First-Order Logic with Two Variables and Unary Temporal Logic*

Kousha Etessami†
Bell Labs
Murray Hill, NJ
email: kousha@research.bell-labs.com

Moshe Y. Vardi‡
Department of Computer Science, Rice University
email: vardi@cs.rice.edu

Thomas Wilke
RWTH Aachen, Lehrstuhl für Informatik VII
52056 Aachen, Germany
email: wilke@informatik.rwth-aachen.de

*Part of the research reported here was conducted while the authors were visiting DIMACS as part of the Special Year on Logic and Algorithms.

†Part of this research conducted while this author was at: Basic Research in Computer Science (BRICS), Centre of the Danish National Research Foundation. The research was supported by the ESPRIT Long Term Research Programme of the EU under project number 20244 (ALCOM-IT).

‡During the preparation of this paper this author was partially supported by an NSF Grant.

Abstract

We investigate the power of first-order logic with only two variables over ω-words and finite words, a logic denoted by FO^2. We prove that FO^2 can express precisely the same properties as linear temporal logic with only the unary temporal operators: "next", "previously", "sometime in the future", and "sometime in the past", a logic we denote by $\mathsf{unary\text{-}TL}$. Moreover, our translation from FO^2 to $\mathsf{unary\text{-}TL}$ converts every FO^2 formula to an equivalent $\mathsf{unary\text{-}TL}$ formula that is at most exponentially larger, and whose operator depth is at most twice the quantifier depth of the first-order formula. We show that this translation is essentially optimal.

While satisfiability for full linear temporal logic, as well as for $\mathsf{unary\text{-}TL}$, is known to be **PSPACE**-complete, we prove that satisfiability for FO^2 is **NEXP**-complete, in sharp contrast to the fact that satisfiability for FO^3 has non-elementary computational complexity. Our **NEXP** upper bound for FO^2 satisfiability has the advantage of being in terms of the *quantifier depth* of the input formula. It is obtained using a small model property for FO^2 of independent interest, namely: a satisfiable FO^2 formula has a model whose "size" is at most exponential in the quantifier depth of the formula. Using our translation from FO^2 to $\mathsf{unary\text{-}TL}$ we derive this small model property from a corresponding small model property for $\mathsf{unary\text{-}TL}$. Our proof of the small model property for $\mathsf{unary\text{-}TL}$ is based on an analysis of $\mathsf{unary\text{-}TL}$ types.

1 Introduction

Over the past three decades a considerable amount of knowledge has accumulated regarding the relationship between first-order and temporal logic over both finite words and ω-words: the first-order expressible properties are exactly those expressible in temporal logic [Kam68, GPSS80, GHR94]; three variables suffice for expressing all the first-order expressible properties [Kam68, IK89]; while satisfiability for first-order logic with three variables has non-elementary computational complexity [Sto74], the satisfiability problem for temporal logic is **PSPACE**-complete [SC85]; moreover, there are classes of first-order formulas with three variables whose smallest equivalent temporal formulas require non-elementarily larger size (a consequence derivable from [Sto74]). In computer science the importance of this work stems from the practical relevance of temporal logic, which is used extensively today to specify and verify properties of reactive systems (see, e.g., [Pnu77] and [MP92]).

In this paper we provide a scaled down study of the relationship between first-order and temporal logic. Looking at first-order logic with only two variables, we show that the tight correspondence to temporal logic persists. We prove that first-order logic with two variables, denoted by FO^2, has precisely the same expressive power as temporal logic with the usual future and past unary temporal operators: "next", "previously", "sometime in the future", and "sometime in the past", but without the binary operators "until" and "since", a logic we denote by unary-TL. In other words, FO^2 coincides with the lowest level of the combined until/since hierarchy (which is known to be infinite [EW96]).

By contrast to the quite difficult proofs available for the correspondence between full first-order logic and temporal logic (cf., e.g., [Kam68, GPSS80, GHR94]), our proof that FO^2 = unary-TL is an easily understood inductive translation. In fact, our proof yields the following much stronger assertions: (1) FO^2 formulas can be translated to equivalent unary-TL formulas that are at most exponentially larger and whose operator depth is at most twice the quantifier depth of the first-order formula, and (2) the translation can be carried out in time polynomial in the size of the output formula.

We show that our translation is essentially optimal by exhibiting a sequence of FO^2 formulas that require exponentially larger unary-TL formulas. Thus, while with just three variables there is already a non-elementary gap between the succinctness of first-order logic and full temporal logic, FO^2 remains more succinct than unary-TL but not nearly as much: an exponential blowup is exactly what is necessary in the worst-case.

The same result that shows that satisfiability for temporal logic is **PSPACE**-complete, [SC85], also shows that satisfiability remains **PSPACE**-complete for unary-TL. We prove on the other hand that satisfiability for FO^2 is **NEXP**-complete. This again contrasts sharply with the non-elementary complexity of satisfiability for FO^3. Moreover, this is surprising given that FO^2 is exponentially more succinct than unary-TL, and that satisfiability for unary-TL is **PSPACE**-complete, leading one to expect that FO^2 satisfiability will be EX**PSPACE**-complete. Indeed, as a consequence of our **NEXP** bound it follows that FO^2 formulas that require "large" (exponentially bigger) unary-TL expressions necessarily have models that are "very small" (subexponential) with respect to the size of their unary-TL expression. Such "very small" models do not exist in general for unary-TL, as we can easily express with an $n^{O(1)}$ size unary-TL formula a "counter" whose smallest model has size 2^n.

An interesting and related aspect of our **NEXP** upper bound is that the time bound is only in terms of the *quantifier depth* of the FO^2 formula. This is because we prove our upper bound using an unusually strong small model property for FO^2, one which states that every satisfiable FO^2 formula has a model whose "size" is at most exponential in the quantifier depth of the given formula, rather than the size of the entire formula, which is how small model properties are usually formulated in the literature. For large but shallow formulas the gap between these quantities can make a significant difference.

It should be noted here that in a recent result Grädel, Kolaitis, and Vardi [GKV97] have shown that satisfiability for two-variable first-order formulas over *arbitrary* relational structures is computable in **NEXP** time. Their results also rely on a small model property. They prove that every satisfiable two-variable formula over arbitrary structures has a finite model of size at most exponential in the size of the formula, improving on a previous doubly-exponential bound obtained by Mortimer [Mor74]. Despite the similarity between the statement of their result and ours, the two are essentially incompatible and neither result implies the other. The reasons for this are two-fold. First, our results hold over words, i.e., over a unary vocabulary with built-in ordering. In particular, unlike arbitrary structures, over words we do not have a *genuine* finite model property: with two variables one can say that for every position in the word there is a greater position. Secondly, our "small" model property (Theorem 5) shows that every satisfiable formula has a model whose "size" is bounded exponentially by the quantifier depth of the formula, whereas the small model property of [GKV97] depends on the size of the entire formula. Moreover, the proof techniques used in the two results are completely different.

Our proof of the "small" model property for FO^2 is facilitated by our translation. It is enough to prove the same small model property for unary-TL (in terms of operator depth instead of quantifier depth) because our translation from FO^2 to unary-TL at most doubles the quantifier/operator depth. The existence of small models for unary-TL is established by an analysis of unary-TL types; these types behave quite differently than types for temporal logic in general.

FO^2 provides built-in binary predicates for a total order and a successor relation (besides free unary predicates). As further evidence of the robust correspondence between first-order and temporal logic we show that even when FO^2 is further restricted by removing the successor predicate, the relationship to temporal logic still persists: the resulting logic has exactly the same power as temporal logic with temporal operators "sometime in the future" and "sometime in the past" only (a logic which is traditionally referred to as "tense logic"). Moreover, we determine the complexity of satisfiability for this further restricted first-order logic, and the corresponding temporal logic, as well as their difference in succinctness.

All our results hold both for finite words and ω-words with only minor technical changes. We will mainly focus on the more interesting case of ω-words.

The paper is organized as follows. Section 2 introduces our notation and terminology. Section 3 presents the translation from FO^2 to unary-TL and shows it is optimal. Section 4 establishes **NEXP**-completeness of satisfiability for FO^2. In Section 5, we establish the small model property. Section 6 is concerned with FO^2 without "successor" and unary-TL without "next" and "previously". We conclude in Section 7.

2 Terminology and Notation

We assume p_0, p_1, ... is an infinite sequence of distinct symbols. For $m > 0$, we write σ_m for the set $\{p_0, \ldots, p_{m-1}\}$ and Σ_m for the power set of σ_m.

We interpret first-order and temporal formulas in ω-words over alphabets Σ_m as defined above.

The first-order signature we use contains unary predicates P_0, P_1, P_2, ... and in addition the built-in predicates "suc" for "successor" and "$<$" for "less than". Each ω-word u over an alphabet Σ_m is identified with a first-order structure $(\{0, 1, 2, \ldots\}, <, \mathrm{suc}, P_0^u, P_1^u, P_2^u, \ldots P_{m-1}^u)$ where $<$ and suc stand for the successor and order relation on the natural numbers and $P_i^u = \{j \mid p_i \in u_j\}$; here, as well as in the future, u_i stands for the letter at position i, and the first position has index 0.

We write $\top$ and $\bot$ denote true and false, respectively.

We fix two distinct variables, x and y, and define an FO^2 formula to be a first-order formula in the above signature in which only x and y occur as variables. An $\mathsf{FO}^2[<]$ formula is an FO^2 formula in which suc is not used.

Without loss of generality, we assume the atomic FO^2 formulas are $x = y$, $\mathrm{suc}(x, y)$, $\mathrm{suc}(y, x)$, $x < y$, $y < x$, $P_i x$, and $P_i y$ for $i \geq 0$. The atomic formulas involving $=$, suc, and $<$ will be referred to as *atomic order formulas*. We say that an FO^2 formula φ is a formula over ρ_m when the unary predicates in φ are among $P_0, P_1, \dots, P_{m-1}$.

We use traditional logical notation, adapted to our situation. When we introduce an FO^2 formula using the notation $\varphi(x)$, we mean that at most x occurs free in φ. Similarly, we use the notation $\varphi(y)$ and $\varphi(x, y)$. When a formula has been introduced as $\varphi(x)$ and we later on write $\varphi(y)$, then this expression stands for the formula which is obtained from φ by exchanging x and y. Symmetrically, when a formula has been introduced as $\varphi(y)$ and we later on write $\varphi(x)$, we mean the formula which is obtained from φ by exchanging x and y.

Given an FO^2 formula φ with at most one free variable, an ω-word u over Σ_m, and a position i, we write $u \models \varphi[i]$ if φ holds in the structure associated with u with respect to the variable assignment that maps the free variable to i (if there is one). When we consider an FO^2 formula φ with two free variables, then x and y are these variables, and we write $u \models \varphi[i, j]$ if φ holds in the structure associated with u with respect to the variable assignment that maps x to i and y to j.

A $\mathsf{unary\text{-}TL}$ formula is built from $p_0, p_1, p_2, \dots$, using the boolean connectives and the unary temporal operators $\oplus$ ("next"), $\ominus$ ("previously"), $\diamond\!\!\!\!+$ ("eventually" or "sometime in the future"), and $\diamond\!\!\!\!-$ ("sometime in the past"). An $\mathsf{unary\text{-}TL}[\diamond\!\!\!\!+]$ formula is a $\mathsf{unary\text{-}TL}$ formula in which neither $\oplus$ nor $\ominus$ is used. A $\mathsf{unary\text{-}TL}$ formula is said to be a formula over σ_m if the atomic propositions used are in σ_m.

The semantics of $\mathsf{unary\text{-}TL}$ is defined via a translation to FO^2. For every $\mathsf{unary\text{-}TL}$ formula φ, we define an FO^2 formula $\hat{\varphi}(x)$ according to the following rules.

- When $\varphi = p_i$ for some i, then $\hat{\varphi}(x) = P_i x$.
- When φ is of the form $\neg\psi$ or $\psi_1 \wedge \psi_2$, then $\hat{\varphi}(x) = \neg\hat{\psi}$ or $\hat{\varphi}(x) = \hat{\psi}_1 \wedge \hat{\psi}_2$, respectively.
- When φ is of the form $\oplus\psi$ or $\ominus\psi$, then $\hat{\varphi}(x) = \exists y(\mathrm{suc}(x, y) \wedge \hat{\psi}(y))$ or $\hat{\varphi}(x) = \exists y(\mathrm{suc}(y, x) \wedge \hat{\psi}(y))$, respectively.
- When φ is of the form $\diamond\!\!\!\!+\psi$ or $\diamond\!\!\!\!-\psi$, then $\hat{\varphi}(x) = \exists y(x < y \wedge \hat{\psi}(y))$ or $\hat{\varphi}(x) = \exists y(y < x \wedge \hat{\psi}(y))$, respectively.

For convenience in notation we write $(u, i) \models \varphi$ for an FO^2 formula φ to denote the fact that $u \models \hat{\varphi}[i]$.

We will say that FO^2 formulas $\varphi(x)$ and $\psi(x)$ are *equivalent* if $\{i \mid u \models \varphi[i]\} = \{i \mid u \models \psi[i]\}$ for all $\alpha \in \Sigma_m^\omega$, $m > 0$. A unary-TL formula φ is then said to be *equivalent* to an FO^2 formula $\psi(x)$ if $\hat{\varphi}(x)$ is equivalent to $\psi(x)$.

An FO^2 formula $\varphi(x, y)$ is said to be *satisfiable* if there is an ω-word u over Σ_m for some m, and natural numbers i and j, such that $u \models \varphi[i, j]$. A unary-TL formula φ is then said to be *satisfiable* if $\hat{\varphi}$ is satisfiable.

When we prove lower bounds on the size of formulas or smallest models (see, for instance, Theorems 3 and 7), we will interpret formulas over finite words. This makes our constructions easier and the statements somewhat stronger. The proofs carry over easily to the setting of ω-words.

The length of a formula φ is denoted by $|\varphi|$. The *quantifier depth* of an FO^2 formula is denoted by $\mathrm{qdp}(\varphi)$, while the *operator depth* of a unary-TL formula is denoted by $\mathrm{odp}(\varphi)$.

3 Unary-TL = FO^2

By definition, every unary-TL formula is equivalent to an FO^2 formula (linear in both size and operator/quantifier depth). That every FO^2 formula $\varphi(x)$ is equivalent to a unary-TL formula follows from the following much stronger statement.

Theorem 1 *Every FO^2 formula $\varphi(x)$ can be converted to an equivalent unary-TL formula φ' with $|\varphi'| \in 2^{\mathcal{O}(|\varphi|(\mathrm{qdp}(\varphi)+1))}$ and $\mathrm{odp}(\varphi') \leq 2\,\mathrm{qdp}(\varphi)$. Moreover, the translation is computable in time polynomial in $|\varphi'|$.*

Before proving the theorem, we note here the contrast between this theorem and what follows from the work in [Sto74]. Namely, there is a non-elementary lower bound in terms of blow-up in size for any translation of first-order formulas with three variables into temporal formulas.

This is because Stockmeyer showed that there are *star-free regular expressions* γ_n, of size polynomial in n, such that the smallest finite word satisfying γ_n has size $\mathrm{tower}(\Omega(\log n), n)$ where $\mathrm{tower}(k, l)$ is $2^{2^{\cdot^{\cdot^{l}}}}$ with a stack of 2's of height k.

Observe that given a star-free expression γ, one can easily write an FO^3 sentence $\hat{\gamma}$ which is equivalent (over finite words) to γ and has size linear in

γ. Inductively, one builds formulas $\gamma'(x,y)$ that hold when $x \leq y$ and the substring between positions x and y belongs to the language defined by γ, and one then sets

$$\hat{\gamma} = \exists x \exists y \forall z (\neg \mathrm{suc}(z,x) \wedge \neg \mathrm{suc}(y,z) \wedge \gamma'(x,y) \ . \tag{1}$$

The only interesting case is when the outermost operation in γ is concatenation, i. e., when γ is of the form $\gamma_1 \cdot \gamma_2$. In this case, one can set:

$$\gamma'(x,y) = \exists z (\hat{\gamma}(x,z) \wedge \exists x (\mathrm{suc}(z,x) \wedge \hat{\gamma}'(x,y))) \ . \tag{2}$$

We can thus conclude, from the fact that every satisfiable temporal formula has a model whose size is exponential in the size of the formula [SC85], that, by contrast to Theorem 1, any translation from FO^3 to temporal logic must incur non-elementary blow-up in size.

Proof of Theorem 1. Given an FO^2 formula $\varphi(x)$ the translation procedure works a follows. When $\varphi(x)$ is atomic, i. e., of the form $P_i x$, it outputs p_i. When $\varphi(x)$ is of the form $\psi_1 \vee \psi_2$ or $\neg\psi$—we say that $\varphi(x)$ is *composite*—it recursively computes ψ'_1 and ψ'_2, or ψ' and outputs $\psi'_1 \vee \psi'_2$ or $\neg\psi'$. The two cases that remain are when $\varphi(x)$ is of the form $\exists x \varphi^*(x)$ or $\exists y \varphi^*(x,y)$. In both cases, we say that $\varphi(x)$ is *existential*. In the first case, $\varphi(x)$ is equivalent to $\exists y \varphi^*(y)$ and, viewing x as a dummy free variable in $\varphi^*(y)$, this reduces to the second case.

In the second case, we can rewrite $\varphi^*(x,y)$ in the form

$$\varphi^*(x,y) = \beta(\chi_0(x,y), .., \chi_{r-1}(x,y), \xi_0(x), .., \xi_{s-1}(x), \zeta_0(y), .., \zeta_{t-1}(y)) \tag{3}$$

where β is a propositional formula, each formula χ_i is an atomic order formula, each formula ξ_i is an atomic or existential FO^2 formula with $\mathrm{qdp}(\xi_i) < \mathrm{qdp}(\varphi)$, and each formula ζ_i is an atomic or existential FO^2 formula with $\mathrm{qdp}(\zeta_i) < \mathrm{qdp}(\varphi)$.

In order to be able to recurse on subformulas of φ we have to separate the ξ_i's from the ζ_i's. We first introduce a case distinction on which of the subformulas ξ_i hold or not. We obtain the following equivalent formulation for φ:

$$\bigvee_{\overline{\gamma} \in \{\top,\bot\}^s} (\bigwedge_{i<s} (\xi_i \leftrightarrow \gamma_i) \wedge \exists y \, \beta(\chi_0, \ldots, \chi_{r-1}, \gamma_0, \ldots, \gamma_{s-1}, \zeta_0, \ldots, \zeta_{t-1})) \ .$$

We proceed by a case distinction on which order relation holds between x and y. We consider five mutually exclusive cases, determined by the following formulas, which we call *order types*: $x = y$, $\mathrm{suc}(x, y)$, $\mathrm{suc}(y, x)$, $x < y \wedge \neg\mathrm{suc}(x, y)$, $y < x \wedge \neg\mathrm{suc}(y, x)$. When we assume that one of these order types is true, each atomic order formula evaluates to either $\top$ or $\bot$, in particular, each of the χ_i's evaluates to either $\top$ or $\bot$; we will denote this truth value by χ_i^τ. We can finally rewrite φ as follows, where Υ stands for the set of all order types:

$$\bigvee_{\overline{\gamma}\in\{\top,\bot\}^s} (\bigwedge_{i<s}(\xi_i \leftrightarrow \gamma_i) \wedge \bigvee_{\tau\in\Upsilon} \exists y(\tau \wedge \beta(\chi_0^\tau, \ldots, \chi_{r-1}^\tau, \overline{\gamma}, \overline{\zeta}))) \enspace .$$

If τ is an order type, $\psi(x)$ an FO^2 formula, and ψ' an equivalent $\mathsf{unary\text{-}TL}$ formula, there is an obvious way to obtain a $\mathsf{unary\text{-}TL}$ formula $\tau\langle\psi\rangle$ equivalent to $\exists y(\tau \wedge \psi(y))$, as displayed in the following table.

τ	$x = y$	$\mathrm{suc}(x, y)$	$\mathrm{suc}(y, x)$	$x < y \wedge \neg\mathrm{suc}(x, y)$	$y < x \wedge \neg\mathrm{suc}(y, x)$
$\tau\langle\psi'\rangle$	ψ	$\oplus\psi$	$\ominus\psi$	$\oplus\Diamond\!\!\!\!+\psi$	$\ominus\Diamond\!\!\!\!-\psi$

Our procedure will therefore recursively compute ξ_i' for $i < s$ and $\zeta_i(x)'$ for $i < t$ and output

$$\bigvee_{\overline{\gamma}\in\{\top,\bot\}^s} \left(\bigwedge_{i<s}(\xi_i' \leftrightarrow \gamma_i) \wedge \bigvee_{\tau\in\Upsilon} \tau \left\langle\beta(\chi_0^\tau, .., \chi_{r-1}^\tau, \overline{\gamma}, \zeta_0(x)', \ldots, \zeta_{t-1}(x)')\right\rangle\right) \enspace . \tag{4}$$

Now we verify that $|\varphi'|$ and $\mathrm{odp}(\varphi')$ are bounded as stated in the theorem. That $\mathrm{odp}(\varphi') \leq 2\,\mathrm{qdp}(\varphi)$ is easily seen. The proof that $|\varphi'| \leq 2^{c|\varphi|(\mathrm{qdp}(\varphi)+1)}$ for some constant c is inductive on the quantifier depth of φ. The base case is trivial, and the only interesting case in the inductive step is when φ is of the form $\exists y\varphi^*(x, y)$ as above. In this case, we have to estimate the length of (4). There are $2^s \leq 2^{|\varphi|}$ possibilities for $\overline{\gamma}$ in (4), and each disjunct in (4) has length at most $d\,|\varphi|\,\max_{i<s,j<t}(|\xi_i'|, |\zeta_j'|)$ for some constant d. By induction hypothesis, the latter is bounded by $d\,|\varphi|\,2^{c|\varphi|\mathrm{qdp}(\varphi)}$, which implies the claim, provided c is chosen large enough.

It is straightforward to verify that our translation to φ' can be computed in time polynomial in $|\varphi'|$. ∎

Obviously, $\mathsf{unary\text{-}TL}[\Diamond\!\!\!\!+]$ can easily be translated into $\mathsf{FO}^2[<]$. A slight modification of the translation from FO^2 to $\mathsf{unary\text{-}TL}$ described in the above proof yields the reverse translation, i. e., $\mathsf{unary\text{-}TL}[\diamondplus] = \mathsf{FO}^2[<]$. In fact, the translation becomes simpler, because we only need to distinguish three order types ($x = y$, $x < y$, and $y < x$). In particular, the operator depth of the translated formula is bounded by the quantifier depth of the given formula.

We have:

Theorem 2 *Every $\mathsf{FO}^2[<]$ formula $\varphi(x)$ can be converted to an equivalent $\mathsf{unary\text{-}TL}[\diamondplus]$ formula φ' with $|\varphi'| \in 2^{\mathcal{O}(|\varphi|\,(\mathrm{qdp}(\varphi)+1)}$ and $\mathrm{odp}(\varphi') \leq \mathrm{qdp}(\varphi)$.*

An exponential blow-up, as incurred in the translation of Theorems 1 and 2, is necessary:

Theorem 3
1. *There is a sequence $(\varphi_n)_{n\geq 1}$ of $\mathsf{FO}^2[<]$ sentences of size $\mathcal{O}(n)$ such that the shortest temporal formulas equivalent to φ_n have size $2^{\Omega(n)}$.*
2. *There is a sequence $(\varphi'_n)_{n\geq 1}$ of FO^2 sentences in one propositional variable of size $\mathcal{O}(n^2)$ such that the shortest temporal formulas equivalent to φ'_n have size $2^{\Omega(n)}$.*

Observe that as usual the successor predicate "suc" can compensate for a bounded vocabulary.

Proof. The formula φ_n is a formula that defines the following property: "any two positions that agree on $p_0, \ldots, p_{n-1}$ also agree on p_n". This is easily defined in FO^2 within size linear in n:

$$\varphi_n = \forall x \forall y ((\bigwedge_{i<n} (P_i x \leftrightarrow P_i y)) \rightarrow (P_n x \leftrightarrow P_n y)).$$

To prove that the shortest temporal formulas equivalent to φ_n have size $2^{\Omega(n)}$, we make use of the tight connection between formulas and automata. Given a temporal formula φ and an alphabet Σ_m, the set $\{u \in \Sigma_m^\omega \mid u \models \varphi[0]\}$ is an ω-language over Σ_m. From [VW86], we know that every such language is recognized by a non-deterministic generalized Büchi automaton[1] with $2^{\mathcal{O}(|\varphi|)}$ states, so that it is enough to show that every generalized Büchi automaton for $L_n = \{u \in \Sigma_{n+1}^\omega \mid u \models \varphi_n\}$ requires at least 2^{2^n} states.

[1] A generalized Büchi automaton uses a family of final state sets instead of a single final state set. A run of such an automaton is accepting if every final state set is visited infinitely often.

Suppose $\mathfrak{A}$ is a generalized Büchi automaton recognizing L_n. Let $a_0, \dots, a_{2^n-1}$ be any sequence of the 2^n symbols of the alphabet Σ_n. For every subset K of $\{0, \dots, 2^n - 1\}$ let w_K be the word $b_0 \dots b_{2^n-1}$ with $b_i = a_i$ if $i \in K$ and else $b_i = a_i \cup \{p_n\}$. Notice that there are 2^{2^n} such words. Also, $w_K^\omega \models \varphi_n$ and $w_K w_{K'}^\omega \not\models \varphi_n$ for $K \neq K'$. Therefore, if $K \neq K'$ and q_K and $q_{K'}$ are the states assumed by $\mathfrak{A}$ in accepting runs for w_K^ω and $w_{K'}^\omega$ after 2^n steps, then q_K and $q_{K'}$ have to be distinct, i. e., $\mathfrak{A}$ needs at least 2^{2^n} states.

For the proof of part 2, let n be an arbitrary natural number and consider the property that contains an ω-word u when the following holds for all positions i and j: if $u_i = u_{i+1} = u_j = u_{j+1} = \emptyset$ and $u_{i+2(k+1)} = u_{j+2(k+1)}$ for all $k < n$, then $u_{i+2(n+1)} = u_{j+2(n+1)}$. By a similar argument as before, one shows that every temporal formula expressing this property has size $\Omega(2^n)$. On the other hand, the property is easily expressed by an FO^2 formula in one propositional variable. The successor predicate is used to access the positions in the neighborhood of a given position; note that $p_0 \in u_{i+k}$ iff $u \models \psi_k[i]$ where $\psi_0(x) = P_0 x$ and $\psi_{l+1} = \exists y(\mathrm{suc}(x, y) \wedge \psi(y))$. ∎

4 Small Model Properties

In this section, we derive several small model properties that we will later use to upper bound the complexity of the satisfiability problem for FO^2, unary-TL, and unary-TL[$\oplus$]. For FO^2, we will obtain two orthogonal small model properties, one in terms of quantifier depth and one in terms of formula length: for long but shallow formulas the former gives the better bound whereas for formulas with a large quantifier depth compared to their lengths the latter gives the better bound.

4.1 Quantifier Depth

Theorem 1 tells us that every FO^2 formula of depth k can be translated into an equivalent unary-TL formula of depth $2k$. Thus a small model property for unary-TL in terms of quantifier depth will immediately give a corresponding small model property for FO^2.

We prove:

Theorem 4 *Every satisfiable* unary-TL *formula φ in m propositional variables has a model of the form uv^ω where the sizes of u and v are bounded by $2^{\mathcal{O}((\mathrm{odp}(\varphi)+1)^2 m)}$.*

And thus, by Theorem 1:

Theorem 5 *Every satisfiable FO^2 formula $\varphi(x)$ in m unary predicates has a model uv^ω where the sizes of u and v are bounded by $2^{\mathcal{O}((\mathrm{qdp}(\varphi)+1)^2 m)}$.*

We first introduce some terminology and sketch a proof of Theorem 4 before going into details.

We fix $m > 0$ and consider only formulas over $p_0, \ldots, p_{m-1}$ and ω-words over Σ_m. Let $k, k' \geq 0$. We say that a unary-TL formula φ is of *depth (at most)* (k, k') if it is of depth (at most) k in $\Diamond\!\!\!\!+$ and $\Diamond\!\!\!\!-$ and of depth (at most) k' in $\oplus$ and $\ominus$. Given an ω-word w and a position $i \geq 0$, the *(k, k')-type of i in w*, denoted $\tau^w_{k,k'}(i)$, is the set of all unary-TL formulas of depth at most (k, k') that hold in w at i. This means that $w \models \varphi$ if and only if $\varphi \in \tau^w_{k,k'}(0)$ for every formula φ of operator depth at most (k, k'). It is thus enough to show that for every ω-word w there exist u and v of size bounded by $2^{\mathcal{O}((k+k'+1)^2 m)}$ such that $\tau^w_{k,k'}(0) = \tau^{w'}_{k,k'}(0)$ for $w' = uv^\omega$. In order to establish this, we first show that for every ω-word w one can find u and v such that w and uv^ω agree on the types of position 0 and such that u and v are bounded polynomially in the number of types that occur in w. We then show that the number of types occurring in a given ω-word is bounded by $2^{\mathcal{O}((k+k'+1)^2 m)}$.

Given positions i and j, we write $\tau^w_{k,k'}(i, j)$ for the set of types that occur between i and j, that is, we set

$$\tau^w_{k,k'}(i, j) = \{\tau^w_{k,k'}(i') \mid i \leq i' \leq j\} \tag{5}$$

We also allow $j = \infty$ in (5); in this case, $\tau^w_{k,k'}(i, \infty) = \{\tau^w_{k,k'}(i') \mid i \leq i'\}$. Furthermore, we write $\tau^w_{k,k'}(\infty)$ for the set of types that occur infinitely often, that is, we set

$$\tau^w_{k,k'}(\infty) = \{\tau^w_{k,k'}(l) \mid \exists^\infty l' \, (\tau^w_{k,k'}(l) = \tau^w_{k,k'}(l'))\} \ . \tag{6}$$

For fixed parameters k and k', there are only finitely many different types. Since the set of formulas of depth (k, k') is closed under boolean combinations, we thus get:

Remark 1 *Let $k, k' \geq 0$, $w \in \Sigma_m^\omega$, and $i \geq 0$.*

Then there exists a unary-TL *formula φ of depth (k, k') such that for all ω-words $w' \in \Sigma_m^\omega$ and all $j \geq 0$ we have:*

$$\tau_{k,k'}^w(i) = \tau_{k,k'}^{w'}(j) \quad \textit{iff} \quad (w', j) \models \varphi \ . \tag{7}$$

It is easy to see that $\oplus$ and $\ominus$ can always be "moved in" without increasing the operator-depth:

Remark 2 *Every* unary-TL *formula of depth (k, k') is equivalent to a* unary-TL *formula of the same depth where each occurrence of $\oplus$ or $\ominus$ is followed by another occurrence of $\oplus$ or $\ominus$ or by a propositional variable p_i.*

The following lemma establishes that the $(k+1, k')$-type of a position i in a given word w is determined uniquely by i's local neighborhood, the (k, k')-types that occur to its right, and the (k, k')-types that occur to its left.

Lemma 1 *Let w and w' be ω-words over Σ_m and i, $i' \geq 0$.*

Then

$$\tau_{0,k'}^w(i) = \tau_{0,k'}^{w'}(i') \quad \textit{iff} \quad w_{i-k'} \dots w_i \dots w_{i+k'} = w'_{i'-k'} \dots w'_{i'} \dots w'_{i'+k'} \tag{8}$$

where, by convention, $w_j = \$$ and $w'_j = \$$ for $j < 0$ ($\$$ being a special symbol), and

$$\tau_{k+1,k'}^w(i) = \tau_{k+1,k'}^{w'}(i') \quad \textit{iff} \quad \begin{cases} \tau_{0,k'}^w(i) = \tau_{0,k'}^{w'}(i') \ , \\ \tau_{k,k'}^w(0, i-1) = \tau_{k,k'}^{w'}(0, i'-1) \ , \\ \tau_{k,k'}^w(i+1, \infty) = \tau_{k,k'}^{w'}(i'+1, \infty) \ . \end{cases} \tag{9}$$

Proof. (8) is clear: A depth k' formula that uses no $\diamondplus$ operator can describe completely the content of the k'-neighborhood of the current position, and nothing more.

To prove (9) we proceed by induction on k. The base case, $k = 0$, is immediate. Assume true for k, we prove the claim for $k + 1$.

($\Rightarrow$) If $\tau_{k+1,k'}^w(i) = \tau_{k+1,k'}^{w'}(i')$, then, in particular, (w, i) and (w', i') agree on all depth $(0, k')$ formulas, and thus $\tau_{0,k'}^w(i) = \tau_{0,k'}^{w'}(i')$.

To show $\tau_{k,k'}^w(i+1, \infty) \subseteq \tau_{k,k'}^{w'}(i'+1, \infty)$, let $\tau' = \tau_{k,k'}^w(j)$ for some $j > i$ and assume φ is the formula from Remark 1 that describes τ'. Then $\diamondplus\varphi$ is a depth $(k+1,k')$ formula that holds at i in w, hence, by assumption, it holds at i' in w'. Therefore, there exists $j' > i'$ at which φ holds in w', which means $\tau' \in \tau^{w'}_{k,k'}(i'+1,\infty)$. A symmetric proof shows that $\tau^{w}_{k,k'}(i+1,\infty) \supseteq \tau^{w'}_{k,k'}(i'+1,\infty)$, and thus $\tau^{w}_{k,k'}(i+1,\infty) = \tau^{w'}_{k,k'}(i'+1,\infty)$. A similar proof shows that $\tau^{w}_{k,k'}(0,i-1) = \tau^{w'}_{k,k'}(0,i'-1)$.

($\Leftarrow$) Assume that the three equalities on the right hand side of (9) hold. We want to show that

$$(w,i) \models \varphi \qquad \text{iff} \qquad (w',i') \models \varphi \ , \tag{10}$$

for every formula φ of depth $(k+1,k')$. Recall Remark 2. This states, in particular, that every unary-TL formula of depth $(k+1,k')$ is equivalent to a boolean combination of formulas of depth $(k+1,k')$ starting with $\diamond\!\!\!\!+$ or $\diamond\!\!\!\!-$, and formulas of depth $(0,k')$. We can thus restrict our attention to such formulas. Moreover, it is enough to consider formulas where the outermost connective is a temporal operator, as (10) is preserved under boolean connectives.

First, assume the outermost connective of φ is $\oplus$ or $\ominus$. Then φ is a depth $(0,k')$ formula. Thus, since by assumption $\tau^{w}_{0,k'}(i) = \tau^{w'}_{0,k'}(i')$, $\varphi \in \tau^{w}_{k+1,k'}(i)$ iff $\varphi \in \tau^{w'}_{k+1,k'}(i')$.

Second, assume the outermost connective of φ is $\diamond\!\!\!\!+$, that is, $\varphi = \diamond\!\!\!\!+\varphi^*$ for some φ^*. Now $(w,i) \models \varphi$ iff there exists a $j > i$ such that $\varphi^* \in \tau^{w}_{k,k'}(j)$. Hence, since by assumption $\tau^{w}_{k,k'}(i+1,\infty) = \tau^{w'}_{k,k'}(i'+1,\infty)$, we have $\varphi^* \in \tau^{w}_{k+1,k'}(j')$ for some $j' > i'$, which implies $\varphi \in \tau^{w'}_{k+1,k'}(i')$. The case when $\varphi = \diamond\!\!\!\!-\varphi^*$ is symmetric. ∎

Using Lemma 1, we can now establish the following lemma which shows how to collapse ω-words in order to get "smaller" ω-words without changing the type structure of the ω-word in an essential way. In the following lemma k' will be fixed, and we adopt the shorthand notation τ^w_k for $\tau^w_{(k,k')}$.

Lemma 2 *Let $w \in \Sigma_m^\omega$ and assume i and j are positions such that $i < j$ and $\tau^w_k(i) = \tau^w_k(j)$.*

1. *Let $w' = w_0 w_1 \ldots w_i w_{j+1} w_{j+2} \ldots$*

   Then
   $$\begin{aligned} \tau^{w'}_k(l) &= \tau^w_k(l) && \text{for } l \le i, \\ \tau^{w'}_k(l) &= \tau^w_k(l+(j-i)) && \text{for } l > i. \end{aligned}$$

2. *Further assume that* $\tau_k^w(i,j-1) = \tau_k^w(j,\infty)$, *and let* $w' = w_0 \dots w_i(w_{i+1}\dots w_j)^\omega$. *Then*

$$\begin{array}{rcll} \tau_k^{w'}(l) &=& \tau_k^w(l) & \textit{for } l \leq i, \\ \tau_k^{w'}(i + r(j-i) + s) &=& \tau_k^w(i+s) & \textit{for } r \geq 0,\ 0 \leq s < j - i. \end{array}$$

Proof. We prove part 1 by induction on k. Base case, $k = 0$. When we cut out a piece of a word, we don't change any of the characters we didn't cut out, and moreover the characters in the k'-neighborhoods of a point remain the same, thus we don't change $(0,k')$-types of any point.

Assume true for k. Suppose $\tau_{k+1}^w(i) = \tau_{k+1}^w(j)$. From (9) it follows that

$$\tau_k^w(i+1, j-1) \subseteq \tau_k^w(0, i-1)\ , \tag{11}$$

$$\tau_k^w(i+1, j-1) \subseteq \tau_k^w(j+1, \infty)\ . \tag{12}$$

Let $\pi(l)$ be the mapping defined by:

$$\pi(l) = \begin{cases} l & \text{if } l \leq i, \\ l + (j-i) & \text{otherwise.} \end{cases}$$

By the inductive hypothesis we know that for all l, $\tau_k^{w'}(l) = \tau_k^w(\pi(l))$. But then $\tau_k^{w'}(l+1,\infty) = \{\tau_k^w(\pi(m)) \mid m > l\} = \tau_k^w(\pi(l)+1,\infty)$, the last equality following from containment (12). Similarly, using containment (11), we have $\tau_k^{w'}(0, l-1) = \{\tau_k^w(\pi(m)) \mid m < l\} = \tau_k^w(0, \pi(l) - 1)$. But then by (9) we have $\tau_{k+1}^{w'}(l) = \tau_{k+1}^w(\pi(l))$, which is what we wanted to prove.

The proof of part 2 is again by induction on k. Base case, $k = 0$. For $l \leq i$, given that i and j have the same k'-neighborhood, the k'-neighborhood of position l in w' is the same as the k'-neighborhood of l in w. Also, for $l = i + r(j-i) + s$, by the same fact, l has the same k'-neighborhood as $i + s$. The base case then follows from (8).

Suppose true for k, we prove the claim for $k+1$. First note that $(k+1,k')$-types constitute a refinement of (k,k')-types, meaning that two positions with the same $(k+1,k')$-type have the same (k,k')-type. Thus, by the inductive hypothesis, we know that $\tau_k^{w'}(l) = \tau_k^w(l)$ for $l \leq i$ and $\tau_k^{w'}(i + r(j-i) + s) = \tau_k^w(i+s)$ for $r \geq 0$ and $0 \leq s < j - i$.

It follows that for every $l \leq i$, we have $\tau_k^{w'}(l+1,\infty) = \tau_k^w(l+1,\infty)$ by the induction hypothesis and the general assumption that $\tau_k^w(i,j-1) = \tau_k^w(j,\infty)$, Thus, for $l \leq i$, $\tau_k^{w'}(l+1,\infty) = \tau_k^w(l+1,\infty)$. In a similar way it follows that $\tau_k^{w'}(0,l-1) = \tau_k^w(0,l-1)$. Thus, by (9), it follows that $\tau_{k+1}^w(l) = \tau_{k+1}^{w'}(l)$.

A similar proof shows that $\tau^{w'}_{k+1}(i + r(j - i) + s) = \tau^{w}_{k+1}(i + s)$, for $r \geq 0$ and $0 \leq s < j - i$. ■

The next lemma states that in ultimately periodic words—words of the form uv^ω—types occur "ultimately periodically." For later use, we primarily phrase this lemma for FO^2; the $\mathsf{unary\text{-}TL}$ version then follows immediately.

Lemma 3 *Let $\varphi(x)$ be an FO^2 formula, u and v words with $|v| > 2$, $w = uv^\omega$, and $d = \mathrm{qdp}(\varphi)$.*

1. *For $r \geq 0$ and $0 \leq s < |v|$,*

$$w \models \varphi[|u| + 2d|v| + s] \qquad \text{iff} \qquad w \models \varphi[|u| + (2d + r)|v| + s] \ . \quad (13)$$

2. *In particular, if, in addition, $k + k' \leq d$, then for $r \geq 0$ and $0 \leq s < |v|$,*

$$\tau^{w}_{k,k'}(|u| + 2d|v| + s) = \tau^{w}_{k,k'}(|u| + (2d + r)|v| + s) \ . \quad (14)$$

3. *In particular, if $\varphi(x) = \exists y \varphi^*(x, y)$ and $uv^\omega \models \varphi[i]$ with $i < |uv^{2d+1}|$, then there exists $j \leq |uv^{2d+3}|$ such that $uv^\omega \models \varphi^*[i, j]$.*

Proof. Part 3 follows from the proof of part 1. Part 2 is an immediate consequence of the definition of the semantics of $\mathsf{unary\text{-}TL}$. The proof for part 1 is by induction on the quantifier depth d.

Base case: When $\varphi(x)$ is quantifier free, the only thing we can say about the only variable x is which predicates hold at x, and clearly the predicates that hold at a position $j = |u| + r|v| + s$ are exactly those that hold at $|u| + s$ (simply because we are at the same position in the word v).

Inductive case: Assume true for d, we prove the assertion for $d + 1$. Our formula $\varphi(x)$ of depth $d + 1$ is a boolean combination of formulas $\varphi'(x)$ of the form:

$$\exists y \beta(\chi_1, \ldots, \chi_l, \psi_1(x), \ldots, \psi_m(x), \gamma_1(y), \ldots, \gamma_c(y))$$

where β denotes a boolean combination of the given formulas and each $\chi_i(x, y)$ is an atomic order relation (i.e., one of $x < y$, $\mathrm{suc}(y, x)$, etc.). We will argue that part 1 holds for formulas of the form φ' and it will follow that it holds for φ as well because the "iff" in part 1 is preserved under boolean combination.

($\Leftarrow$) Suppose $\varphi[j]$ holds for $j = |u| + (2(d+1)+r)|v| + s$, where $r \geq 0$ and $0 \leq s < |v|$. Then there is a witness for y, namely a position k at which $\beta(\chi_1[j,k], \ldots, \chi_l[j,k], \psi_1[j], \ldots, \psi_m[j], \gamma_1[k], \ldots, \gamma_c[k])$ holds. We consider several cases based on the location of k in uv^ω. Let $j'_{d+1} = |u| + 2(d+1)|v| + s$. We want to show that $\varphi[j'_{d+1}]$ also holds.

1. $j \leq k$: In this case we know by the inductive hypothesis that j'_{d+1} satisfies the same ψ_i's as j, and that $j'_{d+1} + (k-j)$ satisfies the same γ_i's as k, and thus is a witness for j'_{d+1} just as k is for j, because their juxtaposition is exactly the same.

2. $|u| + (2d+1)|v| \leq k < j$: In this case, since $|v| > 2$ it can be seen that since k is a witness for j, then so is $j - ((j-k)mod|v|)$, because this point satisfies the same γ_i's as k. Thus, we also find our withness for j'_{d+1} as $j'_{d+1} - ((j-k)mod|v|)$. We can do this because there is an extra copy of v preceeding the point k which, by the inductive hypothesis, satisfies the same γ_i's at each position as the copies of v that succeed it, including the copies preceeding the point j.

3. $k < |u| + (2d+1)|v|$: In this case, we can fix k as a witness for both j and j'_{d+1} because, given that $|v| > 2$, the order type of (k, j'_{d+1}) and (k, j) is the same.

($\Rightarrow$) Suppose that $\varphi[j]$ holds for j where $|uv^{2d+2}| \leq j < |uv^{2d+3}|$. Then the claim is that $\varphi[j']$ holds for $j' = j + r|v|$ and for all r. This is again split into cases based on the location of the witness k.

1. $j \leq k$: But then $j + r|v|$ has a witness at $k + r|v|$.

2. $|u| + (2d+1)|v| \leq k < j$: In this case again $j + r|v|$ has $k + r|v|$ as a witness.

3. $k < |u| + (2d+1)|v|$: Now again as in the third case above k is a witness for both j and $j + r|v|$ because, given that $|v| > 2$, the order types of (k, j) and $(k, j + r|v|)$ are the same. ∎

From the previous lemmas, we conclude:

Lemma 4 *Let w be an ω-word over Σ_m and t the number of (k,k')-types occurring in w.*

Then there exists w' of the form uv^ω such that the length of u and v is less than $(t+1)^2$ and such that $\tau^w_{k,k'}(0) = \tau^{w'}_{k,k'}(0)$.

Proof. Part 2 of Lemma 2 immediately implies there are u and v such that w and uv^ω have the same type in positions 0. By Lemma 3, we can also assume that u and v are chosen such that $\tau^w_{k,k'}(|u|+s) = \tau^w_{k,k'}(|u|+r|v|+s)$ for $r \geq 0$ and $0 \leq s < |v|$. Assume $|v| \geq (t+1)^2$. For every (k,k')-type τ of a position s with $|u| \leq s < |uv|$ pick a position i_τ such that $|u| \leq i_\tau < |uv|$ and $\tau^w_{k,k'}(i_\tau) = \tau$. Since $|v| \geq (t+1)^2$, we can find two positions l and l' carrying the same type such that $|u| \leq l < l' < |uv|$ and either $i_\tau < l$ or $l' < i_\tau$ for each of the i_τ's. The previous lemma thus applies, so that, by part 2 of Lemma 2, we can replace u by $uv_0v_1 \dots v_{l'-|u|-1}$ and v by $v_{l'-|u|}v_{l'-|u|+1} \dots v_{l-|u|}$. Iterating this process leads to u and v such that uv^ω and w have the same type in position 0 and $|v| < (t+1)^2$. By a similar argument, using part 1 of Lemma 2 instead of part 2, we can reduce the length of u to a value less than $t+1$ (which is less than $(t+1)^2$) while keeping the length of v. ■

We now upper bound the number of types that can occur in a given ω-word:

Lemma 5 *The number of (k,k')-types occurring in any ω-word over Σ_m is at most $2^{3k((2k'+1)(m+1)+1)}$, i. e.,*

$$|\tau^w_{k,k'}(0,\infty)| \leq 2^{3k((2k'+1)(m+1)+1)}$$

for every $w \in \Sigma^\omega_m$.

Proof. The proof is by induction on k. Let w be any ω-word over Σ_m. Let $t_{(k,k')}$ be the number of (k,k')-types occurring in w. For the base case, from (8), it is easy to see that $t_{(0,k')} \leq 2^{(2k'+1)(m+1)}$. Now observe that the sequence $(\tau^w_{k,k'}(0,i-1))_{i\geq 0}$ is an increasing sequence containing at most $t_{(k,k')}$ distinct elements. Similarly, the sequence $(\tau^w_{k,k'}(i+1,\infty))_{i\geq 0}$ is a decreasing sequence containing at most $t_{(k,k')}+1$ distinct elements. Therefore, there are only $2t_{(k,k')}+1$ many distinct pairs of the form $(\tau^w_{k,k'}(0,i-1), \tau^w_{k,k'}(i+1,\infty))$, and thus, using (9), $t_{(k+1,k')} \leq (2t_{(k,k')}+1)2^{(2k'+1)(m+1)}$, where, again, $2^{(2k'+1)(m+1)}$

accounts for the number of distinct $(0, k')$-types. The lemma follows by induction. ∎

Theorem 4 now follows from Lemma 4 together with Lemma 5. ∎

We conclude this subsection with two additional theorems. The first one says that Theorem 4 does not hold when unary-TL is replaced by temporal logic; the second one shows that there are unary-TL formulas whose smallest models are exponentially big. These show the limits of how much one could hope to improve Theorem 4.

Theorem 6 *There is a sequence $(\varphi_n)_{n\geq 0}$ of satisfiable temporal formulas of operator depth $\mathcal{O}(n)$ such that the smallest finite model of φ_n is of size* $\text{tower}(\Omega(n), 2)$.

Proof. The rough idea is as follows. Assume we could produce for a given k a family of t formulas of operator depth n all of which have different unique models of size exactly l. Say these models are $u_0, \ldots, u_{t-1}$. For every permutation π of $\{0, \ldots, t-1\}$, we want to construct a depth $k+1$ formula whose unique model is $u_{\pi(0)}\$u_{\pi(1)}\$\ldots u_{\pi(t-1)}$ where \$ is a symbol that serves as a separator. The models of the new formulas would only be bigger by a linear factor: their length would be $t(l+1)-1$. But we would have exponentially more (formulas and) models: approximately $2^{t\log t}l$ many. We would do the same construction again and would get models of size approximately $2^{t\log t}lt(l+1)$, which is exponential in l, provided l was dominated by t. Iterating this two-stage process would give us the desired non-elementary explosion.

In the following, we will make this idea more formal.

For every $n \geq 0$, we will construct a sequence of formulas $\varphi_n^0, \ldots, \varphi_n^{t_n-1}$ with certain properties, as explained below. To phrase these properties correctly, we need some more notation.

For $r, s \geq 0$, we set $\alpha_r^s = p_{r+1} \vee \ldots \vee p_{r+s-1}$. Given formulas φ and ψ, we write $\varphi[\psi/p_0]$ for the result of replacing every occurrence of p_0 in φ by ψ.

We can now state the properties of the φ_n^i's for a fixed n; the symbols c and d stand for appropriate integer constants.

1. The operator depth of each φ_n^i is at most $5n + c$.

2. There is a number l_n and distinct finite words $u_n^i \in \Sigma_{n+d}^+$ of length $l_n - 2$ such that

$$(v,j) \models \varphi_n^i[\alpha_{n+d}^s/p_0] \quad \text{iff} \quad \begin{cases} v_j = \{p_{n+d}\} \ , \\ v_{j+1} \dots v_{j+l_n-1} = u_n^i \ , \\ v_{j+l_n} = \{p_{n+d}\} \ , \end{cases}$$

for every $s \geq 0$ and every finite word $v \in \Sigma_{n+d+s}^+$ and $j < |v|$.

3. The numbers l_n and t_n satisfy:

$$\begin{array}{rcll} t_0 & > & l_0 \ \geq \ 3 \ , & \\ t_{n+1} & \geq & 2^{t_n} & \text{for } n \geq 0, \\ l_{n+1} & = & t_n(l_n - 1) + 3 & \text{for } n \geq 0. \end{array}$$

Condition 3 implies $t_n \geq l_n \geq 3$ for $n \geq 0$, and thus

$$l_{n+2} = t_{n+1}(l_{n+1} - 1) + 3 \geq 2^{t_n}(l_{n+1} - 1) + 3 \geq 2^{l_n}$$

for $n \geq 0$. Hence, we can set $\varphi_n = \varphi_{2n}^0$.

The construction of the φ_n^i is by induction on n, the base case being an easy exercise. Assume $\varphi_n^0, \dots, \varphi_n^{t_n-1}$ are given. Let S_{t_n} denote the symmetric group on $\{0, \dots, t_n - 1\}$. For every $\pi \in S_{t_n}$, we will construct a formula φ_{n+1}^i so that

$$u_{n+1}^i = \{p_{n+d+1}\}\{p_{n+d}\}u_n^{\pi(0)}\{p_{n+d}\} \dots \{p_{n+d}\}u_n^{\pi(t_n-1)}\{p_{n+d}\} \ .$$

We set

$$\varphi_{n+1}^i = p_{n+d+1} \wedge \bigwedge_{i \leq n+d} \neg p_i \wedge \oplus \psi_\pi \ ,$$

where ψ_π is the conjunction of:

$$(\neg p_0 \wedge \neg p_{n+d+1}) \mathsf{U} (p_{n+d+1} \wedge \bigwedge_{i \leq n+d} \neg p_i)) \ , \tag{15}$$

$$\bigwedge_{i < t_n} \neg p_{n+d+1} \mathsf{U} \varphi_n^i[p_{n+d+1}/p_0] \ , \tag{16}$$

$$\bigwedge_{i < t_n} \neg(\neg p_{n+d+1} \mathsf{U} (\varphi_n^i[p_{n+d+1}/p_0] \wedge \neg p_{n+d+1} \mathsf{U} \varphi_n^i[p_{n+d+1}/p_0])) \ , \tag{17}$$

$$\bigwedge_{i < j < t_n} \neg p_{n+d+1} \mathsf{U} (\varphi_{\pi(i)}[p_{n+d+1}/p_0] \wedge \neg p_{n+d+1} \mathsf{U} \varphi_{\pi(j)}[p_{n+d+1}/p_0]) \ , \tag{18}$$

$$(\neg p_{n+d+1} \wedge (p_{n+d} \to \bigvee_{i < t_n} \varphi_n^i[p_{n+d+1}/p_0])) \mathsf{U} p_{n+d+1} \ . \tag{19}$$

The formulas (15) – (19) formalize in a straightforward way what is needed. For instance, (16) says that every substring u_n^i has to occur (at least) once, (17) says that every substring u_n^i should occur at most once, and (18) requires that the u_n^i's to occur in the right order. ∎

Theorem 7 *There is a sequence $(\varphi_n)_{n>0}$ of satisfiable* unary-TL *sentences in one propositional variable, of size polynomial in n, and of depth $(1, \mathcal{O}(n))$ such that the smallest finite model of φ_n has size at least 2^n.*

Proof. The idea is very simple: φ_n is constructed in such a way that every model of φ_n has

$$\{p_0\}\{p_0\}[0]\{p_0\}\{p_0\}[1]\{p_0\}\{p_0\} \ldots \{p_0\}\{p_0\}[2^n - 1]\{p_0\}\{p_0\}$$

as its prefix where the substring $\{p_0\}\{p_0\}$ serves as separator and [0], [1], [2], ... stand for encodings of the binary representations of 0, 1, 2, ... Here, we say that a string $s_{2n-1} \ldots s_0$ over $\{\emptyset, \{p_0\}\}$ is the binary encoding of a number $i < 2^n$ when $s_{2j+1} = \emptyset$ for $j < n$ and when $s_{2j} = \{p_0\}$ iff the j-th bit of the binary representation of i is 1.

Using polynomial size propositional formulas which define addition by one ("+1"), the construction of polysize formulas φ_n with above property becomes easy.

In fact, an appropriate formula φ_n can be constructed in such a way that in addition its depth in $\oplus$ is at most $2n + 5$ (the number of positions required to represent two numbers $< 2^n$ together with three copies of the marker $\{p_0\}\{p_0\}$) and depth 1 in $\boxplus$: we need to say that for all positions in which one finds the marker, if the following number is $< 2^n - 2$, then the number following this position is what one obtains by adding 1. ∎

4.2 Formula Length

We proceed by proving a different small model property for FO^2, which is phrased in terms of formula size.

Theorem 8 *Every satisfiable FO^2 formula φ has a model of the form uv^ω where the sizes of u and v are bounded by $2^{\mathcal{O}(|\varphi|)}$.*

This small model property is obtained using the appropriate notion of "type" and cut-and-paste arguments corresponding to the ones we have seen in Lemmas 2 and 4. We start with the definition of the right notion of type.

Given an FO^2 formula φ, we define the set of formulas *characteristic* for φ, denoted $\mathrm{cf}(\varphi)$, as follows. When φ is an atomic order formula, then $\mathrm{cf}(\varphi) = \emptyset$. When φ is of the form $P_i x$ or $P_i y$, then $\mathrm{cf}(\varphi) = \{\varphi\}$. When φ is of the form $\neg\psi$ or $\psi_1 \vee \psi_2$, then $\mathrm{cf}(\varphi) = \mathrm{cf}(\psi)$ or $\mathrm{cf}(\varphi) = \mathrm{cf}(\psi_1) \cup \mathrm{cf}(\psi_2)$, respectively. Finally, when φ is of the form $\exists x \varphi^*(x, y)$ or $\exists y \varphi^*(x, y)$ with $\varphi^*(x, y)$ as in (3), then

$$\mathrm{cf}(\varphi) = \{\varphi\} \cup \bigcup_{i<s} \mathrm{cf}(\xi_i) \cup \bigcup_{i<t} \mathrm{cf}(\zeta_i) \cup \{\exists x(\tau \wedge \xi_i) \mid i < s \text{ and } \tau \in \Upsilon\}$$

or

$$\mathrm{cf}(\varphi) = \{\varphi\} \cup \bigcup_{i<s} \mathrm{cf}(\xi_i) \cup \bigcup_{i<t} \mathrm{cf}(\zeta_i) \cup \{\exists y(\tau \wedge \zeta_i) \mid i < t \text{ and } \tau \in \Upsilon\} \ ,$$

respectively.

For an FO^2 formula φ, an ω-word u, and a position i in u, we set $\tau^u_\varphi(i) = \{\psi \in \mathrm{cf}(\varphi) \mid u \models \psi[i]\}$, and we call the $\tau^u_\varphi(i)$'s *φ-types.*

We prove the same cut-and-paste as the one we know from the previous subsection:

Lemma 6 *Let φ be an FO^2-formula. Then Lemma 2 holds when k is replaced by φ.*

Proof. The proof of both parts goes by induction on $|\varphi|$. We sketch a proof of the first part. The only interesting case is when φ is an existential formula. In this case, φ is of the form $\exists x \varphi^*(x, y)$ or $\exists y \varphi^*(x, y)$. Without loss of generality, suppose φ is of the first form, and further suppose $\varphi^*(x, y)$ is as in (3). We can directly apply the induction hypothesis to all elements of $\mathrm{cf}(\xi_l)$ for $l < s$ and $\mathrm{cf}(\zeta_q)$ for $q < t$. And to prove the claim it is thus sufficient to show:

$$w \models \exists y(\tau \wedge \zeta_q)[l] \qquad \text{iff} \qquad w' \models \exists y(\tau \wedge \zeta_q)[l] \tag{20}$$

for $l \leq i$, $q < t$, $\tau \in \Upsilon$, and

$$w \models \exists y(\tau \wedge \zeta_q)[l] \qquad \text{iff} \qquad w' \models \exists y(\tau \wedge \zeta_q)[l - (j - i)] \tag{21}$$

for $j \leq l$, $q < t$, $\tau \in \Upsilon$.

One proves this by a case distinction on the order between i, j, and l on the one hand and the order type τ on the other hand. We only deal with the most complicated case where $l < i$ and $\tau = \neg\mathrm{suc}(x, y) \wedge x < y$, and only show the more difficult implication from left to right.

Assume $w \models \exists y(\tau \wedge \zeta_q)[l]$. Then there exists a position $l' > l + 1$ such that $u \models \zeta_q[l']$. We distinguish three cases.

First, $l' \leq i$. Then $\zeta_q \in \tau_\varphi^w(l')$, and by induction hypothesis, $\zeta_q \in \tau_\varphi^{w'}(l')$, which means $w' \models \zeta_q[l']$, and hence $w' \models \exists y(\tau \wedge \zeta_q)$.

Second, $i < l' \leq j$. Then $\exists y(\mathrm{suc}(x, y) \wedge \zeta_q)$ or $\exists y(\neg\mathrm{suc}(x, y) \wedge x < y \wedge \zeta_q)$ is a member of $\tau_\varphi^w(i)$. Hence, by assumption, one of these formulas is a member of $\tau_\varphi^w(j)$. Consequently, there is a position $l'' > j$ such that $w \models \zeta_q[l'']$. We then have $w' \models \zeta_q[l'' - (j - i)]$ by induction hypothesis, which shows $w' \models \exists y(\tau \wedge \zeta_q)[l]$.

Third, $j < l'$. This is even easier than the previous case. ∎

We can now prove the desired small model property.

Proof of Theorem 8. First, observe that Lemma 4 holds for φ-types instead of (k, k')-types. Second, observe that the total number of φ-types is bounded by $2^{6|\varphi|}$ (as there are 5 oder types). So we obtain a model of φ of the form uv^ω where the size of u and v is bounded by $(2^{6|\varphi|} + 1)^2$, which is in $2^{\mathcal{O}(|\varphi|)}$. ∎

4.3 Unary-TL Without "Next" and "Previously"

For unary-TL[$\diamondplus$] formulas we can prove a really small model property:

Theorem 9 *Every satisfiable unary-TL[$\diamondplus$] formula φ has a model of the form uv^ω where the sizes of u and v are bounded by $|\varphi|$.*

Again, we will use a cut-and-paste argument.

First, observe that unary-TL[$\diamondplus$] formulas starting with a temporal operator have very simple truth tables with respect to a given ω-word:

Remark 3 *Let φ be a unary-TL[$\diamondplus$] formula and $u \in \Sigma_m^\omega$.*

1. *There exists a unique $i \in \{0, 1, 2, \ldots, \omega\}$ such that $u, j \models \diamondplus\varphi$ if and only if $j < i$.*

2. *This position i is the last position in u where φ holds (where, by convention, we say that φ holds at ω if it holds infinitely often).*

The symmetric claim holds for $\diamondminus\varphi$.

We call the distinctive position i from the previous remark the *extremal appearance* of $\diamondplus\varphi$ in u, and denote it by $\mathrm{EA}(\diamondplus\varphi, u)$. Formulas of the form $\diamondminus\varphi$ are dealt with in the same way. Also, given a unary-TL[$\diamondplus$] formula φ we write $\mathrm{tf}(\varphi)$ for the set of subformulas of φ starting with a temporal operator.

The next lemma is going to tell us that positions that are no extremal appearance of a subformula of a given formula φ do not influence whether or not u is a model of φ.

Lemma 7 *Let φ be a unary-TL[$\diamondplus$] formula, $u \in \Sigma_m^\omega$, and $i > 0$ a position that is not any extremal appearance of a formula from $\mathrm{tf}(\varphi)$.*

Then $u \models \varphi$ if and only if $u_0 \dots u_{i-1}u_{i+1} \dots \models \varphi$.

Proof. Write v for $u_0 \dots u_{i-1}u_{i+1} \dots$ The proof goes by induction on the structure of φ. The claim we will show is somewhat stronger: for every j, if $j < i$, then $u, j \models \varphi$ iff $v, j \models \varphi$, and, if $j > i$, then $u, j \models \varphi$ iff $v, j-1 \models \varphi$.

The induction base is trivial as well as the inductive step for negation and conjunction. So we are left with formulas that start with a temporal operator. We consider only those formulas that start with $\diamondplus$. Let φ be of the form $\diamondplus\psi$. We proceed by case distinction on how often ψ is true in u. If ψ is true infinitely often in u, it is, by induction hypothesis, true infinitely often in v, and thus $u, j \models \varphi$ and $v, j \models \varphi$ for $j \geq 0$. If ψ is nowhere true in u or only at position 0, then, by induction hypothesis, ψ is nowhere true in v or only at position 0, hence $u, j \not\models \varphi$ and $v, j \not\models \varphi$ for $j \geq 0$. Otherwise, $0 < \mathrm{EA}(\varphi, u) < \omega$, and $\mathrm{EA}(\varphi, u)$ is the maximal position in u where ψ holds. By assumption, $\mathrm{EA}(\varphi, u) < i$ or $\mathrm{EA}(\varphi, u) > i$. By induction hypothesis, $\mathrm{EA}(\varphi, v) = \mathrm{EA}(\varphi, u)$ or $\mathrm{EA}(\varphi, v) = \mathrm{EA}(\varphi, u) - 1$. This implies $u, j \models \varphi$ and $v, j \models \varphi$ for all $j < i$, and $u, j \not\models \varphi$ and $v, j-1 \not\models \varphi$ for all $j > i$. ∎

The next lemma is of a similar style.

Lemma 8 *Let φ be a unary-TL[$\diamondplus$] formula, $uv^\omega \in \Sigma_m^\omega$, and Φ the set of all formulas ψ such that from $\diamondplus\psi \in \mathrm{tf}(\varphi)$ and $\mathrm{EA}(\diamondplus\psi, uv^\omega) = \omega$. Make the following assumptions.*

1. *For every $\psi \in \mathrm{tf}(\varphi)$, if $\mathrm{EA}(\psi, uv^\omega)$ is finite, then $\mathrm{EA}(\psi, uv^\omega) < |u|$.*

2. *For every* $\psi \in \Phi$, i_ψ *is a position such that* $|u| \leq i_\psi < |uv|$ *and* $(uv^\omega, i_\psi + k|v|) \models \diamondplus\psi$ *for* $k \geq 0$.

Let w *be some subword (that is, subsequence of characters) of* v *that contains the positions* $i_\psi - |u|$ *for* $\psi \in \Phi$.

Then uw^ω *is a model of* φ.

Proof. Write w' for uw^ω. Let $i_0 < i_1 < \ldots < i_{r-1}$ be the positions of v that constitute w. We prove inductively that

$$
\begin{array}{rclr}
(w', j) \models \varphi & \text{iff} & (uv^\omega, j) \models \varphi & \text{for } j < |u|, \\
(w', |u| + k|w| + s) \models \varphi & \text{iff} & (uv^\omega, |u| + k|v| + i_s) \models \varphi & \text{for } k \geq 0,\ s < r.
\end{array}
$$

The induction base is trivial, similarly negation and conjunction. So we are left with when φ starts with a temporal operator. We consider only the case where φ starts with $\diamondplus$, i. e., when φ is of the form $\diamondplus\psi$. Just as in the previous proof, we proceed by a case distinction on how often ψ is true in uv^ω. If ψ is true infinitely often in uv^ω, then $(uv^\omega, j) \models \varphi$ for $j \geq 0$ and $\mathrm{EA}(\varphi, uv^\omega) = \omega$, hence $\varphi \in \Phi$, say $i_t = i_\psi$. By induction hypothesis, $(w', |u| + k|w| + t) \models \psi$ for $k \geq 0$, i. e., $(w', j) \models \varphi$ for $j \geq 0$. If ψ is true nowhere in uv^ω or only at position 0, then, by induction hypothesis, ψ is true nowhere in uv^ω or only at position 0, hence $(uv^\omega, j) \not\models \varphi$ and $(w', j) \not\models \varphi$ for $j \geq 0$. Otherwise, there is a maximal position $j > 0$ in uv^ω where ψ is true and $\mathrm{EA}(uv^\omega, \varphi) = j$. By assumption, $j \leq |u|$, so by induction hypothesis, j is the maximal point in w' where ψ is true. Therefore, the claim holds. ■

We can now prove:

Proof of Theorem 9. If φ is satisfiable, it has an ultimately periodic model uv^ω, and, in addition, u and v can be chosen such that the assumptions from Lemma 8 hold. An application of Lemma 8 then shows that there is w with $|w| \leq |\varphi|$ such that $uw^\omega \models \varphi$. Repeatedly applying Lemma 7, we can now remove letters from u to obtain a word u' of length at most $|\varphi|$ such that still $u'w^\omega \models \varphi$. ■

5 The Complexity of Satisfiability

We will show that the satisfiability problem for FO^2 and $\mathsf{FO}^2[<]$ is **NEXP**-complete. This contrasts with the non-elementary lower bound for satisfiability of first-order logic with three variables over words which follows from [Sto74]. Satisfiability for $\mathsf{unary\text{-}TL}$ remains, as with full TL, **PSPACE**-complete [SC85]. On the other hand, satisfiability of $\mathsf{unary\text{-}TL}[\diamondsuit]$ will be shown to be **NP**-complete.

5.1 First-order Logic with Two Variables

We will prove the following two upper bounds for the complexity of FO^2 satisfiability.

Theorem 10 *Satisfiability for FO^2 (and $\mathsf{FO}^2[<]$) is in* ***NEXP****.*

In fact, satisfiability for an FO^2 formula φ in m unary predicates is decidable in non-deterministic time $2^{\mathcal{O}((\mathrm{qdp}(\varphi)^2+1)m)}$ and $2^{\mathcal{O}(|\varphi|)}$.

Proof. The non-deterministic algorithm determines the satisfiability of an FO^2 formula $\varphi(x)$ over ρ_m as follows. It first guesses u and v of length bounded by $2^{\mathcal{O}(\mathrm{qdp}(\varphi)^2 m)}$ or $2^{\mathcal{O}(|\varphi|)}$, respectively. It then builds up a table that contains for every $i < |uv^{2d+1}|$ and for every subformula $\psi(z)$ of $\varphi(x)$ a bit saying whether $uv^\omega \models \psi[i]$. This is done inductively. The entry for an atomic or composite (see proof of Theorem 1) ψ is easily determined. From Lemma 3, part 3, it follows that in order to determine whether or not an existential formula (see proof of Theorem 1) of the form $\exists y\, \beta(\bar{\chi}(x,y), \bar{\xi}(x), \bar{\zeta}(y))$ holds at a position $i < |uv^{2d+1}|$ it suffices to consider only positions $< |uv^{2d+3}|$ for y. Whether or not a formula $\zeta(y)$ holds at such a position can be determined by a lookup in the table according to (13). The algorithm outputs the entry for position 0 and $\varphi(x)$. ∎

Now to conclude that FO^2 and $\mathsf{FO}^2[<]$ satisfiability are **NEXP**-complete, we observe that they are **NEXP**-hard, which can essentially be pulled out of [Le80, Fü84].

Theorem 11 *Satisfiability for $\mathsf{FO}^2[<]$ (and FO^2) is* ***NEXP****-hard.*

In fact,

1. *satisfiability for $\mathsf{FO}^2[\,]$ formulas (that is, FO^2 formulas that neither use "suc" nor "<") is* ***NEXP****-hard, and*
2. *satisfiability for $\mathsf{FO}^2[\mathrm{suc}]$ formulas (that is, FO^2 formulas that do not use "<") in one unary predicate is* ***NEXP****-hard.*

Proof. We first sketch the proof for part 1. We give a reduction from the problem of determining whether for a given tiling system $T \subseteq \{0, 1, \ldots, c-1\}^4$ with c colors and a given initial row $x \in T^+$ of length n there exists a tiling of a $2^n \times 2^n$ square consistent with T and with x occurring in the lower left corner. (Recall that an element $\langle c_1, c_2, c_3, c_4 \rangle \in T$ is considered a square tile with left edge colored by c_1, right edge colored by c_2, etc. A tiling is consistent if adjacent edges carry the same color.) This problem is known to be **NEXP**-complete, see, e. g., [Fü84]. We can, with a short FO^2 formula, name the adjacent positions in a tiling (and check their consistency) by exploiting the fact that addition has poly-sized propositional formulae. The predicates are used to specify the address coordinates, as well as tile content, of positions in the tiling.

To prove part 2, one compensates the lack of an unbounded vocabulary by using the successor relation as usual. ∎

5.2 Unary-TL without "Next" and "Previously"

That satisfiability for $\mathsf{FO}^2[<]$ is no less difficult than satisfiability for FO^2 (both are **NEXP**-complete) contrasts with what happens to satisfiability when passing from unary-TL to unary-TL[$\diamond\!\!\!\!\!\diamond$]. In [SC85], it was shown that satisfiability for the temporal logic where the only temporal operator is "at present or sometime in the future" is in **NP**. We show that satisfiability for unary-TL[$\diamond\!\!\!\!\!\diamond$] (which now includes the past operator $\diamond$) remains in **NP**, and thus is NP-complete.

Theorem 12 *The satisfiability problem for unary-TL[$\diamond\!\!\!\!\!\diamond$] is **NP**-complete.*

Proof. From [SC85] we know that the problem is **NP**-hard. An appropriate NP decision procedure guesses a "polysize" model uv^ω of φ, which we know exists by Theorem 9, and checks in polynomial time that it is indeed a model. ∎

6 Conclusion

We have shown that the close correspondence between first-order and temporal logic over words persists when looking at first-order formulas with only two variables, and we have presented an easily understood translation of these formulas into temporal formulas. Our translation is essentially optimal: the formulas incur at most an exponential blow-up in size and we have proved that this is necessary in the worst case.

The satisfiability problem for unary-TL is known to remain, as with full TL, **PSPACE**-complete, but we have shown that FO^2 satisfiability is drastically simpler than FO^3 satisfiability: the former is **NEXP**-complete, while the latter is known to require non-elementary complexity. Moreover, our **NEXP** upper bound for FO^2 satisfiability, and the corresponding small model properties for FO^2 and unary-TL, have the advantage of being only in terms of quantifier/operator depth and the number of propositions in the vocabulary, rather than the size of the entire formula, a fact that may be of potential use when dealing with large but shallow formulas.

Only recently ([TW96]) it has been shown that given a finite automaton or ω-automaton it is decidable whether or not the language recognized by this automaton is definable in unary-TL or unary-TL[$\diamond$]. This means, in particular, that it is decidable whether or not a given TL formula is equivalent to a unary-TL or unary-TL[$\diamond$] formula. By our translation, this also means that it is decidable whether or not a given FO formula is equivalent to an FO^2 or $\mathsf{FO}^2[<]$ formula.

Some remaining questions: (1) Is the FO^2 quantifier alternation hierarchy strict? This question can also be phrased in terms of operator alternation in unary-TL. (2) Does satisfiability remain **NEXP**-hard for $\mathsf{FO}^2[<]$ formulas (without successor) over a bounded number of predicates? (3) Can the upper bound of the small model property for FO^2 be improved to $2^{\mathcal{O}(\mathrm{qdp}(\varphi)+m)}$? This would make (the proof of) Theorem 8 obsolete.

References

[EW96] K. Etessami and Th. Wilke. An Until hierarchy for temporal logic. In *11th Annual IEEE Symposium on Logic in Computer Science*, New Brunswick, New Jersey, pages 108–117, 1996.

[Fü84] M. Fürer. The computational complexity of the unconstrained domino problem (with implications for logical decision problems). In *Logic and Machines: Decision Problems and Complexity*, pages 312–319. Volume 171 in Lect. Notes in Comput. Sci., Springer, 1984.

[GHR94] D. M. Gabbay, I. Hodkinson, and M. Reynolds. *Temporal Logic*, volume 1. Clarendon Press, Oxford, 1994.

[GPSS80] D. M. Gabbay, A. Pnueli, S. Shelah, and J. Stavi. On the temporal analysis of fairness. In *Conference Record of the 7th Annual ACM Symposium on Principles of Programming Languages*, Las Vegas, Nev., pages 163–173, 1980.

[GKV97] E. Grädel, Ph. G. Kolaitis, and M. Y. Vardi. On the Decision Problem for Two-Variable First-Order Logic. To appear in *Bulletin of the Assoc. for Symbolic Logic*.

[IK89] N. Immerman and D. Kozen. Definability with bounded number of bound variables. *Information and Computation*, 83(2):121–139, 1989.

[Kam68] J. A. W. Kamp. *Tense Logic and the Theory of Linear Order*. PhD thesis, University of California, Los Angeles, 1968.

[Le80] H. R. Lewis. Complexity Results for Classes of Quantificational Formulas. *J. Comput. System Sci.*, 21: 317–353, 1980.

[MP92] Z. Manna and A. Pnueli. *The Temporal Logic of Reactive and Concurrent Systems*, Springer-Verlag, Berlin/New York, 1992.

[Mor74] M. Mortimer. On languages with two variables. *Z. Math. Logik Grundlag. Math.*, 21:135-140, 1975.

[Pnu77] A. Pnueli. The temporal logic of programs. In *Proceedings of the 18th Annual Symposium on Foundations of Computer Science*, Providence, Rhode Island, pages 46–57, 1977.

[SC85] A. P. Sistla and E. M. Clarke. The complexity of propositional linear temporal logics. *J. Assoc. Comput. Mach.*, 32(3):733–749, 1985.

[Sto74] L. J. Stockmeyer. *The Complexity of Decision Problems in Automata Theory and Logic*. PhD thesis, Department of Electrical Engineering, MIT, 1974.

[TW96] D. Thérien and Th. Wilke. "'Over Words, Two Variables Are as Powerful as One Quantifier Alternation: $\mathrm{FO}^2 = \Sigma_2 \cap \Pi_2$."' In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing* (STOC), Dallas, Texas, 41–47, 1998.

[VW86] M. Y. Vardi and P. Wolper. An Automata-Theoretic Approach to Automatic Program Verification. In *Proceedings of the First Annual IEEE Symposium on Logic in Computer Science*, Cambridge, Mass., pages 322-331, 1986.