_{\boldsymbol{a}}(v)$.

Conversely, if $\mathcal{T}(\mathfrak{D}) \models \varphi_{\boldsymbol{a}}(v)$ then there exists a node w such that the constants $\boldsymbol{a}$ occur at all nodes on the link between v and w (and hence correspond to the same tuple $\boldsymbol{d}$) and such that $\mathcal{T}(\mathfrak{D}) \models \alpha_{\boldsymbol{ab}}(w) \wedge \eta_{\boldsymbol{ab}}(w)$ for some tuple $\boldsymbol{b}$. By induction hypothesis this implies that $\mathfrak{D} \models \alpha(\boldsymbol{d}, \boldsymbol{d}') \wedge \eta(\boldsymbol{d}, \boldsymbol{d}')$ for some tuple $\boldsymbol{d}'$, hence $\mathfrak{D} \models \varphi(\boldsymbol{d})$.

Finally, let $\varphi(\boldsymbol{x}) = [\text{LFP} \;\; S\boldsymbol{x} \,.\, \eta(S, \boldsymbol{x})](\boldsymbol{x})$. By definition, $\mathfrak{D} \models \varphi(\boldsymbol{d})$ is true if and only if $\boldsymbol{d}$ is contained in every fixed point of the operator $\eta^{\mathfrak{D}}$, i.e. is in every relation S such that $S = \{\boldsymbol{c} : (\mathfrak{D}, S) \models \eta(S, \boldsymbol{c})\}$.

We first observe that, for loosely guarded tuples $\boldsymbol{d}$, this is equivalent to the seemingly weaker condition that $\boldsymbol{d}$ is contained in every S such that $\boldsymbol{c} \in S$ iff $\mathfrak{D} \models \eta(S, \boldsymbol{c})$ *for all loosely guarded tuples* $\boldsymbol{c}$. Indeed this is obvious, since $\eta(S, \boldsymbol{x})$ is a Boolean combination of quantifier-free formulae not involving $\boldsymbol{x}$, of positive atoms of the form $S\boldsymbol{u}$ where $\boldsymbol{u}$ is a recombination of the variables appearing in $\boldsymbol{x}$ and of formulae starting with a guarded existential quantifier. Therefore the truth values of $S\boldsymbol{c}$ for unguarded tuples $\boldsymbol{c}$ never matters for the question whether a given guarded tuple is in $\varphi^{\mathfrak{D}}(S)$.

Recall that the formula associated with $\varphi(\boldsymbol{x})$ and $\boldsymbol{a}$ is

$$\varphi_{\boldsymbol{a}}(z) := (\forall \bar{S})\Big(\Big(\mathit{consistent}(\bar{S}) \wedge \bigwedge_{\boldsymbol{b}} \forall x (S_{\boldsymbol{b}} x \leftrightarrow \eta_{\boldsymbol{b}}(\bar{S}, x))\Big) \rightarrow S_{\boldsymbol{a}} z\Big).$$

Consider any tuple $\bar{S} = (S_{\boldsymbol{b}})_{\boldsymbol{b} \in K^m}$ of monadic relations on $\mathcal{T}(\mathfrak{D})$ that satisifies the consistency axiom such that

$$(\mathcal{T}(\mathfrak{D}), \bar{S}) \models \bigwedge_{\boldsymbol{b}} \forall x (S_{\boldsymbol{b}} x \leftrightarrow \eta_{\boldsymbol{b}}(\bar{S}, x)).$$

This tuple $\bar{S}$ defines a relation S on $\mathfrak{D}$ such that for all nodes w of T and all tuples $\boldsymbol{c}$ in $\mathfrak{F}(w)$ with $f(\boldsymbol{c}) = \boldsymbol{b}$, we have $\boldsymbol{c} \in S$ iff $w \in S_{\boldsymbol{b}}$. Conversely each relation S on $\mathfrak{D}$ defines such a tuple $\bar{S}$ of monadic relations on $\mathcal{T}(\mathfrak{D})$ which describes the truth values of S on all loosely guarded tuples of $\mathfrak{D}$. Since

$\mathcal{T}(\mathfrak{D}) \models S_{\boldsymbol{b}}w \leftrightarrow \eta_{\boldsymbol{b}}(\bar{S}, w)$ it follows by induction hypothesis that $\mathfrak{D} \models S\boldsymbol{c} \leftrightarrow \eta(S, \boldsymbol{c})$. Further $\boldsymbol{d} \in S$ if and only if $v \in S_{\boldsymbol{a}}$.

Hence the formula $\varphi_{\boldsymbol{a}}(v)$ is true in $\mathcal{T}(\mathfrak{D})$ if and only if $\boldsymbol{d}$ is contained in all relations S over $\mathfrak{D}$ such that for all loosely guarded tuples $\boldsymbol{c}$, $\boldsymbol{c} \in S$ iff $\boldsymbol{c} \in \eta^{\mathfrak{D}}(S)$. By the remarks above this is equivalent to saying that $\boldsymbol{d}$ is in the least fixed point of $\eta^{\mathfrak{D}}$. □

Theorem 4.2. *The satisfiability problem for* μLGF *is decidable.*

Proof. Let ψ be a sentence in μLGF of vocabulary τ and width k. We translate ψ into a monadic second-order sentence ψ^* such that ψ is satisfiable if and only if there exists a countable tree $T = (V, E)$ with $T \models \psi^*$.

Fix a set K of $2k$ constants and let $\bar{\mathcal{O}}$ be the tuple of monadic relations $\mathcal{O}_a$ for $a \in K$. Further, for each m-relation symbol $R \in \tau$, let $\bar{R}$ be the tuple of monadic relation $R_{\boldsymbol{a}}$ where $\boldsymbol{a} \in K^m$. The desired monadic second-order sentence has the form

$$\psi^* := (\exists \bar{\mathcal{O}})(\exists \bar{R})(\chi \wedge \forall x \psi_{\varnothing}(x)).$$

Here χ is the first-order axiom expressing that the tree T expanded by the relations $\bar{\mathcal{O}}$ and $\bar{R}$ does describe a tree structure $\mathcal{T}(\mathfrak{D})$ associated to some τ-structure $\mathfrak{D}$. We have shown above that this can be done in first-order logic. The formula $\psi_{\varnothing}(x)$ is the translation of ψ (and the empty tuple of constants) into monadic second-order logic, as described by Theorem 4.1.

If ψ is satisfiable, then by Theorem 3.9, ψ has a countable model $\mathfrak{D}$ of tree width $k-1$. By Theorem 4.1, the associated tree structure $\mathcal{T}(\mathfrak{D})$ satisfies $\chi \wedge \forall x \psi_{\varnothing}(x)$, hence there exists a tree T such that $T \models \psi^*$. Conversely, if $T \models \psi^*$, then there exists an expansion $\mathcal{T} = (T, \bar{\mathcal{O}}, \bar{R})$ which satisfies χ and hence describes the tree decomposition of a τ-structure $\mathfrak{D}$. Since $\mathcal{T} \models \forall x \psi_{\varnothing}(x)$ it follows by Theorem 4.1 that $\mathfrak{D} \models \psi$.

The decidability of μLGF now follows by the decidability of the monadic second-order theory of countable trees, which has been established by Rabin [13]. □

5 Variations

Instead of reducing the satisfiability problem for μLGF to the monadic second-order theory of trees, we could also define a similar reduction to the μ-calculus with backward modalities and then invoke Vardi's decidability result for this logic [16].

Further, the reduction argument can be generalized to provide in fact a reduction from a (loosely) guarded variant of *second-order logic* over structures of bounded tree width to monadic second-order logic over trees. As a consequence we obtain the following decidability result.

Theorem 5.1.
For every constant k, the satisfiability problem for loosely guarded second-order formulae on structures of tree width at most k is decidable.

References

[1] H. Andréka, J. van Benthem, and I. Németi, *Modal languages and bounded fragments of predicate logic*, Journal of Philosophical Logic, 27 (1998), pp. 217–274.

[2] J. van Benthem, *Dynamic bits and pieces*, ILLC research report, University of Amsterdam, 1997.

[3] A. Emerson and C. Jutla, *The complexity of tree automata and logics of programs*, in Proc. 29th IEEE Symp. on Foundations of Computer Science, 1988, pp. 368–377.

[4] J. Flum, *On the (infinite) model theory of fixed-point logics*, in Models, algebras, and proofs, X. Caicedo and C. Montenegro, eds., no. 2003 in Lecture Notes in Pure and Applied Mathematics Series, Marcel Dekker, 1999, pp. 67–75.

[5] E. Grädel, *On the restraining power of guards*, Journal of Symbolic Logic. To appear.

[6] E. Grädel, P. Kolaitis, and M. Vardi, *On the decision problem for two-variable first-order logic*, The Bulletin of Symbolic Logic, 3 (1997), pp. 53–69.

[7] E. Grädel and M. Otto, *On logics with two variables*, Theoretical Computer Science, (1999). To appear.

[8] E. Grädel, M. Otto, and E. Rosen, *Undecidability results on two-variable logics*, in Proceedings of 14th Symposium on Theoretical Aspects of Computer Science STACS'97, Lecture Notes in Computer Science Nr. 1200, Springer, 1997, pp. 249–260.

[9] E. Grädel and E. Rosen, *On preservation theorems for two-variable logic*, Mathematical Logic Quarterly, (1999). To appear.

[10] E. Grädel and I. Walukiewicz, *Guarded fixed point logic*, in Proc. 14th IEEE Symp. on Logic in Computer Science, 1999.

[11] D. Kozen, *Results on the propositional μ-calculus*, Theoretical Computer Science, 27 (1983), pp. 333–354.

[12] M. Mortimer, *On languages with two variables*, Zeitschrift für Mathematische Logik und Grundlagen der Mathematik, 21 (1975), pp. 135–140.

[13] M. Rabin, *Decidability of second-order theories and automata on infinite trees*, Transactions of the AMS, 141 (1969), pp. 1–35.

[14] B. Reed, *Tree width and tangles: A new connectivity measure and some applications*, in Surveys in Combinatorics, R. Bailey, ed., Cambridge University Press, 1997, pp. 87–162.

[15] M. Vardi, *Why is modal logic so robustly decidable?*, in Descriptive Complexity and Finite Models, N. Immerman and P. Kolaitis, eds., vol. 31 of DIMACS Series in Discrete Mathematics and Theoretical Computer Science, AMS, 1997, pp. 149–184.

[16] ———, *Reasoning about the past with two-way automata*, in Automata, Languages and Programming ICALP 98, vol. 1443 of Springer Lecture Notes in Computer Science, 1998, pp. 628–641.