EXPSPACE-Complete Variant of Guarded Fragment with Transitivity*
(Extended Abstract)

Emanuel Kieroński

Institute of Computer Science
University of Wrocław
kiero@ii.uni.wroc.pl

Abstract. We introduce a new fragment $\mathsf{GF}^2 + \overrightarrow{TG}$ of the first order logic – the two-variable guarded fragment with *one-way* transitive guards. This logic corresponds in a natural way to temporal logics without past operators. We prove that the satisfiability problem for $\mathsf{GF}^2 + \overrightarrow{TG}$ is EXPSPACE-complete. The lower bound, obtained for the *monadic* version of the considered logic without equality, improves NEXPTIME lower bound for the whole two-variable guarded fragment with transitive guards $\mathsf{GF}^2 + TG$, given by Szwast and Tendera [8].

1 Introduction

The *guarded fragment* GF of first-order logic was proposed by Andréka, van Benthem and Németi [1] as a generalization of modal and temporal logics. It turned out that GF retains a lot of nice model theoretic properties of modal logic. In particular Grädel proved that it has the finite model property and that its every satisfiable formula has a tree-like model [5]. The satisfiability problem is decidable and has double exponential complexity. The reason for such a high complexity is unrestricted number of variables in formulas. The bounded version GF^k of GF (allowing only k variables) is EXPTIME-complete. In a subsequent paper Grädel and Walukiewicz [7] investigated *guarded fixpoint logic* $\mu\mathsf{GF}$. They proved that adding least and greatest fixed point operators gives a decidable logic, and moreover, the complexity does not change, i.e. the satisfiability problem for $\mu\mathsf{GF}$ is in 2EXPTIME and for $\mu\mathsf{GF}^k$ – in EXPTIME.

Another question is what happens after adding transitivity to GF, i.e. when we may specify some binary relations as transitive. This question was answered by Ganzinger, Meyer and Veanes [4] who proved that even the two-variable version of such a logic is undecidable. On the other hand they proposed a restricted, *monadic*, two-variable variant $\mathsf{MGF}^2 + TG$ in which all binary relations can appear only in guards but can be required to be transitive. Ganzinger, Meyer and Veanes showed that the satisfiability problem for this variant is decidable but they gave no complexity bound since their result was obtained by reduction to

* Supported by KBN grant 8 T11C 043 19

the Rabin's theory SkS. They also left another open question: the decidability of $\mathsf{GF}+TG$ – the whole guarded fragment with transitive relations where transitive relations are admitted only in guards, but where non-transitive relations and equality can occur elsewhere. The last question was answered recently by Szwast and Tendera [8] who proved that $\mathsf{GF}+TG$ is decidable and after adding transitive guards to GF (similarly to adding fixed point operators) the complexity of the satisfiability problem remains in 2EXPTIME.

Surprisingly, the two-variable fragment behaves differently after adding transitive guards. Szwast and Tendera observed that the satisfiability problem for $\mathsf{GF}^2 + TG$ is NEXPTIME-hard (recall that for GF^2 it is EXPTIME-complete). Their elegant argument goes through a reduction from FO^2 which was shown to be NEXPTIME-complete by Grädel, Kolaitis and Vardi [6]. In this paper we strengthen this result and show that even a very restricted variant of MGF^2+TG is EXPSPACE-hard. Our proof proceeds by a reduction from the problem: "Does an alternating Turing machine M with exponentially bounded time accepts its input w?"

We propose an interesting, nontrivial guarded logic $\mathsf{GF}^2 + \overrightarrow{TG}$ with transitive guards, for which the satisfiability problem is EXPSPACE-complete. We call this variant the two-variable guarded fragment with *one-way* transitive guards since we require all transitive guards of quantifiers $\forall x$ ($\exists x$) to be of the form $T(y, x)$ (guards $T(x, y)$ are forbidden). This is similar to the situation in most temporal logics where you can quantify points in the future but not in the past. A lot of modal and temporal logics including K4, S4 can be encoded in $\mathsf{GF}^2 + \overrightarrow{TG}$. The key observation in our proof is that we can construct *regular* forrest-like models of satisfiable formulas. All important information in such models is stored at the first k levels of trees, where k is at most exponential with respect to the size of the formula. Moreover paths in regular forrest-like models can be constructed independently.

2 Preliminaries

Andréka, van Benthem and Németi [1] defined the *guarded fragment* of first order logic GF inductively:

(1) every atomic formula belongs to GF,
(2) GF is closed under $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$,
(3) if $\boldsymbol{x}, \boldsymbol{y}$ are tuples of variables, $\alpha(\boldsymbol{x}, \boldsymbol{y})$ is an atomic formula, $\psi(\boldsymbol{x}, \boldsymbol{y})$ is in GF and the set of free variables of ψ is contained in the set of free variables of α then formulas $\exists \boldsymbol{y}(\alpha(\boldsymbol{x}, \boldsymbol{y}) \wedge \psi(\boldsymbol{x}, \boldsymbol{y}))$ and $\forall \boldsymbol{y}(\alpha(\boldsymbol{x}, \boldsymbol{y}) \rightarrow \psi(\boldsymbol{x}, \boldsymbol{y}))$ are in GF.

Atoms $\alpha(\boldsymbol{x}, \boldsymbol{y})$ in the above definition are called *guards*. We write GF^k to denote the language obtained by restricting the number of variables in formulas by a constant k (variables can be reused). The *monadic guarded fragment* MGF is a variant in which all non-unary relation symbols may appear only as guards. Ganzinger, Meyer and Veanes [4] investigated $\mathsf{MGF}^2 + TG$ – the two-variable monadic guarded fragment where some binary relations are assumed to be transitive. By $\mathsf{GF}+TG$ we denote the whole guarded fragment with transitive relations where transitive relations are only admitted in guards. GF^2+TG is the two-variable version of this logic.

In this paper we consider the two-variable guarded fragment with *one-way* transitive guards $\mathsf{GF}^2+\overrightarrow{TG}$. Formally, a guarded formula is in $\mathsf{GF}^2+\overrightarrow{TG}$ if it is in GF^2+TG and additionally in all transitive guards of quantifiers $\forall x$ $(\exists x)$ the quantified variable x appears in the second place of the guarding predicate. It means that we allow formulas of the form $\forall x\ T(y,x) \wedge \psi(x,y)$ and forbid formulas of the form $\forall x\ T(x,y) \wedge \psi(x,y)$ (and similarly for the existential quantifier). For non-transitive guards the order of variables can be arbitrary.

Observe that the monadic variant of $\mathsf{GF}^2+\overrightarrow{TG}$ is still, as the variant of Ganzinger, Meyer and Veanes, a nontrivial extension of the *modal fragment* with transitivity since it does not retain the finite model property. A simple example of $\mathsf{MGF}^2+\overrightarrow{TG}$ formula which has only infinite models is : $\exists x\ P(x) \wedge \forall x\ \big(P(x) \rightarrow \exists y\ (x<y\ \wedge\ P(y))\big)\ \wedge\ \neg\exists x\ \big(x<x\big)$, where relation $<$ is transitive.

A *1-type* $t(x)$ over a fixed signature is a maximal consistent set of atomic and negated atomic formulas in variable x. A *2-type* $t(x,y)$ is similarly defined as a maximal consistent set of atomic and negated atomic formulas in variables x,y. We often identify a type with a conjunction of all its formulas. Let $\mathcal{A}$ be a relational structure and $a \in A$. We denote by $type^{\mathcal{A}}(a)$ the unique type $t(x)$ such that $\mathcal{A} \models t(a)$. Similarly for $a,b \in A$, $type^{\mathcal{A}}(a,b)$ is the unique 2-type $t(x,y)$ such that $\mathcal{A} \models t(a,b)$.

An *alternating Turing machine* is a generalization of a nondeterministic Turing machine. States, and hence configurations of such a machine, are partitioned into four groups: existential, universal, accepting and rejecting. The notion of an accepting (rejecting) configuration is extended to the case of configurations in which states are existential or universal. This can be done inductively: an universal configuration is accepting if all its successor configurations, i.e. configurations obtained by performing one machine step according to the transition function, are accepting and an existential configuration is accepting if at least one of its successor configurations is accepting. A machine M accepts its input w if the initial configuration of M on w is accepting. By ATIME(f) we denote the set of problems that can be solved by an alternating Turing machine working in time bounded by a function f. In this paper we will use the following theorem:

Theorem 1. (Chandra, Kozen, Stockmeyer [3]) *For $t(n) \geq n$:*

$$\bigcup_{k=1}^{\infty} ATIME[(t(n))^k] = \bigcup_{k=1}^{\infty} DSPACE[(t(n))^k].$$

If we denote $\bigcup_{k=1}^{\infty} ATIME(2^{n^k})$ by AEXPTIME we have in particular AEXPTIME=EXPSPACE.

For details about alternating Turing machines you can see for example [2].

3 The Proof of Hardness

Szwast and Tendera [8] argued that $\mathsf{GF}^2 + TG$ is NEXPTIME-hard. Here we strengthen their result and prove that even $\mathsf{MGF}^2 + \overrightarrow{TG}$ without equality, with just one transitive relation symbol (which is the only binary symbol) is EXPSPACE-hard.

Theorem 2. *The satisfiability problem for* $\mathsf{MGF}^2 + \overrightarrow{TG}$ *without equality is EXPSPACE-hard.*

Proof. By Theorem 1 every problem in EXPSPACE is in AEXPTIME so let M be an alternating Turing machine deciding a problem in EXPSPACE in time 2^{n^k}. We can assume without loss of generality that every non-final configuration of M has exactly two successor configurations. Let w be an input of size n. We construct a formula Φ whose models encode accepting computation trees of machine M on input w. Every configuration is represented by 2^{n^k} elements of a model, each of them corresponds to a single cell of the tape. We introduce unary relation symbols $C_1, \ldots, C_{n^k}$ and $P_1, \ldots, P_{n^k}$ to encode the number of configuration (i.e. the depth of the configuration in a computation tree) to which an element belongs and the position of this element (the number of a cell) in this configuration respectively ($C_i(x)$ is true if the i-th bit of the number of the configuration to which x belongs is 1, $P_i(x)$ is true if the i-th bit of the position of x in the configuration is 1). We use abbreviations $\bar{C}(x)$ and $\bar{P}(x)$ to describe the above numbers. It is not difficult to express the following properties with quantifier-free formulas of polynomial length:

$$\begin{array}{c}
\bar{C}(x) = l \quad \text{for fixed } l, \quad 0 \le l < 2^{n^k} \ , \quad \bar{P}(x) = l \quad \text{for fixed } l, \quad 0 \le l < 2^{n^k} \ , \\
\bar{C}(x) \ge l \quad \text{for fixed } l, \quad 0 \le l < 2^{n^k} \ , \quad \bar{P}(x) \ge l \quad \text{for fixed } l, \quad 0 \le l < 2^{n^k} \ , \\
\bar{C}(x) = \bar{C}(y) \ , \quad \bar{P}(x) = \bar{P}(y) \ , \\
\bar{C}(x) = \bar{C}(y) + 1 \ , \quad \bar{P}(x) = \bar{P}(y) + 1 \ .
\end{array}$$

We describe a configuration in a standard way: for each symbol a_i from the alphabet of M (including **blank**) we use an unary relation symbol A_i, for each state q_i – an unary symbol Q_i. We also have an unary symbol H describing the head position. For element x representing a tape cell scanned by the head, $H(x)$ and $Q_i(x)$ (for some i) will be true.

Let us start the construction of Φ with enforcing that its every model $\mathcal{A}$ contains a binary tree of depth 2^{n^k}. Every node of the tree consists of 2^{n^k} elements describing a single configuration of M. We define a relation $<$ with the intended meaning: $x < y$ if x and y belong to a description of the same configuration (they belong to the same node of the tree) and x describes the tape cell with the smaller number, or y belongs to a description of a configuration reachable from the configuration x belongs to. Let the relation $<$ be transitive. We want to distinguish between successors of a node. Therefore we introduce a relation symbol L that is true for elements belonging to the left son of some node. We assume that for elements belonging to the initial configuration L is also true. In addition we use a special symbol N that is true for all elements belonging to the description of the computation tree. For further purposes we abbreviate the formula $\bar{C}(x) = \bar{C}(y) \wedge \bar{P}(y) = \bar{P}(x) + 1$ to $NEXT(x,y)$.

$$\exists x \ \big(N(x) \wedge L(x) \wedge \bar{C}(x) = 0 \wedge \bar{P}(x) = 0\big) \quad ,$$

$$\begin{aligned} \forall x \ \Big(& N(x) \Rightarrow \bar{P}(x) \neq 2^{n^k} - 1 \\ & \quad \Rightarrow \ \exists y \ \big(x < y \wedge N(y) \wedge (L(x) \Leftrightarrow L(y)) \wedge NEXT(x,y)\big)\Big) \quad , \end{aligned}$$

$$\begin{aligned} \forall x \ \Big(& N(x) \Rightarrow (\bar{P}(x) = 2^{n^k} - 1 \wedge \bar{C}(x) \neq 2^{n^k} - 1) \\ & \Rightarrow \ \exists y \ \big(x < y \wedge N(y) \wedge L(y) \wedge \bar{P}(y) = 0 \wedge \bar{C}(y) = \bar{C}(x) + 1\big) \\ & \quad \wedge \exists y \ \big(x < y \wedge N(y) \wedge \neg L(y) \wedge \bar{P}(y) = 0 \wedge \bar{C}(y) = \bar{C}(x) + 1\big)\Big) \quad . \end{aligned}$$

Observe that in fact a structure defined by the above formulas does not need to contain a "real" tree (for example a node can be a son of two or more distinct nodes) but as we see later it is not important.

Now we state that a model of our formula fulfils several basic properties of an accepting computation tree. There is exactly one alphabet symbol in every tape cell:

$$\forall x \ \big(N(x) \Rightarrow \bigvee_i A_i(x)\big) \quad \wedge \quad \bigwedge_i \big(\forall x \ (A_i(x) \Rightarrow \bigwedge_{i \neq j} \neg A_i(x))\big) \quad .$$

In each configuration at most one element is scanned by the head:

$$\forall x, y \ \big(x < y \Rightarrow \bar{C}(x) = \bar{C}(y) \Rightarrow \neg(H(x) \wedge H(y))\big) \quad .$$

Exactly those elements which represent tape cells observed by the head store information about state:

$$\bigwedge_i \big(\forall x \ (Q_i(x) \Rightarrow H(x))\big) \quad \wedge \quad \forall x \ \big(H(x) \Rightarrow \bigvee_i Q_i(x)\big) \quad .$$

The root of the tree describes the initial configuration of M on input $w = w_0 \ldots w_{n-1}$:

$$\begin{gathered} \forall x \ \big(N(x) \Rightarrow (\bar{C}(x) = 0 \wedge \bar{P}(x) = 0) \Rightarrow H(x)\big) \quad , \\ \bigwedge_{i<n} \big(\forall x \ \big(N(x) \Rightarrow (\bar{C}(x) = 0 \wedge \bar{P}(x) = i) \Rightarrow W_i(x)\big)\big) \quad , \\ \forall x \ \big(N(x) \Rightarrow (\bar{C}(x) = 0 \wedge \bar{P}(x) \geq n) \Rightarrow BLANK(x)\big) \quad . \end{gathered}$$

If a tape cell of a configuration is not scanned by the head then in the same cell of both successor configurations the alphabet symbol does not change:

$$\begin{aligned} \forall x, y \ \big(x < y \Rightarrow & (\neg H(x) \wedge \bar{C}(y) = \bar{C}(x) + 1 \wedge \bar{P}(x) = \bar{P}(y)) \\ & \qquad \Rightarrow \ \bigwedge_i (A_i(x) \Leftrightarrow A_i(y))\big). \end{aligned}$$

Consider now a node t of a tree and a configuration c that is described by this node. There are two cases: the state of the machine in this configuration is existential or it is universal. In the first case we enforce that the configuration represented by the left son of t is created by applying one of the two possible transitions on c (we do not say anything about the right son of t in this case). Assume that for an existential state q and a letter a there are two possible transitions: $(q,a) \Rightarrow (q', a', \rightarrow)$ and $(q,a) \Rightarrow (q'', a'', \leftarrow)$. We write such a formula:

$$\begin{aligned}\forall x, y \Big(x < y \Rightarrow (&A(x) \wedge Q(x) \wedge H(x) \wedge L(y) \\ &\wedge \bar{C}(y) = \bar{C}(x) + 1 \wedge \bar{P}(x) = \bar{P}(y) + 1) \\ \Rightarrow \big(&\forall x\ (y < x \Rightarrow NEXT(y,x) \\ &\Rightarrow A'(x) \wedge \forall y\ (x < y \Rightarrow NEXT(x,y) \Rightarrow Q'(y) \wedge H(y))) \\ &\vee (Q''(y) \wedge H(y) \wedge \forall x\ (y < x \Rightarrow NEXT(y,x) \Rightarrow A''(y))))\Big)\ .\end{aligned}$$

Other possible situations (both transitions move the head forward or both transitions move the head backward) can be handled similarly.

Consider now the case of universal configuration. We enforce that the left son of t is created by applying the first transition and the right son – the second one. For an universal state q, a letter a and transitions like above we write formula:

$$\begin{aligned}\forall x, y \Big(x < y \Rightarrow (&A(x) \wedge Q(x) \wedge H(x) \wedge \bar{C}(y) = \bar{C}(x) + 1 \wedge \bar{P}(x) = \bar{P}(y) + 1) \\ \Rightarrow \big(&L(y) \Rightarrow \big(\forall x\ (y < x \Rightarrow NEXT(y,x) \\ &\Rightarrow A'(x) \wedge\ \forall y\ (x < y \Rightarrow NEXT(x,y) \Rightarrow Q'(y) \wedge H(y)))) \\ &\wedge \neg L(y) \Rightarrow \big(Q''(y) \wedge H(y) \wedge \forall x\ (y < x \Rightarrow NEXT(y,x) \Rightarrow A''(y))))\big)\Big)\ .\end{aligned}$$

We add formulas of the above form for every pair of a state and a letter.

To finish our construction we ensure that the machine never enters the rejecting state (we assume that q_r is the only rejecting state):

$$\forall x\ (Q_r(x) \Rightarrow \mathbf{false})\ .$$

We define Φ as the conjunction of all the above formulas. Observe that the number of conjuncts and size of each of them are polynomial in n. We have the following fact: Φ is satisfiable iff M accepts w. Indeed, from each model of Φ we can extract a description of an accepting computation tree – even if a model represents some distinct tape cells (or for example whole configurations) by the same element (or the same node respectively) we can make a copy of this element (node). Conversely, if M accepts w then a model of Φ can be constructed easily from an accepting computation tree. □

4 The Upper Bound

In this section we prove that the satisfiability problem for $\mathsf{GF}^2 + \overrightarrow{TG}$ is in EXPSPACE. Szwast and Tendera [8] gave double exponential algorithm for the satisfiability of the whole GF+TG. To obtain this result they established a kind of the tree-model property – from an arbitrary model of a formula they construct a so-called *ramified model*. Of course such a construction can be repeated also in our case, but to achieve a lower complexity we need a stronger result. Therefore we prove that every satisfiable $\mathsf{GF}^2 + \overrightarrow{TG}$ sentence Φ in normal form has a *regular forrest-like model*. A regular forrest-like model can be represented as a set of disjoint trees whose nodes are elements of the model. For every path in a tree, longer than $2^{O(|\Phi|)}$, there exists an element a on this path, belonging to the m-th level, $m < 2^{O(|\Phi|)}$, such that the subtree rooted in a is isomorphic to a subtree rooted in an element b which lies on the path from the root of the tree to a or is a son of another element lying on this path.

4.1 The Normal Form

Let us begin with an adaptation of the normal form theorem for $\mathsf{GF}{+}TG$ [8].

Definition 1. *A $\mathsf{GF}^2 + \overrightarrow{TG}$ sentence Φ is in normal form if it is a conjunction of sentences of the following form:*

$$\begin{aligned}
&(e1)\ \exists x\ (U(x)\ \wedge\ \phi(x))\ ,\\
&(w1)\ \forall x\ (U(x) \to \exists y\ (B(x,y)\ \wedge\ \phi(x,y)))\ ,\\
&(w2)\ \forall x\ (U(x) \to \exists y\ (B(y,x)\ \wedge\ \phi(x,y)))\ ,\\
&(w3)\ \forall x\ (U(x) \to \exists y\ (T(x,y)\ \wedge\ \phi(x,y)))\ ,\\
&(b1)\ \forall x,y\ (B(x,y) \to \phi(x,y))\ ,\\
&(b2)\ \forall x,y\ (T(x,y) \to \phi(x,y))\ ,\\
&(u1)\ \forall x\ (U(x) \to \phi(x))\ ,\\
&(u2)\ \forall x\ (B(x,x) \to \phi(x))\ ,\\
&(u3)\ \forall x\ (T(x,x) \to \phi(x))\ ,
\end{aligned}$$

where U stands for an unary relation symbol, B stands for a binary symbol which is not transitive, T stands for a transitive binary symbol and ϕ is quantifier-free.

Lemma 1. *For every $\mathsf{GF}^2 + \overrightarrow{TG}$ sentence Φ of the length n there exists a set $\{\psi_1, \ldots, \psi_d\}$ of $\mathsf{GF}^2 + \overrightarrow{TG}$ sentences in normal form such that:*

- *Φ is satisfiable if and only if at least one of the formulas $\psi_1, \ldots, \psi_d$ is satisfiable;*
- *d is at most exponential with respect to n and for every $i \le d$ the length of ψ_i is polynomial with respect to n;*
- *the set $\{\psi_1, \ldots, \psi_d\}$ can be computed deterministically in exponential time.*

Proof given by Szwast and Tendera [8] works also for our case since it retains the number of variables and does not change the "orientation" of transitive guards.

4.2 Cliques and Ramified Models

Let us recall some definitions and facts from [8].

Definition 2. *Let A be a structure over a fixed language σ and let $\mathcal{C}$ be a substructure of $\mathcal{A}$. For a transitive symbol T we say that $\mathcal{C}$ is a* T-clique *if for every $a, b \in C$ we have $\mathcal{A} \models T(a,b)$. For a binary symbol R and $a \in A$ we denote by $[a]^{\mathcal{A}}_R$ the maximal R-clique containing a or the one-element substructure of $\mathcal{A}$ containing a if such a clique does not exist or R is not a transitive symbol.*

Szwast and Tendera observed that in $\mathsf{MGF}^2{+}TG$ we can define cliques of size exponential with respect to the number of relation symbols in the language σ. Such cliques can be also defined in $\mathsf{MGF}^2{+}\overrightarrow{TG}$. Here is an example. The language σ consists of unary relation symbols $N, P_1, \ldots P_n$ and a transitive binary symbol $<$. We use abbreviation $\overline{P}$ introduced in the hardness proof:

$$\exists x\ (N(x) \wedge \bar{P}(x) = 0)\ ,$$
$$\forall x\ (N(x) \Rightarrow \bar{P}(x) < 2^n - 1 \Rightarrow \exists y\ (x < y \wedge \bar{P}(y) = \bar{P}(x) + 1))\ ,$$
$$\forall x\ (N(x) \Rightarrow \bar{P}(x) = 2^n - 1 \Rightarrow \exists y\ (x < y \wedge \bar{P}(y) = 0))\ ,$$
$$\forall x\ (N(x) \Rightarrow \bar{P}(x) = 0 \Rightarrow \neg\exists y\ (x < y \wedge x \neq y \wedge \bar{P}(y) = 0))\ .$$

Definition 3. *Let Φ be a $\mathsf{GF}^2 + TG$ sentence in the normal form. A structure $\mathcal{A}$ is a* ramified model *for Φ if the following conditions hold:*

1. *$\mathcal{A} \models \Phi$,*
2. *for every $a, b \in A, a \neq b$ there is at most one transitive relation symbol T such that $\mathcal{A} \models T(a,b) \vee T(b,a)$,*
3. *if S and T are distinct transitive relation symbols then for every $a, b, c \in A$, such that $a \neq b$ and $a \neq c$ if $b \in [a]^{\mathcal{A}}_S$ and $c \in [a]^{\mathcal{A}}_T$ then for every binary symbol R we have $\mathcal{A} \not\models R(b,c)$,*
4. *for every $a \in A$, for every binary symbol T, the cardinality of $[a]^{\mathcal{A}}_T$ is at most exponential with respect to the number of relation symbols in the language.*

Theorem 3. (Szwast, Tendera [8]) *Every satisfiable $\mathsf{GF}^2 + TG$ sentence has a ramified model.*

In our construction of a regular forrest-like model for a satisfiable $\mathsf{GF}^2 + \overrightarrow{TG}$ sentence, we use only part 4 of Definition 3 (but we obtain also the remaining properties).

4.3 Regular Forrest-Like Models

In this section we are going to define the notion of a regular forrest-like model for a $\mathsf{GF}^2 + \overrightarrow{TG}$ sentence in the normal form. We will work with trees whose nodes are elements of the model and whose edges are labelled with "oriented" binary relation symbols. More precisely for a non-transitive symbol B labels have the form $\overleftarrow{B}$ and $\overrightarrow{B}$, for a transitive T we use labels $\overrightarrow{T}$ and $\overleftrightarrow{T}$. We define $L(a)$ to be the label of the edge connecting a with its father if the father exists or ϵ in the other case. By $\overline{L}(a)$ we denote the relation symbol in $L(a)$. A *path* in a tree is a maximal, possibly infinite sequence of nodes $s_1, \ldots, s_k, \ldots$ such that for $i > 0$, s_{i+1} is a son of s_i. For a transitive symbol T, a *T-path* is a maximal, possibly infinite sequence of nodes $s_1, \ldots, s_k, \ldots$ such that for $i > 0$, s_{i+1} is a son of s_i and the edge between s_i and s_{i+1} is labelled with $\overrightarrow{T}$ or $\overleftrightarrow{T}$. For convenience we will also use the notion of a B-path for a non-transitive symbol B but in this case it is rather unnatural and denotes a pair of elements connected by an edge labelled with $\overrightarrow{B}$ or $\overleftarrow{B}$.

By $tree(a)$ we denote the subtree rooted in a. For a relation symbol R and a node a let $tree^R(a)$ denote the fragment of $tree(a)$ such that it contains subtrees rooted in all nodes connected with a by edges labelled with $\overleftrightarrow{R}$, $\overrightarrow{R}$ or $\overleftarrow{R}$. We say that an element a *R-precedes* an element b, for a binary symbol R, if a and b lie on the same R-path and a is an ancestor of b or, in the case of transitive R, there exists an element c such that c and b lie on the same R path, c is an ancestor of b, a is a son of c and the edge between a and c is labelled with $\overleftrightarrow{R}$. For convenience we denote by $\pi(a)$ the element a if $a \in L_0$ or the initial fragment of the $\overline{L}(a)$-path ending at element a, and by $\Pi(a)$ the initial fragment of the path ending at a. Let $prec(a)$ denote the set consisting of the $type(a)$ and the set of 1-types of elements that $\overline{L}(a)$-precede a.

Regular forrest-like models are ramified models with some additional properties. Let us describe it informally first. A regular forrest-like model can be represented as a set of disjoint trees whose roots are elements fulfilling existential conjuncts of the form (e1). Labels of edges describe certain properties of 2-types of pairs of elements. If elements a, b lie on the same B-path (B is not transitive) then the $type(a, b)$ contains $B(x, y)$ or $B(y, x)$ depending on the orientation of the label of the edge between a and b. If a T-precedes b then $type(a, b)$ contains $T(x, y)$, moreover it contains $T(y, x)$ if and only if a and b belong to the same T-clique. We will construct models in such a way that 2-types of pairs of elements will not contain any other positive occurrences of binary transitive symbols. All elements belonging to the same T-clique are stored at two consecutive levels of a tree. One of elements, say a, belongs to the i-th level and all the other elements of the clique belong to the $(i+1)$-st level. There are edges between a and the remaining elements of the clique and these edges are labelled with $\overleftrightarrow{T}$.

Let ϕ be a formula of the form $\forall x\ (U(x) \rightarrow \exists y \gamma(x, y))$. If for an element a, $U(a)$ is true in the model and for an element b, $\gamma(a, b)$ is true in the model then we say that b is a *witness* for a and ϕ. Witnesses for an element a, belonging to the i-th level, and formulas of the form (w1)-(w2), i.e. formulas without transitive guards, are located in the $(i+1)$-st level. A situation can be different for formulas of the form (w3) $\forall x\ (U(x) \rightarrow \exists y\ (T(x, y)\ \wedge\ \phi(x, y)))$. In this case we look for a witness in the T-clique of a first. We do so since the equality symbol can appear in formulas and for example it can be specified that on a T-path there can be at most one element of a certain 1-type.

The most important feature of our models is regularity – all subtrees added after exponential number of levels are isomorphic in a sense to some subtrees added before. It means that to answer if a formula has a model it is enough to check if we can construct an exponential number of levels of a regular forrest-like structure. Since elements of our model can be connected by binary relations only if one of them R-precedes the second, for some binary R, we can construct all the paths of a model independently. This fact leads to an algorithm using only exponential space.

Now we are ready to give a precise definition of the notion of the regular forrest-like model.

Definition 4. *A* regular forrest-like model *for a* $\mathsf{GF}^2 + \overrightarrow{TG}$ *sentence* Φ *in normal form is a structure* $\mathcal{M}$ *such that:*

1. $\mathcal{M} \models \Phi$;
2. $\mathcal{M}$ *can be represented as a set of disjoint trees whose nodes are elements of* M *and edges are labelled with "oriented" binary relation symbols, i.e. symbols of the form* $\vec{R}$, $\overset{\leftarrow}{R}$ *or* $\overset{\leftrightarrow}{R}$;
3. *the universe of* $\mathcal{M}$ *can be split into levels* $L_0, L_1, \ldots$ *where* L_0 *consists of roots of trees and for* $i > 0$ *elements of* L_i *are sons of elements from* L_{i-1};
4. *the number of elements in* L_0 *is at most polynomial, and the number of sons of each element in a tree is at most exponential with respect to* $|\Phi|$;
5. *if* a *is an element of a* T*-clique and the father of* a *does not exist or does not belong to the same* T*-clique then all remaining elements of* $[a]_T^{\mathcal{M}}$ *are sons of* a *and edges between* a *and these elements are labelled with* $\overset{\leftrightarrow}{T}$;
6. *for every conjunct of* Φ *of the form* $\exists x\ (U(x)\ \wedge\ \phi(x))$ *there exists an element* $a \in L_0$ *such that* $\mathcal{M} \models U(a)\ \wedge\ \phi(a)$;
7. *for every conjunct of* Φ *of the form* $\forall x\ (U(x)\ \rightarrow\ \exists y\ \gamma(x,y))$ *and every element* $a \in L_i$ *such that* $\mathcal{M} \models U(a)$ *one of the following conditions holds:*
   - *there exists a son* $b \in L_{i+1}$ *of* a *such that* $\mathcal{M} \models \gamma(a,b)$*; moreover if* γ *is of the form* $R(x,y) \wedge \phi(x,y)$ *then the edge between* a *and* b *is labelled with* $\vec{R}$ *or* $\overset{\leftrightarrow}{R}$ *and if* γ *is of the form* $R(y,x) \wedge \phi(x,y)$ *then this edge is labelled with* $\overset{\leftarrow}{R}$;
   - b *is the father of* a*, the edge between* a *and* b *is labelled with* $\overset{\leftrightarrow}{T}$*,* γ *is of the form* $T(x,y) \wedge \phi(x,y)$ *and* $\mathcal{M} \models \gamma(a,b)$;
   - b *is the father of* a*,* c *is a son of* b *(*$c \neq a$*), the edges between* b *and* a*,* b *and* c *are labelled with* $\overset{\leftrightarrow}{T}$*,* γ *is of the form* $T(x,y) \wedge \phi(x,y)$ *and* $\mathcal{M} \models \gamma(a,c)$;
   - $\mathcal{M} \models \gamma(a,a)$;
8. *if* T *is a transitive symbol and the edge between* a *and* b *is labelled with:*
   - $\vec{T}$ *then* $\mathcal{M} \models T(a,b) \wedge \neg T(b,a)$;
   - $\overset{\leftarrow}{T}$ *then* $\mathcal{M} \models T(b,a) \wedge \neg T(a,b)$;
   - $\overset{\leftrightarrow}{T}$ *then* $\mathcal{M} \models T(a,b) \wedge T(b,a)$;
9. *for elements* a, b, $a \neq b$ *if there is no relation symbol* R *such that* a R*-precedes* b *or* b R*-precedes* a *in a tree then for every binary symbol* R', $\mathcal{M} \not\models R'(a,b)$;

10. *for elements a, b, $a \neq b$ if there is no transitive symbol T such that a T-precedes b or b T-precedes a then for every transitive symbol T', $\mathcal{M} \not\models T'(a,b)$;*
11. *if an element a T-precedes b (for a transitive T) then for every transitive symbol $T' \neq T$, $\mathcal{M} \not\models T'(a,b)$ and $\mathcal{M} \not\models T'(b,a)$;*
12. *for every path $s_1, \ldots, s_k, \ldots$ which is infinite or longer than $2^{O(|\Phi|)}$, there exist constants $c < d \leq 2^{O(|\Phi|)}$, such that for $x = s_c$ or x being a son of s_c (in the latter case $c < d-1$ and x does not belong to the path $s_1, \ldots, s_k, \ldots$), tree(s_d) $\cong$ tree(x), i.e. there exists a bijection I between nodes of tree(s_d) and nodes of tree(x) such that if $I(a) = b$ then $type^{\mathcal{M}}(a) = type^{\mathcal{M}}(b)$ and if $I(a_1) = b_1$, $I(a_2) = b_2$ and a_2 is a son of a_1 then b_2 is a son of b_1 and $L(a_2) = L(b_2)$.*

Lemma 2. *Every satisfiable $\mathsf{GF}^2 + \overrightarrow{TG}$-sentence Φ in normal form has a regular forrest-like model.*

Proof. Unfortunately, due to the space limit, we are not able to give here a detailed proof. Therefore we present only a sketch of our construction. Proof will be given in the full version of this paper which will be available at the following URL: `http://ii.uni.wroc.pl/~kiero`.

Let $\mathcal{A}$ be a model of Φ fulfilling part 4 of Definition 3. We construct a regular forrest-like model $\mathcal{M}$ of Φ recursively:

Stage 0. For every conjunct of Φ of the form $\exists x\ \phi(x)$ find an element $a \in A$ such that $\mathcal{A} \models \phi(a)$ and add a copy of this element (i.e. an element of the same 1-type) to L_0.

Stage k. For every element $a \in L_{k-1}$, for which $tree(a)$ is not defined:

1. if $L(a) = \overrightarrow{T}$ for a transitive symbol T and there exists an element $b \in \pi(a)$ such that $L(b) \neq \overleftrightarrow{T}$ (i. e. $L(b) = \overrightarrow{T}$ or b is the first element of $\pi(a)$), $type^{\mathcal{M}}(b) = type^{\mathcal{M}}(a)$ and $prec(b) = prec(a)$ then take a copy of $tree(b)$ as $tree(a)$,
2. if $L(a) = \overrightarrow{B}$ or $L(a) = \overleftarrow{B}$ for non-transitive symbol B then if there exists an element b in $\Pi(a)$ such that $L(b)$ is not of the form $\overleftrightarrow{T}$ and $type^{\mathcal{M}}(b) = type^{\mathcal{M}}(a)$ then take a copy of $tree(b)$ as $tree(a)$;
3. if none of the previous steps was performed and there exists an element $b \in \Pi(a)$ such that $L(b) = L(a)$ and $type^{\mathcal{M}}(b) = type^{\mathcal{M}}(a)$ then for every binary symbol R, except the symbol $\overline{L}(a)$, take a copy of $tree^R(b)$ as $tree^R(a)$ (after this step $[a]^{\mathcal{M}}_R$ is defined for such R);
4. for every transitive symbol T, if $[a]^{\mathcal{M}}_T$ is not defined then for every element $b' \in A$, belonging to the clique of the element $a' \in A$ which a is a copy of, add a copy of this element to L_k and set the label of the edge between a and this copy to $\overleftrightarrow{T}$;
5. for every conjunct γ of the form $\forall x\ (U(x) \rightarrow \exists y\ \delta(x,y) \wedge \phi(x,y)))$ if $U(x) \in type^{\mathcal{M}}(a)$ and there is no witness for a and γ in the clique of a then find a witness b' for γ and the element $a' \in A$ which a is a copy of, add a copy of b' to L_k and set the label between a and this copy to $\overrightarrow{T}$.

For each pair a, b of elements such that for every binary R, a does not R-precede b and b does not R-precede a, $type^{\mathcal{M}}(a, b)$ contains no positive occurences of binary symbols. For each pair a, b such that a R-precedes b for a symbol R, $type^{\mathcal{M}}(a, b)$ is based on the type of some pair of elements from A and constructed according to the definition. Parts 1-3 of the Stage k guarantee that part 12 of definition 4 is fulfilled. $\square$

An alternating procedure working in exponential time can be naturally derived from the construction. This procedure, Lemma 1 and Theorems 1, 2 allow us to state the main result of the paper:

Theorem 4. *Satisfiability problem for* $\mathsf{GF}^2 + \overrightarrow{TG}$ *is EXPSPACE-complete.*

5 Conclusion

In this paper we studied the complexity of the two-variable guarded fragment with one-way transitive guards $\mathsf{GF}^2 + \overrightarrow{TG}$. We proved that the satisfiability problem for this logic can be solved in exponential space. We showed that this is optimal since even for monadic version of this logic without equality the satisfiability problem is EXPSPACE-hard. The exact complexity of $\mathsf{GF}^2 + TG$ is still an open question. The best known upper bound is the one given by Szwast and Tendera [8] – 2EXPTIME. Closing this gap can be very interesting. We think that $\mathsf{GF}^2 + TG$ is 2EXPTIME-hard and believe that we will be able to give a proof of this fact soon. Results of this paper suggest that exponentially large cliques are not responsible for a higher complexity of $\mathsf{GF}^2 + TG$. The main difficulty lies in the fact that we can demand for some elements both greater and smaller witnesses with respect to a transitive relation.

References

1. H. Andréka, J. van Benthem, I. Németi, *Modal Languages and Bounded Fragments of Predicate Logic*, ILLC Research Report, 1996. Journal ver.: *J. Philos. Logic*, 27 (1998), No. 3, 217-274.
2. J. Balcázar, J. Diaz, J. Gabarró, *Structural Complexity II*, Springer 1990.
3. A. Chandra, D. Kozen, L. Stockmeyer, *Alternation*, JACM, 28, No. 1 (1981), pages 114-133.
4. H. Ganzinger, C. Meyer, M. Veanes *The Two-Variable Guarded Fragment with Transitive Relations*, Proc. of 14-th IEEE Symp. on Logic in Computer Science, pages 24-34, 1999.
5. E. Grädel, *On the restraining power of guards*, Journal of Symbolic Logic 64:1719-1742, 1999.
6. E. Grädel, P. Kolaitis, M. Vardi, *On the decision problem for two-variable first order logic*, Bull. of Symbolic Logic, 3(1):53-96, 1997.
7. E. Grädel, I. Walukiewicz, *Guarded Fixpoint Logic*, Proc. of 14th IEEE Symp. on Logic in Computer Science, pages 45-54, 1999.
8. W. Szwast, L. Tendera *On the decision problem for the guarded fragment with transitivity*, Proc. of 16-th IEEE Symp. on Logic in Computer Science, pages 147-156, 2001.