Optimal Complexity Bounds for Positive LTL Games

Jerzy Marcinkowski* and Tomasz Truderung**

Institute of Computer Science, Wrocław University
jma@ii.uni.wroc.pl, tt@ii.uni.wroc.pl

Abstract. We prove two tight bounds on complexity of deciding graph games with winning conditions defined by formulas from fragments of LTL.

Our first result is that deciding $LTL_+(\Diamond, \wedge, \vee)$ games is in PSPACE. This is a tight bound: the problem is known to be PSPACE-hard even for the much weaker logic $LTL_+(\Diamond, \wedge)$. We use a method based on a notion of, as we call it, persistent strategy: we prove that in games with positive winning condition the opponent has a winning strategy if and only if he has a persistent winning strategy.

The best upper bound one can prove for our problem with the Büchi automata technique, is EXPSPACE. This means that we identify a natural fragment of LTL for which the algorithm resulting from the Büchi automata tool is one exponent worse than optimal.

As our second result we show that the problem is EXPSPACE-hard if the winning condition is from the logic $LTL_+(\Diamond, \circ, \wedge, \vee)$. This solves an open problem from [AT01], where the authors use the Büchi automata technique to show an EXPSPACE algorithm deciding more general $LTL(\Diamond, \circ, \wedge, \vee)$ games, but do not prove optimality of this upper bound

1 Introduction

LTL (linear temporal logic) is one of possible specification languages for correctness conditions in reactive systems verification [MP91]. Two sorts of decision problems arise in this context. One of them is **model checking**. We ask here, for a given transition graph $\mathcal{G}$ of a system, and for a formula φ of LTL, whether φ is valid on all possible computation paths in $\mathcal{G}$. This question is natural when a closed system is verified, by which we mean one whose future behavior only depends on its current state but not on any kind of environment. Model checking for LTL conditions is known to be PSPACE-complete [SC85] (combined complexity). Although, if $\Diamond$ and $\Box$ are the only modalities allowed in the formula then model-checking is NP-complete [SC85]. Other fragments of LTL with easy model-checking problem (in NP or even in P) are identified in [DS98].

* Partially supported by Polish KBN grant 2 PO3A 01818.
** Partially supported by Polish KBN grant 8T11C 04319.

In this paper we are interested in the second kind of decision problems in this area, which is **deciding a game** with condition φ. The computation path here is a result of an infinite game played by two players $\mathcal{S}$ (as *System*) and $\mathcal{E}$ (as *Environment*) on some game graph $\mathcal{G}$. Each vertex of $\mathcal{G}$ is either existential, when $\mathcal{S}$ decides on the next move, or universal, when $\mathcal{E}$ is the one who moves. The goal of $\mathcal{S}$ is to make the formula φ valid on the computation path. This paradigm is being considered in the context of automated synthesis. The future behavior of the system depends here not only on its current state but also on the inputs supplied by some unpredictable environment. It is known that deciding which of the players has a winning strategy in such a graph game is doubly exponential for general LTL formula φ [PR89].

1.1 Previous Work

Positive results. A classical technique for deciding an LTL game is to transform the winning condition φ into a deterministic ω-automaton A_φ, so called generator of φ, which accepts an infinite path if and only if φ is true on this path. Then take $\mathcal{B} = \mathcal{G} \times A_\varphi$ as a new game (where $\mathcal{G}$ is the game graph under consideration). The type of the game $\mathcal{B}$ (Büchi, Rabin, etc.) is the same as the type of the generator A_φ. The winning condition on $\mathcal{B}$ is defined in such a way that the same player who had a winning strategy in the φ game on $\mathcal{G}$ has a winning strategy in the game on $\mathcal{B}$.

In [AT01] Alur and La Torre consider fragments of LTL which have deterministic generators being Büchi automata, and thus the resulting game is a Büchi game and the winning player has a memoryless strategy. It is easy to decide such a game: this can be done in a quadratic time with respect to the size (number of vertices) of the game graph [Tho95]. Alur and La Torre improve on this: they notice that one can decide a Büchi game in SPACE($d \log n$), where n is the size of the game graph and d is another parameter called the longest distance of the game graph. They carefully construct Büchi generators for different fragments of LTL, trying to keep the longest distance as small as possible. In this way they show that deciding $LTL(\Diamond, \wedge)$ games is in PSPACE and that the same problem for $LTL(\Diamond, \circ, \wedge, \vee)$ (and thus also for $LTL(\Diamond, \wedge, \vee)$) is in EXPSPACE.

Lower bounds. It is known since [PR89] that the doubly exponential algorithm deciding general LTL games is optimal. In their study of the complexity of games with conditions from fragments of LTL [AT01] Alur and La Torre show the PSPACE lower bound for $LTL_+(\Diamond, \wedge)$ (this proof is very easy) and the EXPTIME lower bound for $LTL(\Diamond, \circ, \wedge)$, and thus for $LTL(\Diamond, \circ, \wedge, \vee)$.

1.2 Our Contribution

Lower bound for $LTL_+(\Diamond, \circ, \wedge, \vee)$. In Section 5 we solve an open problem from [AT01] proving:

Theorem 1. *Deciding games with the winning condition in $LTL_+(\Diamond, \circ, \wedge, \vee)$ is EXPSPACE-hard.*

This is an optimal result, and a surprisingly strong one: it turns out that the problem for the positive part $LTL_+(\Diamond, \circ, \wedge, \vee)$ is as hard as for its boolean closure $LTL(\Diamond, \circ, \wedge, \vee)$.

In our proof we use the fact that EXPSPACE can be viewed as a variant of alternating EXPTIME. The game graph is defined in such a way that in the first stage of a play the opponents, by turn, construct (or, as we say, *declare*) a sequence which is intended to be a computation of an alternating machine. Then, in the second stage, some way must be provided to detect all possible sorts of cheating against the legality of this computation. And this is where our main tool comes, which we call *the objection graph.* It appears that a formula of $LTL(\Diamond, \circ, \wedge, \vee)$ expressing the property *there are two equal patterns of length n on the path, both beginning with the the state p* requires the size exponential in n. But as we show, if we have two players declaring a sequence, and each of them can "raise an objection" by moving the play into the objection graph, then a small (polynomial-length) formula of $LTL(\Diamond, \circ, \wedge, \vee)$ is enough to detect equality of patterns of length n, as well as all the legality violations we need to detect. Since we wanted to keep the formula positive, we could only grant to $\mathcal{S}$ the ability of raising objections. This means that his cheats in the first stage could remain undetected. This is why we need to construct the first stage with some care.

Positive result for $LTL_+(\Diamond, \vee, \wedge)$. In Section 4 we prove:

Theorem 2. *Deciding games with the winning condition in $LTL_+(\Diamond, \vee, \wedge)$ is in PSPACE.*

Again, it follows from [AT01] that this result is optimal. $LTL_+(\Diamond, \vee, \wedge)$ may appear to be quite a simple logic but still it requires huge generators. Indeed, while studying $LTL(\Diamond, \vee, \wedge)$ the authors of [AT01] show that a deterministic generator for the formula $\Diamond((p_1 \vee \Diamond q_1) \wedge (p_2 \vee \Diamond q_2) \wedge \ldots (p_k \vee \Diamond q_k))$ of the logic $LTL_+(\Diamond, \vee, \wedge)$, requires exponential longest distance and doubly exponential size. This means that with their Büchi automata methodology no upper bound better than EXPSPACE can be achieved for $LTL_+(\Diamond, \vee, \wedge)$ games. And this, as we prove, is one exponent worse than optimal. The core of our technique is the notion of a *persistent strategy*[1] (see Definition 1). In Section 3 we prove that if $\mathcal{E}$ has a winning strategy in any positive game then he also has a persistent winning strategy. And deciding an $LTL_+(\Diamond, \vee, \wedge)$ game if $\mathcal{E}$ uses a persistent strategy is in PSPACE, as we show, in Section 4.

2 Preliminaries

Linear Temporal Logic. Let P be a given finite set of *atomic propositions.* *Linear temporal logic* (LTL) formulas are built according to the grammar:

$$\varphi ::= s \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \circ\varphi \mid \Diamond\varphi \mid \Box\varphi \mid \varphi \,\mathsf{U}\, \varphi,$$

[1] The notion of persistent strategy is a very natural one. We believe it can have other applications. That is why we would not be surprised to learn that it has been studied before. However, we are not currently aware of any reference to such a study.

where s is a *state predicate*, that is a boolean combination of atomic propositions. Temporal operators $\bigcirc, \Diamond, \Box$, U are usually read as *next*, *eventually*, *always*, and *until* respectively. LTL formulas are interpreted in the standard way on infinite sequences over the alphabet $\Sigma = 2^P$.

Fragments of LTL. We denote by $LTL_+(op_1, \ldots, op_k)$ the set of LTL formulas built from state predicates using only boolean and temporal connectives $op_1, \ldots, op_k$. Furthermore, following [AT01], we denote by $LTL(op_1, \ldots, op_k)$ the set of formulas obtained as boolean combinations of $LTL_+(op_1, \ldots, op_k)$.

Game Graphs. A two-player φ game on $\mathcal{G}$ is given by an LTL formula φ, called a *winning condition*[2], and a *game graph* $\mathcal{G} = (V, V_\forall, V_\exists, E, v_0, \delta)$ with the set of vertices V partitioned into $V_\forall$ and $V_\exists$, the set of edges $E \subseteq V \times V$, the initial vertex $v_0 \in V$, and a function $\delta : V \to 2^P$ which assigns to each vertex a set of atomic propositions. We say that p is true in v if $p \in \delta(v)$. Elements of $V_\forall$ are called *universal vertices*, and elements of $V_\exists$ are called *existential vertices*. To denote elements of V we will use letters $u, v, w, \ldots$

A *finite play* is a sequence $u_0 \ldots u_k \in V^*$ such that u_0 is the initial vertex, and $\langle u_{i-1}, u_i \rangle \in E$, for all $i \in \{1, \ldots, k\}$. Similarly, an *infinite play* is an infinite sequence $u_0 u_1 \ldots$ of elements from V such that u_0 is the initial vertex, and $\langle u_{i-1}, u_i \rangle \in E$, for all $i \geq 1$. To denote (finite or infinite) plays we will use letters $\mathbf{u}, \mathbf{v}, \mathbf{w}, \ldots$ During a game, two players $\mathcal{S}$ (the System) and $\mathcal{E}$ (the Environment) construct a sequence $\mathbf{v_0}, \mathbf{v_1}, \mathbf{v_2}, \ldots$ of finite plays. They begin with $\mathbf{v_0} = v_0$. If $\mathbf{v_i} = \mathbf{w}w$ for some $\mathbf{w}w$ then $\mathbf{v_{i+1}} = \mathbf{w}ww'$, where w' is selected by $\mathcal{S}$ if w is existential, and by $\mathcal{E}$ if w is universal. Let $\mathbf{v}$ be the infinite play which is the limit of $\mathbf{v_0}, \mathbf{v_1}, \mathbf{v_2}, \ldots$ Then $\mathcal{S}$ wins if $\mathbf{v} \models \varphi$. A *strategy* and a *winning strategy* for $\mathcal{S}$ (or $\mathcal{E}$) is defined in the standard way.

The problem of deciding $LTL(op_1, \ldots, op_k)$ (or $LTL_+(op_1, \ldots, op_k)$) *games* is a problem of deciding whether $\mathcal{S}$ has a winning strategy for a given game graph, and a winning condition given as an $LTL(op_1, \ldots, op_k)$ (or $LTL_+(op_1, \ldots, op_k)$) formula.

3 Positive Games and Persistent Strategies

Definition 1. *The strategy of the player $\mathcal{P}$ is* persistent *if for each play $v_1 v_2 \ldots v_k$ played by $\mathcal{P}$ according to this strategy, if $v_i = v_j$, for some $1 \leq i, j < k$, and v_i is a vertex where $\mathcal{P}$ is to move, then $v_{i+1} = v_{j+1}$.*

In other words, a strategy of the player $\mathcal{P}$ is persistent if, each time $\mathcal{P}$ decides on a move in some vertex v, he repeats the decision he made when v was visited for the first time.

One of the most well-studied kind of strategies are memoryless strategies: this means that the way the player behaves only depends on the vertex of the graph, not on the history of the game. Being persistent is a weaker property than being memoryless:

[2] In general many types of winning conditions are considered (see [Tho90]).

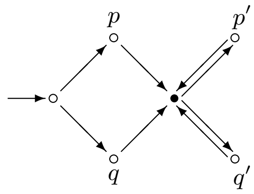

Fig. 1.

Example. Let $\mathcal{G}$ be a game graph with $V = \{u, u_p, u_q, u_{p'}, u_{q'}, v\}$ where u is the initial vertex and all vertices except v are existential (Fig. 1). The edges in E are: $\langle u, u_p\rangle, \langle u, u_q\rangle, \langle u_p, v\rangle, \langle u_q, v\rangle, \langle v, u_{p'}\rangle, \langle v, u_{q'}\rangle, \langle u_{p'}, v\rangle, \langle u_{q'}, v\rangle$. The variables p, q, p', q' are true in vertices $u_p, u_q, u_{p'}$ and $u_{q'}$ respectively. Let φ be the formula $\Diamond((p \wedge \Diamond p') \vee (q \wedge \Diamond q'))$. Then $\mathcal{E}$ does not have a memoryless winning strategy in the φ game on $\mathcal{G}$ but he does have a persistent winning strategy. □

As we are soon going to see the existence of a persistent winning strategy in the example above is not a coincidence.

Notations. For two plays $\mathbf{w}$ and $\mathbf{v}$ we will use the notation $\mathbf{w} \leq \mathbf{v}$ to say that $\mathbf{w}$ is a prefix of $\mathbf{v}$. Let $\mathbf{v}, \mathbf{w}$ be two plays, finite or not. Then by $\mathbf{w} \sqsubseteq \mathbf{v}$ we mean the expression "$\mathbf{w}$ is a subsequence of $\mathbf{v}$" (where *abc* is a subsequence of *adbdc*).

Definition 2. *We call a game* positive*, if for each two infinite plays* $\mathbf{w}$ *and* $\mathbf{v}$*, if* $\mathcal{S}$ *wins the play* $\mathbf{w}$ *and* $\mathbf{w} \sqsubseteq \mathbf{v}$ *then* $\mathcal{S}$ *wins also the play* $\mathbf{v}$*.*

It will be convenient in this section to see a strategy for $\mathcal{E}$ as a tree of all possible finite plays played according to this strategy. The following definition is consistent with the standard way of defining strategy:

Definition 3. *A* strategy for $\mathcal{E}$ *is a set* T *of finite plays such that:*

(i) $v_0 \in T$*, where* v_0 *is (the word consisting of) the initial vertex of* $\mathcal{G}$*;*
(ii) *if* $\mathbf{w} \in T$ *and* $\mathbf{v} \leq \mathbf{w}$ *is a nonempty prefix of* $\mathbf{w}$ *then* $\mathbf{v} \in T$*;*
(iii) *if* $\mathbf{w}w \in T$*, where* w *is an existential vertex of* $\mathcal{G}$*, then* $\mathbf{w}wv \in T$ *for each vertex* v *such that* $(w, v) \in E$*;*
(iv) *if* $\mathbf{w}w \in T$*, where* w *is a universal vertex of* $\mathcal{G}$*, then* $\mathbf{w}wv \in T$ *for exactly one vertex* v *such that* $(w, v) \in E$*.*

A strategy for $\mathcal{E}$, as defined above, has a natural structure of an infinite tree, and is *winning* if each infinite path of this tree is a play won by $\mathcal{E}$.

Lemma 1. *Let* T *be a winning strategy for* $\mathcal{E}$ *in some positive game. Let* T' *be a strategy for* $\mathcal{E}$ *with the property that for each* $\mathbf{w} \in T'$ *there exists* $\mathbf{v} \in T$ *such that* $\mathbf{w} \sqsubseteq \mathbf{v}$*. Then* T' *is also a winning strategy for* $\mathcal{E}$*.* □

The main result of this section is:

Theorem 3. *If* $\mathcal{E}$ *has a winning strategy in a some positive game on some graph* $\mathcal{G}$ *with* n *vertices, then he has a winning strategy which is persistent.*

The proof of the theorem will take the rest of this section. The following notation will be useful:

Definition 4. *Let T be a strategy for $\mathcal{E}$ and let v be a universal vertex. Then by T^v we denote the set of those $\mathbf{v} \in T$ which are of the form $\mathbf{v}'v$ for some $\mathbf{v}'$. Similarly, by T^{vw} we denote the set of those $\mathbf{v} \in T$ which are of the form $\mathbf{v}'vw$ for some $\mathbf{v}'$. By $T^v_{\mathbf{w}}$ (and $T^{vw}_{\mathbf{w}}$) we will denote the set of those $\mathbf{u} \in T^v$ ($\mathbf{u} \in T^{vw}$) for which $\mathbf{u} \geq \mathbf{w}$ holds.*

We will need a local version of the notion of a persistent strategy:

Definition 5. *Let T be a strategy for $\mathcal{E}$, and $\mathbf{v} \in T^v$ for some universal v. Let w be the (unique) vertex of V such that $\mathbf{v}w \in T$. Then T is $\mathbf{v}$-persistent if for each play $\mathbf{w} \in T^v_{\mathbf{v}}$ we have $\mathbf{w}w \in T$.*

The meaning of the definition is that T is $\mathbf{v}$-persistent for some $\mathbf{v} \in T^v$ if the decision about the way $\mathcal{E}$ plays in vertex v made at the moment after the play $\mathbf{v}$, will not be changed in the future. It is easy to see that a strategy T is persistent if and only if it is $\mathbf{v}$-persistent for each $\mathbf{v} \in T$ such that $\mathbf{v} \in T^v$ for some universal v.

To end the proof of Theorem 3, it will be enough to prove:

Lemma 2. *Let T be a winning strategy for $\mathcal{E}$. For each universal v and each $\mathbf{v} \in T^v$, there exists a winning strategy $\mathcal{T}(T, \mathbf{v})$ for $\mathcal{E}$ such that:*

1. *$\mathcal{T}(T, \mathbf{v})$ is $\mathbf{v}$-persistent;*
2. *if $\mathbf{v} \not\leq \mathbf{w}$ then $\mathbf{w} \in \mathcal{T}(T, \mathbf{v})$ if and only if $\mathbf{w} \in T$;*
3. *if $\mathbf{v} \leq \mathbf{u}$ and $\mathbf{u}uu' \in \mathcal{T}(T, \mathbf{v})$ for some universal $u \neq v$, then there exists $\mathbf{w}$ such that $\mathbf{v} \leq \mathbf{w}$ and $\mathbf{w}uu' \in T$.*

With the last lemma a persistent winning strategy for $\mathcal{E}$ can be constructed from any winning strategy for $\mathcal{E}$: going from the root of T down each path, replace T with $\mathcal{T}(T, \mathbf{v})$ each time a play $\mathbf{v} \in T^v$ is reached, where v is universal and v does not occur in $\mathbf{v}$ earlier than as its last symbol. This procedure converges to some winning strategy for $\mathcal{E}$, because on each path such a replacement will be done at most n times. By item 2 of the lemma two such replacements performed in $\leq$–incomparable points do not interfere. By item 3, if $\mathbf{u} \leq \mathbf{w}$ then $\mathcal{T}((\mathcal{T}(T, \mathbf{u})), \mathbf{w})$ remains $\mathbf{u}$-persistent, so the later replacements do not destroy the effect of the earlier.

Proof of Lemma 2. Let $\leq_v$ be the prefix ordering on T^v (so $\leq_v$ coincides on $T^v \times T^v$ with the relation $\leq$, the prefix ordering on the set of all finite plays).

There are 2 cases.

Case 1. There is a play $\mathbf{w} \in T^v_{\mathbf{v}}$ which is $\leq_v$ maximal.

It is easy to see that in this case we can put $\mathbf{s} \in \mathcal{T}(T, \mathbf{v})$ for $\mathbf{v} \not\leq \mathbf{s}$ such that $\mathbf{s} \in T$, and $\mathbf{vs} \in \mathcal{T}(T, \mathbf{v})$ for each $\mathbf{ws} \in T$. By Lemma 1, the obtained strategy is a winning strategy for $\mathcal{E}$.

Case 2. There is no such $\leq_v$ maximal play in $T^v_{\mathbf{v}}$.

This case is more complicated. We will need:

Definition 6. *In the situation of Case 2, let* $\mathbf{u} \in T^v_{\mathbf{v}}$ *be such that* $\mathbf{u}v' \in T$*. We will say that* $\mathbf{u}$ *is* v'*-dense if, for each* $\mathbf{w} \in T^v_{\mathbf{u}}$*, the set* $T^{vv'}_{\mathbf{w}}$ *is non-empty.*

It turns out that:

Lemma 3. *There exists* $\mathbf{u}v' \in T$*, where* $\mathbf{u} \in T^v_{\mathbf{v}}$*, such that* $\mathbf{u}$ *is* v'*-dense.*

Proof. Suppose there is no such $\mathbf{u}$. We define by induction a sequence $\mathbf{w_1} \leq \mathbf{w_2} \leq \mathbf{w_3} \ldots$ of plays, and a sequence $w_1, w_2, \ldots$ of vertices. Let $\mathbf{w_1} = \mathbf{v}$. Suppose that $\mathbf{w}_i \in T^v_{\mathbf{v}}$ is already defined, and let w_i be such that $\mathbf{w}_i w_i \in T$. We know that $\mathbf{w}_i$ is not w_i dense. This means that there exists $\mathbf{u} \in T^v_{\mathbf{w}_i}$ such that $T^{vw_i}_{\mathbf{u}}$ is empty. Define $\mathbf{w}_{i+1} = \mathbf{u}$. Notice that if $i > j$ then $T^{vw_j}_{\mathbf{w_i}}$ is empty (because $T^{vw_j}_{\mathbf{w}_i} \subseteq T^{vw_j}_{\mathbf{w}_{j+1}}$). Thus $w_i \neq w_j$. We get a contradiction since there are only finitely many elements of V. □

Once we have $\mathbf{w}$ which is w-dense for some w we are ready to construct $\mathcal{T}(T, \mathbf{v})$. Consider a play $\mathbf{u} \geq \mathbf{w}$. Then: $\mathbf{u} = \mathbf{w}\mathbf{v_1}v\mathbf{v_2}v \ldots v\mathbf{v_m}$ where v does not occur in $\mathbf{v_1}\mathbf{v_2} \ldots \mathbf{v_m}$. Let $\alpha(\mathbf{s})$ be $\mathbf{s}$ if the first symbol of $\mathbf{s}$ is w and the empty word otherwise. Define $\beta(\mathbf{u})$ as $\mathbf{v}\alpha(\mathbf{v_1}v)\alpha(\mathbf{v_2}v) \ldots \alpha(\mathbf{v_m})$.

Now: $\mathcal{T}_0(T, \mathbf{v}) = \{\mathbf{u} : \mathbf{v} \not\leq \mathbf{u} \wedge \mathbf{u} \in T\} \cup \{\beta(\mathbf{u}) : \mathbf{u} \geq \mathbf{w} \wedge \mathbf{u} \in T\}$.

$\mathcal{T}_0(T, \mathbf{v})$ is not yet a strategy: condition (iv) of Definition 3 may not hold in this tree: it is possible that some plays ending with a universal vertex different than v will have more than one direct successor there. One can prune the tree $\mathcal{T}_0(T, \mathbf{v})$ in any way, so the result satisfies Definition 3 (iv), and call the result $\mathcal{T}(T, \mathbf{v})$. With the use of Lemma 1 one can now verify that $\mathcal{T}(T, \mathbf{v})$ is a strategy as required by Definition 3 and by Lemma 2. This ends the proof of Lemma 2. □

4 Proof of Theorem 2

Notations. Let $n = |V|$ where V is the set of vertices of the game graph $\mathcal{G}$. By φ we always mean a formula of $LTL_+(\Diamond, \vee, \wedge)$ in this section.

Since φ is positive the following lemma holds:

Lemma 4. *For a given game graph* $\mathcal{G}$ *and a formula* φ *there exists* $\rho(\mathbf{s})$ *such that:*

1. $\rho(\mathbf{s})$ *is a positive boolean combination of expressions of the form* $\mathbf{w} \sqsubseteq \mathbf{s}$*, where* $\mathbf{s}$ *is the variable which is free in* ρ*, and each* $\mathbf{w}$ *is some fixed word of length not greater than* l*, where* l *is the* $\Diamond$*-depth of* φ*;*
2. ρ *and* φ *are equivalent in the sense that for each infinite play* $\mathbf{v}$ *it holds that* $\rho(\mathbf{v})$ *if and only if* $\mathbf{v} \models \varphi$*.*

Proof. Induction on l. □

By the last lemma, if $\mathbf{v}$ is some infinite play won by $\mathcal{S}$ then there exists a finite prefix $\mathbf{v'}$ of $\mathbf{v}$ such that, for every infinite play $\mathbf{w}$, if $\mathbf{v'} \leq \mathbf{w}$ then $\mathbf{w}$ is won by $\mathcal{S}$. In such a case we say that $\mathcal{S}$ *secures his win after the play* $\mathbf{v'}$. One can

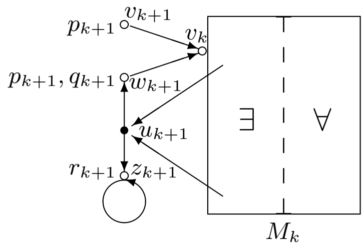

Fig. 2. Graph M_{k+1}

prove that if $\mathcal{S}$ has a winning strategy in the φ game on $\mathcal{G}$ then he can secure his win after a number of steps which is exponential with respect to the combined size of the instance. Our first conjecture was that the win of $\mathcal{S}$ can be secured in such a case already after polynomial number of steps. If true, this would give a straightforward way of proving Theorem 2. It would be enough to perform the mini-max search on the tree of all plays of polynomial depth, a procedure which is clearly in PSPACE. Our conjecture is, however, false:

Theorem 4. *There exist a formula φ and a graph $\mathcal{G}$ such that $\mathcal{S}$ has a winning strategy, but he is not always able to secure his win after a polynomial number of steps.*

Proof. Let M_0 be a game graph consisting of only one existential vertex v_0 where p_0 is true, and of one edge $E(v_0, v_0)$. Let φ_0 be the formula $\Diamond p_0$. We define φ_{k+1} as $\Diamond(p_{k+1} \wedge \Diamond\varphi_k \wedge (q_{k+1} \vee \Diamond r_{k+1}))$. Graph M_{k+1} (Fig. 2) consists of all the vertices and edges of graph M_k, of new existential vertices v_{k+1},w_{k+1} and z_{k+1} and a new universal vertex u_{k+1}. There are also new edges: from v_{k+1} and from w_{k+1} to the initial vertex of M_k, from each existential vertex of M_k to u_{k+1}, from u_{k+1} both to w_{k+1} and to z_{k+1}, and a loop from z_{k+1} to itself. The initial vertex of the new graph is v_{k+1}. The variables which are true in the vertices of M_k remain true in the same vertices in M_{k+1}. For the new vertices: p_{k+1} is true in v_{k+1} and in w_{k+1}, r_{k+1} is true in z_{k+1} and q_{k+1} is true in w_{k+1}.

Now, one can prove by induction on k that, for every k, $\mathcal{S}$ has a winning strategy in the φ_k game on M_k. Assume the claim is true for some k and consider the φ_{k+1} game on M_{k+1}. $\mathcal{S}$ moves from v_{k+1} to v_k and then uses his winning strategy in the φ_k game on M_k. Once he secures his win in the φ_k game on M_k he uses one of the new edges to leave M_k, and goes to u_{k+1}. Now $\mathcal{E}$ is to move. If he decides to go to z_{k+1} then the formula $\Diamond(p_{k+1} \wedge \Diamond\varphi_k \wedge \Diamond r_{k+1})$, which implies φ_{k+1}, is true on the constructed play. If $\mathcal{E}$ prefers to move to w_{k+1} instead of z_{k+1} then $\mathcal{S}$ enters M_k and once again uses his winning strategy in the φ_k game on M_k. Once he secures his win in this smaller game again, the formula $\Diamond(p_{k+1} \wedge \Diamond\varphi_k \wedge q_{k+1})$, which implies φ_{k+1}, holds true on the resulting game.

We also use induction on k in order to show that $\mathcal{E}$ can survive 2^k of steps before the win of $\mathcal{S}$ in the φ_k game on M_k is secured. Assume the claim is true for some k, and consider the situation for $k+1$. If $\mathcal{S}$ makes the step from the M_k part to u_{k+1} before he secures the win in the φ_k game there, then $\mathcal{E}$ can move to z_{k+1} and win. So $\mathcal{S}$ cannot enter u_{k+1} before 2^n moves are made. If now $\mathcal{E}$ moves from u_{k+1} to w_{k+1} then the only way to secure win for $\mathcal{S}$ is to move to v_k and win the φ_k game on M_k again, which again takes at least 2^n moves. □

But it turns out that in spite of Theorem 4 we are still able to find a way of restricting the search only to game trees of polynomial depth. Notice that the game under consideration is positive. So thanks to Theorem 3 we can assume that $\mathcal{E}$ is using a persistent strategy. To end the proof of Theorem 2 it is enough to prove:

Lemma 5. *If $\mathcal{S}$ has a winning strategy in a φ game on $\mathcal{G}$ and he plays against an opponent who uses a persistent strategy, then $\mathcal{S}$ can secure his win after a polynomial number of steps.*

4.1 Proof of Lemma 5

In this subsection we assume that $\mathcal{E}$ uses a persistent strategy.

Lemma 6. *Suppose $\mathbf{v} = v_1 v_2 \dots v_m$ is a play such that v_m is an existential vertex and $\mathcal{S}$ has a winning strategy after $\mathbf{v}$ is played. Then there exists a play $\mathbf{v} u_1 u_2 \dots u_k$, with k polynomial, such that:*

1. *if u_i is universal, for some $1 \le i \le k-1$, then $u_i = v_j$, for some $1 \le j \le m-1$ and $u_{i+1} = v_{j+1}$;*
2. *either the win of $\mathcal{S}$ is already secured after the play $\mathbf{v} u_1 u_2 \dots u_k$ or u_k is a universal vertex which does not occur in $\mathbf{v} u_1 u_2 \dots u_{k-1}$, and $\mathcal{S}$ has a winning strategy after the play $\mathbf{v} u_1 u_2 \dots u_k$.*

Let us first show that Lemma 5 follows from 6. Notice that $\mathcal{E}$ has no opportunity between v_m and u_k in the lemma to make any decisions about the way the moves are being made. They are either made by $\mathcal{S}$, or are already determined, since the strategy of $\mathcal{E}$ is persistent. So, once the play $\mathbf{v}$ has been played, it is up to $\mathcal{S}$ if $\mathbf{v} u_1 u_2 \dots u_k$ is played.

Notice also, that if $\mathbf{v} = v_1 v_2 \dots v_m$ is a play such that v_m is a universal vertex and $\mathcal{S}$ has a winning strategy after $\mathbf{v}$ is played, then either $\mathcal{E}$ enters some existential vertex sooner than after n new steps, or he will enter a loop of universal vertices, and then the win of $\mathcal{S}$ will be secured after at most nl new steps (where l is the $\Diamond$-depth of φ).

Hence Lemma 5 follows from Lemma 6 and from the fact that there are only less than n universal vertices in $\mathcal{G}$.

Proof of Lemma 6. If the play $\mathbf{v}$ is like in the lemma then one can clearly find a continuation of this play $\mathbf{v} v_{m+1} \dots v_{m'}$ such that:

1. for each $m+1 \le i \le m'-1$, if v_i is universal then $v_i = v_j$ for some $1 \le j \le m-1$ and $v_{i+1} = v_{j+1}$,

2. either the win of $\mathcal{S}$ is already secured after the play $\mathbf{v}v_{m+1} \dots v_{m'}$ or $v_{m'}$ is a universal vertex which does not occur in $\mathbf{v}v_{m+1} \dots v_{m'-1}$ and $\mathcal{S}$ has a winning strategy after the play $\mathbf{v}v_{m+1} \dots v_{m'}$.

Consider a directed graph $\mathcal{H}$ whose vertices are the elements of the sequence $v_{m+1} \dots v_{m'}$ and such that (w_1, w_2) is an edge of $\mathcal{H}$ if w_1 is existential and (w_1, w_2) is an edge of $\mathcal{G}$, or if w_1 is universal and the move from w_1 to w_2 was already chosen by $\mathcal{E}$ as a part of his persistent strategy (i.e. $w_1 = v_i$ and $w_2 = v_{i+1}$ for some $0 \leq i < m$). Let $\sim$ be an equivalence on the vertices of $\mathcal{H}$ such that $w_1 \sim w_2$ if w_1 and w_2 are reachable from each other in $\mathcal{H}$. Let $\mathcal{H}_0$ be $\mathcal{H}/_\sim$. For two equivalence classes $[w_1]_\sim$ and $[w_2]_\sim$ in $\mathcal{H}_0$ define $[w_1]_\sim \succ [w_2]_\sim$ if $[w_1]_\sim \neq [w_2]_\sim$ and w_2 is reachable from w_1 in $\mathcal{H}$. Let now the sequence $w_1, w_2, \dots w_s$ be such that $w_1 = v_{m+1}$, and w_{i+1} is the first element of $v_{m+1} \dots v_{m'}$ which is right of w_i and such that $w_{i+1} \notin [w_i]_\sim$. Obviously $[w_{i+1}]_\sim \prec [w_i]_\sim$ and so $s \leq n$. Now we construct the sequence $u_1, u_2 \dots u_k$: to do it, we first visit each element of $[w_1]_\sim$. Then we visit them again, and again, l times, where l is like in Lemma 4. This is possible since the elements of $[w_1]_\sim$ are reachable from each other. Then we go to $[w_2]_\sim$ and again visit each vertex of this class l times. Then we do the same for $[w_3]_\sim, \dots [w_s]_\sim$. We stop at $u_k = v_{m'}$. The resulting sequence $u_1, u_2 \dots u_k$ is obviously polynomially long. It is easy to see that if $\mathbf{w} \sqsubseteq v_0 v_1 \dots v_m v_{m+1} \dots v_{m'}$ holds, for some word $\mathbf{w}$ with $|\mathbf{w}| \leq l$, then also $\mathbf{w} \sqsubseteq v_0 v_1 \dots v_m u_1 \dots u_k$ holds. Our claim follows now from Lemma 4. □

5 Proof of Theorem 1

Suppose that M is a Turing machine which, for an input z of length n, uses exponential space, that is space bounded by 2^{n^k} for some integer k. We can assume, without the loss of generality, that the tape alphabet of M is $\{0, 1\}$, and that M has only one accepting configuration. In this configuration the state of M is q_f, and all the cells of the tape contain 0.

Let $z \in \{0, 1\}^*$ be the input word. Let $n = |z|$ and $N = n^k$. We will construct a game $(\mathcal{G}_z, \varphi_z)$ in which $\mathcal{S}$ has a winning strategy if and only if M *does not accept* z. It is easy to verify that this construction can be done in logarithmic space with respect to n.

The game graph $\mathcal{G}_z$, and the formula φ_z will be constructed in such a way that in order to keep φ_z false, $\mathcal{E}$ will need to *declare* in each *stage* s (from 0 up to $2^N - 1$) of the play a triple $\bar{a}(s), \bar{b}(s), \bar{c}(s)$ of configurations of M. This declaration will be understood as his claim that **l:** $\bar{b}(s)$ is reachable from $\bar{a}(s)$ in no more than $2^{(2^N - s + 1)}$ computation steps of M, and **r:** $\bar{c}(s)$ is reachable from $\bar{b}(s)$ in such a number of steps. $\mathcal{E}$ will be also forced to declare $\bar{a}(0)$ as the initial configuration of M on z and $\bar{c}(0)$ as the unique accepting configuration.

At the end of each stage $\mathcal{S}$ will be allowed to say if he wishes to see the proof of **l** or the proof of **r**. If he decides on **l** then $\mathcal{E}$ will be supposed to declare $\bar{a}(s+1) = \bar{a}(s)$ and $\bar{c}(s+1) = \bar{b}(s)$. Analogously, if $\mathcal{S}$ decides on **r** after the stage s, then $\mathcal{E}$ will be supposed to declare $\bar{a}(s+1) = \bar{b}(s)$ and $\bar{c}(s+1) = \bar{c}(s)$. If $\mathcal{E}$

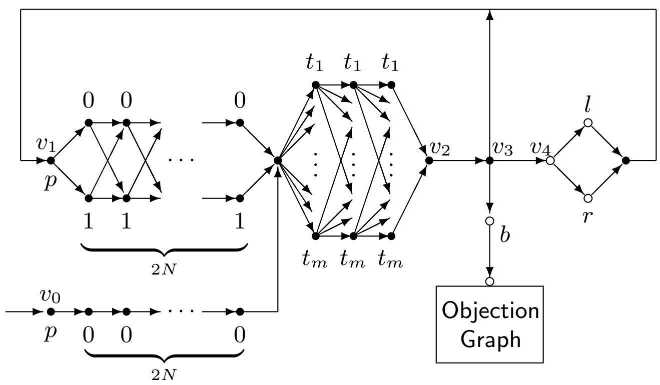

Fig. 3. Graph $\mathcal{G}_z$

would like to cheat here, then finally, when the play reaches the objection graph, $\mathcal{S}$ will have the possibility of *raising objection*, and proving that he was cheated. Finally, φ_z will be written in such a way that the only chance for $\mathcal{E}$ to win will be either to declare $\bar{a}(2^N - 1)$ and $\bar{c}(2^N - 1)$ as equal, or such that $\bar{a}(2^N - 1)$ yields $\bar{c}(2^N - 1)$ in one computation step of M.

5.1 The Game Graph

Let $T = \{t_1, \dots, t_m\} = \{0, 1\} \times (Q \cup \{-\})$, where Q is the set of states of M, and '$-$' is not an element of Q. Notice that $\bar{x} \in T^{2^N}$ can represent a configuration of M. In fact, $\bar{x}_0, \dots \bar{x}_{2^N-1}$ represent values of tape cells. If the head of M is over the i-th cell containing y, and the state of M is q, then $\bar{x}_i = (y, q)$. For all the other cells $\bar{x}_i$ has the form $(y, -)$, where y is the content of cell i.

Graph $\mathcal{G}_z$ is shown in Fig. 3. Vertices are labeled by those atomic propositions which are true at them. Vertices labeled by $t_1, \dots, t_m$ are placed in three columns in such a way that each vertex in the first and the second column is connected with every vertex in the next column. Solid circles represent universal vertices, whereas empty circles are existential. The definition of the objection graph will be given later. Notice that, whenever $\mathcal{E}$ is in the vertex v_1 labeled by p, he can choose any path of length $2N$ of vertices labeled by 0 or 1, thus he can choose any sequence $\bar{x} \in \{0, 1\}^{2N}$ which can be treated as a binary representation of a pair (s, c), where $0 \le s, c < 2^N$. In that case we say that $\mathcal{E}$ *declares* (s, c). The play begins in the vertex v_0, also labeled by p, where $\mathcal{E}$ has to declare $(0, 0)$.

Definition 7. $\mathcal{E}$ plays fair *if and only if the following conditions are satisfied:*

(i) *each time he is in v_1, he declares a pair (s, c) which is the immediate successor of (s', c') declared previously (i.e. $s = s'$ and $c = c' + 1$ if $c' < 2^N - 1$, and $s = s' + 1$ and $c = 0$ if $c' = 2^N - 1$),*

(ii) *each time he is in v_3, immediately after declaring (s, c) for $c < 2^N - 1$, he chooses v_1,*

(iii) *each time he is in* v_3*, immediately after declaring* $(s, 2^N - 1)$ *for* $s < 2^N - 1$*, he chooses* v_4*,*
(iv) *each time he is in* v_3*, immediately after declaring* $(2^N - 1, 2^N - 1)$*, he chooses the vertex labeled by* b*.*

As one can see, if $\mathcal{E}$ plays fair, then he declares each pair from $(0, 0)$ up to $(2^N - 1, 2^N - 1)$ in increasing order. After declaring $(2^N - 1, 2^N - 1)$, $\mathcal{E}$ terminates the play choosing the vertex labeled by b. Furthermore, $\mathcal{E}$, each time after declaring $(s, 2^N - 1)$, goes to vertex v_4, where $\mathcal{S}$ can choose between the vertices labeled by l or r.

Definition 8. *Suppose that* $\mathcal{E}$ *plays fair.*

Define as $a(s,i), b(s,i)$ *and* $c(s,i)$ *the three elements of* T *which are labels of the vertices selected by* $\mathcal{E}$ *from the first, second and third column immediately after declaring* (s,i)*. Let* $\bar{a}(s) = \langle a(s,0), \ldots, a(s, 2^N - 1)\rangle$*,* $\bar{b}(s) = \langle b(s,0), \ldots, b(s, 2^N - 1)\rangle$*, and* $\bar{c}(s) = \langle c(s,0), \ldots, c(s, 2^N - 1)\rangle$*. We say that* $\mathcal{E}$ *declares configurations* $\bar{a}(s), \bar{b}(s), \bar{c}(s)$ *in stage* s*.*

We say that $\mathcal{S}$ answers $h \in \{l, r\}$ in stage s *if and only if he chooses vertex labeled by* h *immediately after* $\mathcal{E}$ *declares* $(s, 2^N - 1)$*. In that case we denote* h *by* $h(s)$*.*

It is easy to check that if $\mathcal{E}$ plays fair then, for each stage s, $\bar{a}(s), \bar{b}(s), \bar{c}(s)$ are well-defined, and for each stage $s < 2^N - 1$, also $h(s)$ is well-defined.

Definition 9. $\mathcal{E}$ plays according to M and z *if and only if he plays fair, and:*

(v) $\bar{a}(0)$ *corresponds to the initial configuration of* M *on the input* z*, and* $\bar{c}(0)$ *corresponds to the accepting configuration of* M*,*
(vi) *either* $\bar{a}(2^N - 1) = \bar{c}(2^N - 1)$*, or the configuration* $\bar{a}(2^N - 1)$ *yields the configuration* $\bar{c}(2^N - 1)$ *in one computation step of* M*,*
(vii) *for each stage* $s \in \{0, \ldots, 2^N - 2\}$*, if* $h(s) = l$ *then* $\bar{a}(s+1) = \bar{a}(s)$ *and* $\bar{c}(s+1) = \bar{b}(s)$*, and similarly, if* $h(s) = r$ *then* $\bar{a}(s+1) = \bar{b}(s)$ *and* $\bar{c}(s+1) = \bar{c}(s)$*.*

Lemma 7. $\mathcal{E}$ *is able to play according to* M *and* z *if and only if* M *accepts* z*.*

Proof. Rewrite the proof of the fact that EXPSPACE = AEXPTIME. □

We will call a formula γ of $LTL_+(\Diamond, \circ, \vee, \wedge)$ *local* if it is small (polynomial) and has the form $\Diamond\gamma'$ where γ' is $\Diamond$-free. By local formulas we can express existence, on an infinite play, of some patterns of polynomial length. One can see that there exists a disjunction φ_1 of local formulas which is valid for exactly those plays which violate one of the conditions (i)-(vi) of Definitions 7 and 9.

Example. The subformula of φ_1 which holds if and only if condition (ii) of Definition 1 is violated could be as follows:

$$\Diamond\big(p \wedge (\circ^{N+1}0 \vee \cdots \vee \circ^{2N}0) \wedge (\circ^{2N+7}(\neg p))\big),$$

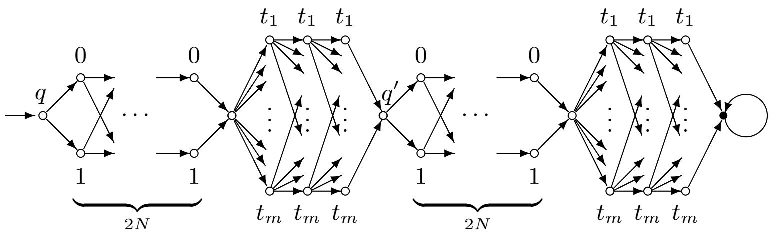

Fig. 4. The Objection Graph

where $\circ^k 0$ stands for the sequence of operators $\circ$ of length k followed by 0. □

Things are more complicated with point (vii) of Definition 9: the formula written in the naive way would be exponentially big. That is because in this case we have to express some relation between two remote fragments of a play. To deal with this problem, we need some participation of $\mathcal{S}$. That is the point where the objection graph is used. In the next section we give the description of the objection graph, the definition of φ_2, and show that $\mathcal{S}$ can make φ_2 true if and only if point (vii) of Definition 9 is violated.

Now, we can define the winning condition of our game:

$$\varphi_z = \varphi_1 \vee \varphi_2.$$

The following lemma is a consequence of Lemma 7, and completes the proof of Theorem 1.

Lemma 8. *$\mathcal{S}$ has the winning strategy in game $(\mathcal{G}_z, \varphi_z)$ if and only if M does not accept z.*

5.2 Raising Objections

In this section we describe a mechanism which allows $\mathcal{S}$ to *raise an objection*, and consequently, to win the game whenever $\mathcal{E}$ violates point (vii) of Definition 9 for some pair of stages s and $s+1$. There are two symmetrical subcases: when $\mathcal{S}$ answers l in the stage s, and when $\mathcal{S}$ answers r in this stage, and so φ_2 will be a disjunction of two symmetrical formulas φ_l and φ_r. We will show how to write the first of them.

Once $\mathcal{S}$ enters the objection graph (Fig. 4) he first declares two numbers of length N. We will call the numbers s^1 and p^1. Then he declares three elements of T, call them a^1, b^1 and c^1, then again two numbers of length N which we call s^2 and p^2, and finally, before the play enters an infinite loop, he declares a^2, b^2 and c^2, again elements of T. One can easily write a local formula ρ expressing the fact that $p^1 = p^2$ and $s^1 + 1 = s^2$ but $a^1 \neq a^2$ or $b^1 \neq c^2$.

Assume that we have a formula ψ_q which is true in vertex v of an infinite play $\mathbf{v}$ if and only if the pattern of length $2N+4$ beginning in the direct successor of v is equal to the pattern of length $2N+4$ beginning in the direct successor of the vertex where q is true. We consider here two patterns to be equal if the same atomic propositions are true in respective vertices of the patterns. Let $\psi_{q'}$ be like ψ_q but with q' instead of q. We can write φ_l as: $\rho \wedge \Diamond(p \wedge \psi_q \wedge \Diamond(l \wedge \Diamond(p \wedge \psi_{q'})))$.

Now, if indeed $\mathcal{E}$ violates point (vii) of Definition 9 in the way described in the beginning of this subsection, then the strategy for $\mathcal{S}$ is to find the number d of a position in the sequence where $\bar{a}(s)$ is not equal to $\bar{a}(s+1)$, or where $\bar{b}(s)$ is not equal to $\bar{c}(s+1)$, enter the objection graph, declare s as s^1, d as p^1, $a(s,d), b(s,d), c(s,d)$ as a^1, b^1 and c^1, then declare $s+1$ as s^2, again d as p^2 and finally $a(s+1,d), b(s+1,d), c(s+1,d)$ as a^2, b^2 and c^2.

It remains to define formula ψ_q:

$$\psi_q = \bigwedge_{i=1}^{2N+4} \psi_q^i,$$

where $\psi_q^i = \big(\bigcirc^i s_1 \wedge \Diamond(q \wedge \bigcirc^i s_1)\big) \vee \cdots \vee \big(\bigcirc^i s_l \wedge \Diamond(q \wedge \bigcirc^i s_l)\big)$, and $\{s_1, \ldots, s_l\} = T \cup \{0, 1\}$.

References

[AT01] R. Alur and S. La Torre, *Deterministic generators and games for LTL fragments*, Proceedings of LICS 2001, Springer Verlag, 2001, pp. 291–300.

[DS98] S. Demri and P. Schnoebelen, *The complexity of propositional linear temporal logics in simple cases*, proceedings of STACS 1998, Springer Verlag, 1998, pp. 61–72.

[MP91] Z. Manna and A. Pnueli, *The temporal logic of reactive and concurent systems*, 1991.

[PR89] A. Pnueli and R. Rosner, *On the synthesis of a reactive module*, Proceedings of 16th ACM POPL, ACM Press, 1989, pp. 179–190.

[SC85] A. P. Sistla and E.M. Clarke, *The complexity of propositional temporal logics*, The Journal of ACM **32** (1985), no. 733, 733–749.

[Tho90] W. Thomas, *Automata on infinite objects*, Handbook of Theoretical Computer Science (J. van Leeuven, ed.), vol. B, Elsevier Science Publishers, 1990, pp. 133–186.

[Tho95] W. Thomas, *On the synthesis of strategies in infinite games*, Proceedings of STACS 1995, LNCS 900, Springer Verlag, 1995, pp. 1–13.