On the restraining power of guards

Erich Grädel*
RWTH Aachen

Revised version
July 1998

Abstract

Guarded fragments of first-order logic were recently introduced by Andréka, van Benthem and Németi; they consist of relational first-order formulae whose quantifiers are appropriately relativized by atoms. These fragments are interesting because they extend in a natural way many propositional modal logics, because they have useful model-theoretic properties and especially because they are decidable classes that avoid the usual syntactic restrictions (on the arity of relation symbols, the quantifier pattern or the number of variables) of almost all other known decidable fragments of first-order logic.

Here, we investigate the computational complexity of these fragments. We prove that the satisfiability problems for the *guarded fragment* (GF) and the *loosely guarded fragment* (LGF) of first-order logic are complete for deterministic double exponential time. For the subfragments that have only a bounded number of variables or only relation symbols of bounded arity, satisfiability is EXPTIME-complete. We further establish a *tree model property* for both the guarded fragment and the loosely guarded fragment, and give a proof of the *finite model property* of the guarded fragment.

It is also shown that some natural, modest extensions of the guarded fragments are undecidable.

1 Introduction

At the beginning of this century, Hilbert formulated the *classical decision problem* for first-order logic: Find an algorithm which, given any first-order sentence, determines whether it is satisfiable. This was an essential part of his formalist programme for the foundations of mathematics and he considered it to be the central problem of mathematical logic. Early results, by Löwenheim, Behmann, Skolem, Ackermann, Bernays, Schönfinkel, Gödel and others showed that decision algorithms do indeed exist for certain fragments of first-order logic, such as the monadic class or formula classes that satisfy certain restrictions on the occurrences of quantifiers. However, in the 1930's, Church and Turing proved that the classical decision problem is algorithmically unsolvable. The classical decision problem was then transformed into a classification problem: which formula classes are decidable for satisfiability and which are not? Traditionally most attention has been given to fragments of

* Mathematische Grundlagen der Informatik, RWTH Aachen, D-52056 Aachen, graedel@informatik.rwth-aachen.de

first-order logic that are determined by the quantifier prefix and the vocabulary of relation and function symbols. With respect to such fragments the classical decision problem is completely solved, i.e. we have a complete list of all decidable prefix-vocabulary fragments of first-order logic, and for almost all of these classes, also the complexity has been precisely determined. A comprehensive account of these results is given in [8].

Modal and temporal logics provide another interesting class of logical systems with decidable satisfiablity problems. The simplest case is propositional modal logic which extends propositional logic by the modal operators $\Diamond$ and $\Box$, so that from any given formula ψ, one can also build the formulae $\Diamond\psi$ ("possibly ψ") and $\Box\psi$ ("necessarily ψ"). Semantics for modal logic is provided by *Kripke structures*, which consist of a set W of possible worlds (or states), an interpretation function that assigns to every propositional variable P the set $\pi(P) \subseteq W$ of worlds at which P is true, and a binary accessibility (or 'possibility') relation $R \subseteq W \times W$ among worlds. Intuitively $(v, w) \in R$ means that the world w is considered possible in the world v. One defines by induction the notion of a modal formula ψ to be true in the Kripke structure $\mathcal{K}$ at world v (in symbols: $\mathcal{K}, v \models \psi$). The interpretation function π immediately gives this for the case that ψ is a propositional variable: $\mathcal{K}, v \models P$ if and only if $v \in \pi(P)$. For Boolean combinations, truth values are defined as in propositional logic, and for formulae $\Diamond\psi$ and $\Box\psi$, we put

$$\mathcal{K}, v \models \Diamond\psi, \text{ if } \mathcal{K}, w \models \psi \text{ for } \textit{some} \text{ world } w \text{ such that } (v, w) \in R.$$
$$\mathcal{K}, v \models \Box\psi, \text{ if } \mathcal{K}, w \models \psi \text{ for } \textit{all} \text{ worlds } w \text{ such that } (v, w) \in R.$$

Note that $\Diamond\psi$ is equivalent to $\neg\Box\neg\psi$, so it would actually be sufficient to have only the modal operator $\Box$.

This basic modal logic can be generalized in numerous ways, for instance by considering a number of different modalities $\Box_1, \ldots, \Box_k$ with different possibility relations $R_1, \ldots, R_k$, and/or by imposing certain conditions on the possibility relations (e.g. that it should be symmetric or transitive), by introducing additional modal connectives, and so on. Particularly important are temporal logics, where the Kripke structures represent the structure of time (linear or branching) and the modalities $\Box$ and $\Diamond$ are interpreted as "from now on forever" and "eventually in the future"; besides these two basic modalities, one uses further modal connectives e.g. for "next", "until" and "since".

These logics can also be seen as fragments of first-order logic. Indeed, a Kripke structure is a first-order structure in the usual sense of Tarski and any formula ψ of the standard propositional modal logic can be translated into a first-order formula $\psi'(x)$ with one free variable, such that $\mathcal{K}, w \models \psi$ if and only if $\mathcal{K} \models \psi'(w)$. This translation takes an atomic proposition P to the atom Px, commutes with the Boolean connectives, i.e. $(\neg\psi)'(x) := \neg\psi'(x)$ and $(\psi \circ \varphi)'(x) := \psi'(x) \circ \varphi'(x)$ for $\circ \in \{\wedge, \vee, \rightarrow, \leftrightarrow\}$ and translates the modalities by quantifiers as follows:

$$(\Diamond\psi)'(x) := \exists y(Rxy \wedge \psi'(y))$$
$$(\Box\psi)'(x) := \forall y(Rxy \rightarrow \psi'(y))$$

where $\psi'(y)$ is obtained from $\psi'(x)$ by replacing all occurences of x by y and vice versa.

The *modal fragment* of first-order logic is the image of propositional modal logic under this translation. By working with the modal fragment, one can see modal logic as a part of first-order logic and investigate the former by using the techniques and results of the latter.

It has turned out that the modal fragment has very interesting and useful algorithmic and model-theoretic properties (see [4]).

On the other side, the modal fragment is a rather small part of first-order logic. First of all, it is properly contained in FO^2, relational first-order logic with only two variables. But although FO^2 is decidable and has the finite model property (see [25, 11]), it lacks the nice model-theoretic properties [1, 15] and, in particular, the robust decidability properties of modal logics [28]. Indeed modal logics (and its syntactic variants such as for instance description logics [9]) can be extended by a number of important features like path quantification, transitive closure operators, counting quantifiers, least and greatest fixed points etc. and these extensions are still decidable, algorithmically manageable and indeed of considerable practical importance. On the other side, most of the corresponding extensions of FO^2 are highly undecidable (see [12, 13]).

The embedding in FO^2 therefore does not explain the good properties of modal logic. Also algorithmically, modal logic is much simpler than FO^2: while the satisfiablility problem for modal logic is PSPACE-complete [24], the satisfiability problem for FO^2 is NEXPTIME-complete [11].

Note that in the translation of modal formulae into first-order formulae, the quantifiers are used only in a very restricted way. To explain and generalize the nice properties of modal logic, Andréka, van Benthem and Németi [1] defined the *guarded fragment* of first-order logic. They dropped the restriction to use only two variables and only monadic and binary predicates, but imposed that all quantifiers must be relativized by atomic formulae. This means that quantifiers appear only in the form

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

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

The guarded fragment GF extends the modal fragment and turns out to have interesting properties [1]:

(1) The satisfiability problem for GF is decidable.

(2) GF has the finite model property, i.e. every satisfiable formula in the guarded fragment also has a finite model.

(3) Many important model theoretic properties (like the interpolation property, the Beth definiability property, the Łos-Tarski property etc.) which hold for first-order logic and modal logic, but not, say, for the bounded-variable fragments FO^k, do hold also for the guarded fragment.

(4) The notion of equivalence under guarded formulae can be characterized by a straightforward generalization of bisimulation or, equivalently, by a variant of the Ehrenfeucht-Fraïssé game.

In a further paper, van Benthem [6] generalized the guarded fragment to the *loosely guarded fragment* (LGF) where quantifiers are guarded by conjunctions of atomic formulae of certain forms. (Details will be given in the next section.) The loosely guarded fragments has very similar properties as the guarded fragment (although it is still open whether LGF has the finite model property).

An advantage of the guarded fragments with respect to other decidable fragments of first-order logic is that the usual restrictions of the latter on the number and arity of the relation symbols or on the quantifier pattern are avoided. We believe that due to these nice syntactic, model-theoretic and algorithmic properties, the guarded fragments will be very useful in many different areas of applications.

It thus is interesting to determine the power of the guarded fragment, or to put it differently: how much do the guards restrict expressiveness? One answer to this has already been given by Andréka, van Benthem and Németi who described the expressive power of GF via an appropriate generalization of bisimulation, called *guarded bisimulation*. But expressiveness also has a complexity theoretic aspect, and in this paper, we give a complexity theoretic answer by showing that the satisfiability problems for the guarded fragment and the loosely guarded fragment are complete for deterministic double exponential time. This is rather unusual: most of the previously known decidable fragments of first-order logic are in NEXPTIME, i.e. are decidable nondeterministically in (single) exponential time (see [8]). The upper complexity bound follows from a new decidability proof, which is based on extension properties of atomic types. This proof also establishes a *tree model property* for the guarded fragments, saying that every satisfiable loosely guarded formula has a model whose tree width is bounded by the number of variables in the formula. The lower bound proof is based on the construction of a family of polynomial-size guarded formulae that force their models to contain a binary tree of double exponential depth. We also show that the subfragments of GF and LGF with a bounded number of variables or bounded arity of predicates have deterministic exponential time complexity.

In the last part of the paper, we investigate some modest extensions of the guarded fragment and show that, unfortunately, these extensions lead to undecidable systems. This also proves certain limitations of the expressive power of the guarded fragment: for instance it cannot express functionality or transitivity.

2 Guarded fragments of first-order logic

In this paper, first-order logic (FO) means relational first-order logic with equality and constants, i.e. FO-formulae may contain relation symbols of arbitrary arity and constant symbols, but no function symbols of positive arity.

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

(1) Every atomic formula belongs to GF.

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

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

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

belong to GF.

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

Obviously, GF is a fragment of first-order logic. Formulae in GF are called *guarded formulae.* Further, for $k \in \mathbb{N}$, we denote by FO^k the fragment of FO that consists of the formulae all whose variables (free or bound) are among $x_1, \ldots, x_k$; let $\mathrm{GF}^k := \mathrm{GF} \cap \mathrm{FO}^k$, the guarded fragment with only k variables.

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

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

which not guarded in the sense of Definition 2.1. However, the quantifier $\forall z$ in this formula is guarded in a weaker sense, which lead van Benthem [6] to the following generalization of GF.

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

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

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

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

In the translation of (ψ **until** φ) described above, the quantifier $\forall z$ is loosely guarded by $(x \leq z \wedge z < y)$ since z coexists with both x and y in some conjunct of the guard. On the other side, the transitivity axiom

$$\forall xyz(Exy \wedge Eyz \rightarrow Exz)$$

is not in LGF. The conjunction $Exy \wedge Eyz$ is not a proper guard of $\forall xyz$ since x and z do not coexist in any conjunct. We will see later that indeed there is no way to express transitivity in LGF.

Notation. We will use the notation $(\exists \boldsymbol{y} \,.\, \alpha)$ and $(\forall \boldsymbol{y} \,.\, \alpha)$ for relativized quantifiers, i.e. we write guarded formulae in the form $(\exists \boldsymbol{y} \,.\, \alpha)\psi(\boldsymbol{x}, \boldsymbol{y})$ and $(\forall \boldsymbol{y} \,.\, \alpha)\psi(\boldsymbol{x}, \boldsymbol{y})$. When this notation is used, then it is always understood that α is indeed a proper guard as specified by condition (3) or (3)'.

Elimination of constants. As mentioned above, we admit relation symbols of arbitrary arity and constant symbols in guarded formulae (but no function symbols of positive arity). We now show that, as far as the satisfiability problem is concerned, we can do away with the constants.

Lemma 2.3. *Every (loosely) guarded sentence ψ with constants can be translated in linear time into a (loosely) guarded sentence without constants which is satisfiable if and only if ψ is.*

Proof. Let $\boldsymbol{c} = c_1, \ldots, c_k$ be the tuple of the constant symbols appearing in ψ. For every m-ary relation symbol R of ψ, let R^* be a new $(k+m)$-ary relation symbol, and further, let Z be a k-ary relation symbol not contained in ψ. If ψ is guarded, then so is the formula

$$\psi' := (\exists \boldsymbol{c} \,.\, Z\boldsymbol{c})\psi[R\boldsymbol{x}/R^*\boldsymbol{c}\boldsymbol{x}]$$

where $\psi[R\boldsymbol{u}/R^*\boldsymbol{c}\boldsymbol{u}]$ is the formula obtained by replacing every atom of form $R\boldsymbol{u}$ of ψ by the corresponding atom $R^*\boldsymbol{c}\boldsymbol{u}$. The constants $\boldsymbol{c}$ of ψ are now variables of ψ', and since they occur in all guards of ψ' the resulting formula remains properly guarded.

It is obvious, that ψ and ψ' are satisfiable over the same domains. $\square$

3 Tree model property, finite model property and decidability

In this section, we prove the decidability and the tree model property for the guarded fragment and the loosely guarded fragment. Further, we establish the finite model property of the guarded fragment. The decidability results are due to Andréka, van Benthem and Németi [1, 5]. The proof presented here is different, although some of the ideas are adapted from the original decidability proofs and from a paper by Andréka, Hodkinson and Németi [2]. This new proof establishes the tree model property and, as we will show in the following section, leads to an essentially optimal upper complexity bound for both GF and LGF.

We start our analysis by establishing a normal form for guarded formulae, which is similar to the Scott normal form for FO^2 (see e.g. [8, 11]); this will allow us to simplify the model construction. We will then describe an abstract criterion, in terms of extension properties of atomic types, for the satisfiability of any loosely guarded sentence in normal form. It will be easy to see that this criterion is necessary for satisfiability, and we will prove that the criterion is in fact sufficient for the existence of a *tree model* for the given sentence, and, at least in the case of the guarded fragment, also for the existence of a *finite model.*

It is well-known that many modal logics have a tree model property, and this has been identified as one of the main reasons for the robust decidability properties of modal logics (see e.g. [28]). We believe that also in the case of the guarded fragments, the tree model property may become of crucial importance for decidability results of extended logics and for the design of practical (e.g. automata-based) algorithms.

3.1 A normal form

Lemma 3.1. *For every (loosely) guarded formula $\varphi(\boldsymbol{x})$ of vocabulary τ one can compute a (loosely) guarded sentence ψ of an extended vocabulary $\sigma \supseteq \tau$ such that ψ has the form*

$$\exists \boldsymbol{x} C\boldsymbol{x} \wedge \bigwedge_{i\in I} (\forall \boldsymbol{x} \,.\, \alpha_i)(\exists \boldsymbol{y} \,.\, \beta_i)\varphi_i(\boldsymbol{x},\boldsymbol{y}) \wedge \bigwedge_{j\in J} (\forall \boldsymbol{x} \,.\, \gamma_j)\psi_j(\boldsymbol{x})$$

where $\alpha_i(\boldsymbol{x})$, $\beta_i(\boldsymbol{x},\boldsymbol{y})$ and $\gamma_j(\boldsymbol{x})$ are guards[1]*, φ_i and ψ_j are quantifier-free and the following conditions hold.*

(i) $\psi \models \exists \boldsymbol{x}\varphi$.

(ii) Every model for φ can be expanded to a model of ψ.

(iii) $|\psi| = O(|\varphi|)$ *and the translation from φ to ψ is computable in polynomial time.*

Proof. We first treat the case of guarded formulae; in a second step we then describe the necessary modifications for the case of loosely guarded formulae.

We start by transforming the given formula $\varphi(\boldsymbol{x}) \in \mathrm{GF}$ in such a way that every formula $\exists \boldsymbol{y}\vartheta$, which is a proper subformula of φ, has at least one free variable (equivalently: that φ has no proper subsentences). If this is not already the case for φ, then we increment the arity of every predicate by one, choose a new variable symbol z and replace every atom $R\boldsymbol{u}$ in φ by $R\boldsymbol{u}z$. The resulting formula $\varphi'(\boldsymbol{x}, z)$ is guarded and is satisfiable if and only if $\varphi(\boldsymbol{x})$ is satisfiable.

We can thus assume that the given formula $\varphi(\boldsymbol{x})$ has no proper subsentences; it is therefore either quantifier-free or contains a subformula

$$\chi(\boldsymbol{z}) := (\exists \boldsymbol{y} \,.\, \alpha)\vartheta(\boldsymbol{y}, \boldsymbol{z})$$

with guard $\alpha(\boldsymbol{y}, \boldsymbol{z})$ and quantifier-free $\vartheta(\boldsymbol{y}, \boldsymbol{z})$. Let $\varphi[\chi/R_\chi]$ be the formula obtained by replacing $\chi(\boldsymbol{z})$ in φ by the atom $R_\chi \boldsymbol{z}$ (where R_χ is a new relation symbol). Now, the formula

$$\tilde{\varphi}(\boldsymbol{x}) := \varphi[\chi/R_\chi] \wedge (\forall \boldsymbol{z} \,.\, R_\chi \boldsymbol{z})\chi(\boldsymbol{z}) \wedge (\forall \boldsymbol{y}\boldsymbol{z} \,.\, \alpha)(\vartheta(\boldsymbol{y}, \boldsymbol{z}) \rightarrow R_\chi \boldsymbol{z})$$

is guarded and is satisfiable over the same domains as $\varphi(\boldsymbol{x})$.

This transformation step is repeated until a formula $\varphi''(\boldsymbol{x}) \wedge \eta$ is obtained where $\varphi''(\boldsymbol{x})$ is quantifier-free and η is a conjunction of guarded sentences of form $(\forall \boldsymbol{x} \,.\, \alpha)\exists \boldsymbol{y}\, \vartheta(\boldsymbol{x}, \boldsymbol{y})$ or $(\forall \boldsymbol{x} \,.\, \alpha)\vartheta(\boldsymbol{x})$ with quantifier-free ϑ. Now, choose a new relation symbol C and let

$$\psi := \exists \boldsymbol{x} C\boldsymbol{x} \wedge (\forall \boldsymbol{x} \,.\, C\boldsymbol{x})\varphi''(\boldsymbol{x}) \wedge \eta.$$

Obviously ψ is a guarded sentence with the required properties *(i)*, *(ii)* and *(iii)*.

Suppose now that the originally given formula is loosely guarded. The only problem that may arise when we eliminate a subformula $\chi(\boldsymbol{z}) = (\exists \boldsymbol{y} \,.\, \alpha)\vartheta(\boldsymbol{y}, \boldsymbol{z})$ as above concerns the conjunct $(\forall \boldsymbol{y}\boldsymbol{z} \,.\, \alpha)(\vartheta(\boldsymbol{y}, \boldsymbol{z}) \rightarrow R_\chi \boldsymbol{z})$. Although $\alpha(\boldsymbol{y}, \boldsymbol{z})$ guards the quantifier $\exists \boldsymbol{y}$, it need not be a proper guard for the quantifier $\forall \boldsymbol{y}\boldsymbol{z}$ since there may be variables z_i, z_j that do not coexist in any conjunct of α.

We take care of this problem as follows: each time when we have such a situation, we select a new relation symbol G and add to $\alpha(\boldsymbol{y}, \boldsymbol{z})$ the conjunct Gz_iz_j. At the end we thus obtain a sentence ψ which is loosely guarded, but contains a number of new atoms of form Gz_iz_j that need to be axiomatized. For each such Gz_iz_j we look where z_i, z_j came from. There are two possible cases.

[1]Here $\boldsymbol{x}$ and $\boldsymbol{y}$ stand for tuples of distinct variable symbols. The lengths of these tuples may change for different i and j.

In the first case, both variables are free in the original formula $\varphi(\boldsymbol{x})$, i.e. z_i, z_j are components of $\boldsymbol{x}$. In this case we add to ψ the conjunct $(\forall \boldsymbol{x} \,.\, C\boldsymbol{x}) G z_i z_j$.

In the second case, $\chi(\boldsymbol{z})$ originally was in the scope of a quantifier that binds at least one of z_i, z_j, i.e. $\chi(\boldsymbol{z})$ is a subformula of $(\exists \boldsymbol{u} \,.\, \beta)\gamma$ where z_i belongs to $\boldsymbol{u}$ and z_j either occurs free in this formula or also belongs to $\boldsymbol{u}$. But since β guards the quantifier $\exists \boldsymbol{u}$, it must contain a conjunct $\beta'(\boldsymbol{v})$ such that both z_i, z_j belong to $\boldsymbol{v}$. In this case we add to ψ the conjunct $(\forall \boldsymbol{v} \,.\, \beta') G z_i z_j$.

The modified sentence is loosely guarded and satisfies the required properties. $\square$

Remark. Note that the conjuncts of form $(\forall \boldsymbol{x} \,.\, \alpha)\varphi(\boldsymbol{x})$ can be seen as special cases of the conjuncts $(\forall \boldsymbol{x} \,.\, \alpha)(\exists \boldsymbol{y} \,.\, \beta)\varphi(\boldsymbol{x}, \boldsymbol{y})$. Further, for satisfiablity it doesn't really make a difference whether the *existential* quantifiers are guarded. Indeed a formula $(\forall \boldsymbol{x} \,.\, \alpha)\exists \boldsymbol{y} \varphi(\boldsymbol{x}, \boldsymbol{y})$ with unguarded existential quantifier is satisfiable if and only if the properly guarded sentence

$$(\forall \boldsymbol{x} \,.\, \alpha)\exists \boldsymbol{y}\, R\boldsymbol{x}\boldsymbol{y} \wedge (\forall \boldsymbol{x}\boldsymbol{y} \,.\, R\boldsymbol{x}\boldsymbol{y})\varphi(\boldsymbol{x}, \boldsymbol{y})$$

is satisfiable. We will therefore restrict attention to sentences of the form

$$\psi := \exists \boldsymbol{x} C\boldsymbol{x} \wedge \bigwedge_{i \in I} (\forall \boldsymbol{x} \,.\, \alpha_i)\exists \boldsymbol{y} \psi_i(\boldsymbol{x}, \boldsymbol{y})$$

with ψ_i quantifier-free.

3.2 A criterion for satisfiability

We will describe an abstract criterion, in terms of extension properties of atomic types, for the satisfiability of loosely guarded sentences in normal form. It will be easy to see that this criterion is necessary for satisfiability. In the following sections, we will show that the criterion is in fact sufficient for ψ to admit a *tree model* and, at least in the case of the guarded fragment, for the existence of a a *finite model.* We thus establish the tree model property, the finite model property and the decidability of the loosely guarded fragment.

Definition 3.2. An *(atomic) k-type* of the relational vocabulary τ is a maximal consistent set of τ-literals (i.e. atoms and negated atoms) in the variables $x_1, \dots, x_k$.[2] Given a structure $\mathfrak{A}$ and and a tuple $\boldsymbol{a} \in A^k$, we write $\mathrm{atp}_{\mathfrak{A}}(\boldsymbol{a})$ for the uniquely determined k-type t with $\mathfrak{A} \models t(\boldsymbol{a})$. A k-type t is *realized* in $\mathfrak{A}$ if $\mathfrak{A} \models t(\boldsymbol{a})$ for some tuple $\boldsymbol{a}$. The *size* of a k-type t is the number of *distinct* components in any tuple that realizes it; the size of t may be smaller than k since t may contain equalities $x_i = x_j$ for $i \neq j$. A $(k+\ell)$-type t *extends* a k-type s if $s \subseteq t$. Further, we say that t extends s *by m new elements* if t extends s and m is the difference between the sizes of t and s.

A k-type s is a *reduction* of an m-type t, if there exists a substitution $\rho : \{1, \dots, k\} \to \{1, \dots, m\}$ such that, for any realization $\boldsymbol{a} = (a_1, \dots, a_m)$ of t, the tuple $(a_{\rho(1)}, \dots, a_{\rho(k)})$ realizes s. Note that this is the case if and only if $t(x_1, \dots, x_m) \models s(x_{\rho(1)}, \dots, x_{\rho(k)})$.

Definition 3.3. Suppose that we have a guarded sentence

$$\psi := \exists \boldsymbol{x} C\boldsymbol{x} \wedge \bigwedge_{i \in I} (\forall \boldsymbol{x} \,.\, \alpha_i)\exists \boldsymbol{y} \psi_i(\boldsymbol{x}, \boldsymbol{y})$$

[2]Only atomic types will be used in this paper, so we simply call them types.

and let m be the number of distinct variables in ψ. A *witness* for the satisfiability of ψ consists of P, t_0 and $(\mathrm{ext}_i)_{i\in I}$ where

- $P = \bigcup_{k\leq m} P^{(k)}$, where each $P^{(k)}$ is a set of k-types and P is closed under reductions (i.e. if $t \in P$, then so are all its reductions that have at most m variables).
- $t_0 \in P$ is a distinguished type with $t_0(\boldsymbol{x}) \models C\boldsymbol{x}$.
- For each conjunct $(\forall x_1 \cdots x_k \,.\, \alpha_i)\exists y_1 \cdots y_\ell\, \psi_i(\boldsymbol{x}, \boldsymbol{y})$ of ψ we have an extension function $ext_i : P^{(k)} \to P^{(k+\ell)}$ such that
  - $\mathrm{ext}_i(t)$ extends t;
  - $\mathrm{ext}_i(t) \models \alpha_i(\boldsymbol{x}) \to \psi_i(\boldsymbol{x}, \boldsymbol{y})$.

Lemma 3.4. *If ψ is satisfiable, then it has a witness for satisfiability.*

Proof. Let $\mathfrak{A} \models \psi$. Choose a tuple $\boldsymbol{b}$ such that $\mathfrak{A} \models C\boldsymbol{b}$ and for each conjunct

$$(\forall x_1 \cdots x_k \,.\, \alpha_i(\boldsymbol{x}))\exists y_1 \cdots y_\ell\, \psi_i(\boldsymbol{x}, \boldsymbol{y})$$

a Skolem function $f_i : A^k \to A^\ell$ such that $\mathfrak{A} \models \alpha_i(\boldsymbol{a}) \to \psi_i(\boldsymbol{a}, f_i(\boldsymbol{a}))$ for all $\boldsymbol{a} \in A^k$. For all $k \leq m$, let $P^{(k)}$ be the set of k-types that are realized in $\mathfrak{A}$, and let t_0 be the atomic type of $\boldsymbol{b}$. Select for every type $t \in P$ a tuple $h(t)$ that realizes t in $\mathfrak{A}$. The extension function ext_i maps $s \in P^k$ to the type of the tuple $(h(s), f_i(h(s)))$, i.e. the atomic type of the tuple $h(s)$, extended by its value under the Skolem function f_i. It is easy to verify that P, t_0 and the ext_i have the required properties. □

3.3 The tree model

We now want to prove that every satisfiable sentence in the loosely guarded fragment has a tree-like model. Since, contrary to the case of propositional modal logics, the guarded fragments are not restricted to unary and binary predicates, we first have to make precise the notion of a tree-like model that is appropriate for our context.

Definition 3.5. A structure $\mathfrak{B}$ (with universe B and arbitrary vocabulary τ) is a *k-tree* if there exists a tree (i.e. an acyclic, connected graph) $T = (V, E)$ and a function $F : V \to \{X \subseteq B : |X| \leq k\}$, assigning to every node v of T a set $F(v)$ of at most k elements of $\mathfrak{B}$, such that the following two conditions hold.

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

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

Further, we say that $\mathfrak{B}$ has finite (resp. bounded) degree if for every node of T, the number of its neighbours is finite (resp. bounded by some fixed $k \in \mathbb{N}$).

Remark. Readers familiar with the notion of *tree width* in graph theory and the notion of the *Gaifman graph* of a structure will notice that a structure $\mathfrak{B}$ is a k-tree if and only the Gaifman graph of $\mathfrak{B}$ has tree width at most k. Definition 3.5 gives a natural notion of tree width for arbitrary structures. The very same definition has been introduced independently in [16].

Let $\mathfrak{B}$ be a k-tree, with T and F as in Definition 3.5. We call a tuple $\boldsymbol{b} = (b_1, \dots, b_r)$ in $\mathfrak{B}$ *local* if $\{b_1, \dots, b_r\} \subseteq F(v)$ for some node v of T. It is immediate from the definition that every tuple $\boldsymbol{b}$ that satisfies an atomic formula in $\mathfrak{B}$ is local. A simple graph-theoretic argument shows that the same is true also for every tuple $\boldsymbol{b}$ that satisfies a guard in the sense of LGF.

Lemma 3.6. *Let $\alpha(x_1, \dots, x_r)$ be a conjunction of atomic formulae such that every x_i coexists with every x_j in some conjunct of α. If $\mathfrak{B} \models \alpha(\boldsymbol{b})$, then $\boldsymbol{b}$ is local.*

Proof. For $i = 1, \dots, r$, let $V_i := \{v : b_i \in F(v)\}$. By condition *(ii)* of Definition 3.5, each V_i induces a subtree of T. Further, since $\mathfrak{B} \models \alpha(\boldsymbol{b})$ and since every pair b_i, b_j appears together in some conjunct of $\alpha(\boldsymbol{b})$, it follows from condition *(i)* of Definition 3.5 that $V_i \cap V_j \neq \varnothing$ for all i, j. A simple and well-known result in graph theory says that any collection of pairwise overlapping subtrees of a tree has a common node (see e.g. [26, p. 94]). Hence there exists a node v such that $\{b_1, \dots, b_r\} \subseteq F(v)$. $\square$

Theorem 3.7 (Tree model property). *Let ψ be a loosely guarded sentence with $m-1$ variables. If ψ is satisfiable, then there exists an m-tree with bounded degree which is a model of ψ.*

Proof. Note that the transformation into normal form introduces at most one new variable. We may hence assume that

$$\psi := \exists \boldsymbol{x} C\boldsymbol{x} \wedge \bigwedge_{i \in I} (\forall \boldsymbol{x} \,.\, \alpha_i) \exists \boldsymbol{y} \psi_i(\boldsymbol{x}, \boldsymbol{y}),$$

with m variables, and that ψ has a witness $\langle P, t_0, (\mathrm{ext}_i)_{i \in I} \rangle$.

The tree model that we construct for ψ is based on a finitely branching, directed tree T; with every node v of T we associate a set of at most m elements $F(v)$; some of them are *old elements* at v, which means that they already belong to $F(u)$ for some predecessor u of v in T. The others are the *new elements* which first come into the model at node v. The universe of the tree model $\mathfrak{B}$ to be constructed is $B := \bigcup_{v \in T} F(v)$.

We will present a simultaneous inductive construction of the tree T, the sets $F(v)$ for each node v, and the substructure $\mathfrak{F}(v) \subseteq \mathfrak{B}$ induced by $F(v)$. This will give us a partial structure, in the sense that atomic types are specified only for tuples which are entirely contained in some set $F(v)$. This will then be extended in the 'minimal' possible way to give a tree model for ψ.

For the root λ of T we set $F(\lambda) = \{\lambda_1, \dots, \lambda_r\}$ where r is the arity of C and set

$$\mathrm{atp}_{\mathfrak{B}}(\lambda_1, \dots, \lambda_r) := t_0.$$

This defines $\mathfrak{F}(\lambda)$. (Note that the elements $\lambda_1, \dots, \lambda_r$ need not be distinct.)

Suppose that we have constructed the partial structure up to node v, with $F(v) = \{v_1, \dots, v_r, v'_1, \dots, v'_s\}$ where $v_1, \dots, v_r$ are the old elements and $v'_1, \dots, v'_s$ the new elements at v. Then for the (fully determined) substructure $\mathfrak{F}(v) \subseteq \mathfrak{B}$ induced by $F(v)$ it is the case that every k-tuple $\boldsymbol{a}$ over $F(v)$ (with $k \leq m$) realizes a type in P (since P is closed under reductions).

For each $i \in I$, with extension function $\mathrm{ext}_i : P^{(k)} \to P^{(k+\ell)}$, and each k-tuple $\boldsymbol{a} = (a_1, \ldots, a_k)$ in $F(v)$, we create a successor node $w := succ_{i,\boldsymbol{a}}(v)$ of v if and only if the following conditions are satisfied:

- at least one component of the tuple $\boldsymbol{a}$ is a new element at v;
- $\mathfrak{F}(v) \models \alpha_i(\boldsymbol{a})$;
- if $s = \mathrm{atp}_{\mathfrak{F}(v)}(\boldsymbol{a})$ and $t = ext_i(s)$ then t extends s by at least one element.

If these conditions are satisfied, and t extends s by j elements, then we put $F(w) = \{a_1, \ldots, a_k, w_1, \ldots, w_j\}$ where $w_1, \ldots, w_j$ are the new elements at w, and define the atomic type of $F(w)$ so that $\mathfrak{F}(w) \models t(\boldsymbol{a}, \boldsymbol{w})$ for some tuple $\boldsymbol{w}$ in $F(w)$ (which contains all of $w_1, \ldots, w_j$).

This inductive process defines an infinite partial tree structure. We complete the structure in the minimal possible way: Recall that a tuple $\boldsymbol{b} = (b_1, \ldots, b_r)$ is *local* if $\{b_1, \ldots, b_r\} \subseteq F(v)$ for some node v of the underlying tree T. So far we have specified truth values in $\mathfrak{B}$ for all atomic statements $R\boldsymbol{b}$ where $\boldsymbol{b}$ is local. To complete the construction we impose that $\mathfrak{B} \models \neg R\boldsymbol{b}$ for all non-local tuples $\boldsymbol{b}$ and all relation symbols $R \in \tau$. This makes sure that $\mathfrak{B}$ is indeed an m-tree. Further it is clear that $\mathfrak{B}$ has bounded degree.

It remains to prove that $\mathfrak{B} \models \psi$. Clearly $\mathfrak{B} \models \exists \boldsymbol{x} C\boldsymbol{x}$. So consider a conjunct $(\forall \boldsymbol{x} \,.\, \alpha_i)\exists \boldsymbol{y} \psi_i(\boldsymbol{x}, \boldsymbol{y})$ of ψ and any tuple $\boldsymbol{a} \in B^k$. If $\mathfrak{B} \models \neg\alpha_i(\boldsymbol{a})$ nothing must be proved. If $\mathfrak{B} \models \alpha_i(\boldsymbol{a})$, then, by Lemma 3.6, $\boldsymbol{a}$ is local. Take the first node v of the tree such that $\boldsymbol{a}$ is entirely contained in $F(v)$. Then, at least one component of $\boldsymbol{a}$ is new at v. The atomic type $s = \mathrm{atp}_{\mathfrak{B}}(\boldsymbol{a})$ is determined by $\mathfrak{F}(v)$, and $s(\boldsymbol{x}) \models \alpha_i(\boldsymbol{x})$. Let $t(\boldsymbol{x}, \boldsymbol{y}) = \mathrm{ext}_i(s)$. Note that $t \models \psi_i(\boldsymbol{x}, \boldsymbol{y})$ (by the definition of a witness for ψ), so we only have to prove that t is realized in $\mathfrak{B}$. If t does not extend s by any *new* elements, then t is realized already in $\mathfrak{F}(v)$, i.e. there is a tuple $\boldsymbol{b}$ with $\{b_1, \ldots, b_\ell\} \subseteq \{a_1, \ldots, a_k\}$ such that $\mathfrak{F}(v) \models t(\boldsymbol{a}, \boldsymbol{b})$ and hence $\mathfrak{B} \models t(\boldsymbol{a}, \boldsymbol{b})$. If t extends s by $j \geq 1$ new elements, then we have created the successor node w of v in which t is realized.

We thus have proved that for i and all tuples $\boldsymbol{a}$

$$\mathfrak{B} \models \alpha_i(\boldsymbol{a}) \to \exists \boldsymbol{y} \psi_i(\boldsymbol{a}, \boldsymbol{y}).$$

Thus, $\mathfrak{B}$ is indeed a model of ψ, and we have established the tree model property for the loosely guarded fragment. □

3.4 The finite model property

The finite model property of GF follows by Herwig's Theorem, which says that every finite relational structure $\mathfrak{A}$ of finite vocabulary has a finite extension $\mathfrak{A}^+$ such that any partial isomorphism of $\mathfrak{A}$ is induced by some automorphism of $\mathfrak{A}^+$. Herwig's Theorem appears in [20, 21] and generalizes an earlier result due to Hrushowski [22].

Definition 3.8. A tuple $(a_1, \ldots, a_m)$ of elements of a structure $\mathfrak{A}$ is *live* in $\mathfrak{A}$ if $\mathfrak{A} \models \beta(a_1, \ldots, a_m)$ for some atomic formula $\beta(x_1, \ldots, x_m)$ in which all of $x_1, \ldots, x_m$ occur.

Theorem 3.9 (Herwig). *Let $\mathfrak{A}$ be any finite structure with finite relational vocabulary. Then there exists a finite extension $\mathfrak{A}^+$ of $\mathfrak{A}$ with the following properties*

(i) Every partial isomorphism of $\mathfrak{A}$ extends to an automorphism of $\mathfrak{A}^+$.

(ii) If $\boldsymbol{a}$ is a live tuple in $\mathfrak{A}^+$, then there exists an automorphism g of $\mathfrak{A}^+$ such that $g\boldsymbol{a}$ is in $\mathfrak{A}$.

We call $\mathfrak{A}^+$ a *Herwig extension* of $\mathfrak{A}$.

Theorem 3.10 (Finite model property). *Every satisfiable sentence in the guarded fragment has a finite model.*

Proof. Let ψ be a satisfiable guarded sentence. We can assume that ψ has the form

$$\exists \boldsymbol{x} C\boldsymbol{x} \wedge \bigwedge_{i\in I} (\forall \boldsymbol{x} \,.\, \alpha_i)\exists \boldsymbol{y}\psi_i(\boldsymbol{x},\boldsymbol{y}).$$

Let $\mathfrak{B} \models \psi$ be the tree model constructed in the previous section, with underlying tree T.

There are only finitely many isomorphism types of substructures $\mathfrak{F}(v) \subseteq \mathfrak{B}$ that are induced by the sets $F(v)$, v a node of T (since the sets $F(v)$ are bounded and the vocabulary of $\mathfrak{B}$ is finite). There exists a finite subtree S of T such that for every node $v \in T$ there is a node $w \in S$ such that $\mathfrak{F}(v)$ and $\mathfrak{F}(w)$ are isomorphic. Let $\mathfrak{A}$ be the the substructure induced by $A := \bigcup_{v\in S} F(v)$, and let $\mathfrak{A}^+$ be a Herwig extension of $\mathfrak{A}$. We claim that $\mathfrak{A}^+ \models \psi$.

It is clear that $\mathfrak{A}$ and hence $\mathfrak{A}^+$ satisfy $\exists \boldsymbol{x} C\boldsymbol{x}$. To see that for all i,

$$\mathfrak{A}^+ \models (\forall \boldsymbol{x} \,.\, \alpha_i)\exists \boldsymbol{y}\, \psi_i(\boldsymbol{x},\boldsymbol{y}),$$

suppose that $\mathfrak{A}^+ \models \alpha_i(\boldsymbol{a})$ for some tuple $\boldsymbol{a}$ in $\mathfrak{A}^+$. Then $\boldsymbol{a}$ is live in $\mathfrak{A}^+$, so there exists an automorphism g of $\mathfrak{A}^+$ such that the tuple $\boldsymbol{b} := g(\boldsymbol{a})$ lies in $\mathfrak{A}$. Since $\boldsymbol{b}$ is live in in $\mathfrak{A}$, $\boldsymbol{b}$ is local, i.e. it is entirely contained in some set $F(v)$, for $v \in S$. In the infinite tree model $\mathfrak{B}$, there is a successor node w of v such that $\boldsymbol{b}$ consists of the old elements of w and $\mathfrak{F}(w) \models \alpha_i(\boldsymbol{b}) \wedge \exists \boldsymbol{y}\psi_i(\boldsymbol{b},\boldsymbol{y})$. By definition $\mathfrak{A}$ contains a substructure $\mathfrak{F}(u)$ which is isomorphic to $\mathfrak{F}(w)$. Thus, there is a partial isomorphism p of $\mathfrak{A}$, mapping $\boldsymbol{b}$ to a tuple $\boldsymbol{c}$ in $\mathfrak{F}(u)$. Let f be the automorphism of $\mathfrak{A}^+$ that extends p. Then the composed automorphism fg maps $\boldsymbol{a}$ to $\boldsymbol{c}$ and since $\mathfrak{F}(u) \models \exists \boldsymbol{y}\psi_i(\boldsymbol{c},\boldsymbol{y})$ it follows that $\mathfrak{A}^+ \models \exists \boldsymbol{y}\psi_i(\boldsymbol{c},\boldsymbol{y})$ and hence $\mathfrak{A}^+ \models \exists \boldsymbol{y}\psi_i(\boldsymbol{a},\boldsymbol{y})$.

This proves that $\mathfrak{A}^+ \models \psi$. □

Remark. It is not known whether Theorem 3.10 also holds for the loosely guarded fragment. It is not difficult to formulate a stronger version of Herwig's Theorem that would imply the finite model property of LGF, but the status of this stronger version seems to be open.

4 The complexity of the guarded fragments

We prove in this section that the satisfiability problems for GF and LGF are complete for deterministic double exponential time. In fact we will treat both the upper and lower complexity bound in terms of *alternating space complexity* and use the correspondence between alternating space and deterministic time complexity classes.

4.1 Deterministic and alternating complexity classes

Let us briefly recall some basic definitions and results on deterministic and alternating complexity classes.

Definition 4.1. For any function $t : \mathbb{N} \to \mathbb{N}$, $\textsc{Dtime}(t)$ is the class of all decision problems that can be solved by a deterministic Turing machine in time $t(n)$, where n is the length of the input. We denote by $2\textsc{Exptime}$ the union, taken over all polynomials $p(n)$, of the classes $\textsc{Dtime}(2^{2^{p(n)}})$. A problem B is $2\textsc{Exptime}$-complete if it is in $2\textsc{Exptime}$ and, moreover, every problem $A \in 2\textsc{Exptime}$ can be reduced to B by a polynomial-time algorithm.

Alternating Turing machines are defined as nondeterministic Turing machines whose states are partitioned into four classes: existential, universal, accepting and rejecting states. This induces a partitioning of the configurations (instantenous descriptions) of M into existential, universal, accepting and rejecting configurations. With every alternating Turing machine M and every input x, we associate the computation tree $\mathcal{T}(M,x)$; its root is the input configuration of M on x, and the children of each node C are the immediate successor configurations of C. Thus, a computation of M on x corresponds to a path through $\mathcal{T}(M,x)$ from the root to a leaf. We can assume that existential and universal configurations have at least one successor configuration, so that the leaves of the tree are the accepting and rejecting configurations.

The acceptance condition of M on an input x is best described by a game, played by two players $\exists$ and $\forall$ on the tree $\mathcal{T}(M,x)$. The game starts at the root, and as the game proceeds, the players define a path through the tree from the root to a leaf. At existential configurations C, Player $\exists$ selects a successor configuration of C and at universal configurations, Player $\forall$ makes a similar move. Player $\exists$ wins at accepting configurations and looses at rejecting configurations. We say that M accepts x if and only if Player $\exists$ has a winning strategy for the game on $\mathcal{T}(M,x)$.

Any strategy S for $\exists$ (winning or not) corresponds to a subtree $T(S)$ of $\mathcal{T}(M,x)$ containing all nodes that are reachable if Player $\exists$ moves according to strategy S and Player $\forall$ moves arbitrarily. Thus $T(S)$ contains with each universal node also all its children, and with each existential configuration precisely one child of it. The strategy S is a winning strategy for Player $\exists$ if all leaves in the associated subtree $T(S)$ are accepting; in this case we call $T(S)$ an accepting subtree of $\mathcal{T}(M,x)$.

Definition 4.2. For functions $s, t : \mathbb{N} \to \mathbb{N}$, we denote by $\textsc{Atime}(t)$ and $\textsc{Aspace}(s)$ be the classes of those decision problems B for which there exists an alternating Turing machine M which accepts precisly the inputs $x \in B$ and such that every computation of M on an input of length n accepts or reject within $t(n)$ steps (resp. using at most $s(n)$ tape cells).

It is known (see [3, Chapter 3]) that for all functions $s(n) \geq \log n$,

$$\textsc{Aspace}(s(n)) = \bigcup_{d \in \mathbb{N}} \textsc{Dtime}(2^{ds(n)}).$$

In particular $\textsc{Aexpspace} = 2\textsc{Exptime}$ (where $\textsc{Aexpspace}$ is the union, over all polynomials $p(n)$, of the classes $\textsc{Aspace}(2^{p(n)})$).

4.2 The upper bound

The upper complexity bound for the satisfiability problem of LGF (and hence also of GF) can be extracted from the decidability proof given in the previous section.

Theorem 4.3. *The satisfiability problem for* LGF *is in* 2EXPTIME.

Proof. Given a guarded sentence with vocabulary τ and m variables, let $N(\psi)$ be the total number of atomic types of vocabulary τ in at most m variables. If τ has r predicates of maximal arity ℓ, then $N(\psi) \leq 2^{O(rm^\ell)}$. Hence, if the number of variables and the number and arities of the relation symbols are bounded by n, then $N(\psi) = 2^{2^{O(n \log n)}}$ and each atomic type can be represented by a binary string of length $2^{O(n \log n)}$.

We present an alternating exponentially space-bounded satisfiability test for loosely guarded sentences in normal form. Since alternating exponential space coincides with deterministic double exponential time, this yields the desired result.

In the description of alternating algorithms, it is common practice to describe moves by the existential player as *guesses*, and moves by the universal player as *universal choices*.

Alternating Satisfiability Test for LGF

Input: $\psi := \exists \boldsymbol{x} C\boldsymbol{x} \wedge \bigwedge_{i \in I}(\forall \boldsymbol{x} \,.\, \alpha_i)\exists \boldsymbol{y}\, \psi_i(\boldsymbol{x}, \boldsymbol{y}))$ in normal form
Compute $N(\psi)$
guess a type t
if $t \models \neg C\boldsymbol{x}$ **then reject**
else for $j = 1$ **to** $N(\psi)$ **do**
  universally choose $i \in I$ and a reduction $s(x_1, \ldots, x_k)$ of t (for appropriate k)
  guess an extension t of s
  if $t \not\models \alpha_i \to \psi_i$ **then reject**
  od
accept

It should be obvious that each computation of the satisfiability test on ψ requires no more than exponential space: the data to be stored are the current values of s, t and of the counter j up to $N(\psi)$.

Suppose that ψ is satisfiable: Then ψ has a witness $\langle P, t_0, (\mathrm{ext}_i)_{i \in I} \rangle$. By making the existential guesses as given by this witness, the satisfiability test accepts ψ.

Conversely, suppose that the test accepts ψ, i.e. that Player $\exists$ has a winning strategy on input ψ. This strategy defines the guesses of $\exists$ during the computation. We can assume that these guesses of Player $\exists$ are consistent, in the sense that if she has to guess at several times an extension for the same s and i then she always guesses the same extension t. It suffices to prove that we can extract a witness $\langle P, t_0, (\mathrm{ext}_i)_{i \in I} \rangle$ for ψ from such a winning strategy.

Let P be the set of all types that are assumed by the variables s and t in the accepting subtree associated with the winning strategy. Further, let t_0 be the type that is guessed at the beginning of the computation, and for any s and i, let $\mathrm{ext}_i(s)$ be the extension t, that is guessed by Player $\exists$ for s and i. Since the strategy leads to acceptance, $t \models \alpha_i \to \varphi_i$

for every such t, so it only remains to show that ext_i is defined for all $s \in P^{(k)}$, (for the appropriate k as given by i). But this is clear since on every computation, $N(\psi)+1$ types t are guessed; this means that at least one type appears twice, i.e. the computation loops. $\square$

4.3 Completeness

We will now prove that the upper bound that we just proved is essentially optimal: the satisfiability problem for GF (and hence for LGF) is in fact complete for deterministic double exponential time.

Theorem 4.4. *The satisfiability problem for the guarded fragment is* 2EXPTIME*-hard.*

Proof. Let A be a problem in 2EXPTIME, and let M be an *alternating* procedure, deciding A using exponential space. We can assume that M is a one-tape machine, that every computation of M accepts or rejects in at most double exponential time, and that every non-final configuration of M has precisely two successor configurations.

We first construct a family of guarded sentences φ_n such that every model $\mathfrak{A} \models \varphi_n$ encodes a binary tree T of double exponential depth. Every node of T is represented by a pair (a, b) of elements of $\mathfrak{A}$, and the depth of (a, b) (i.e. its distance from the root) is encoded by the truth-values of an n-ary predicate D on $\{a, b\}$. Note that there are 2^n n-tuples over $\{a, b\}$; if we put $a < b$ then every $\boldsymbol{x} \in \{a, b\}^n$ corresponds to a number $r(\boldsymbol{x}) < 2^n$, namely the rank of $\boldsymbol{x}$ in the lexicographical order on $\{a, b\}^n$. Thus the sequence of truth-values of D on $\{a, b\}^n$ can be viewed as the binary representation of a number $D(a, b) < 2^{2^n}$.

Further, note that we can compare the ranks of two tuples $\boldsymbol{x}, \boldsymbol{x'}$ in $\{a, b\}^n$ by the quantifier-free formulae

$$\begin{aligned} \mathit{less}(\boldsymbol{x}, \boldsymbol{x'}) &:= \bigvee_i (x_i = a \land x'_i = b \land \bigwedge_{j>i} x_i = x'_i) \\ \mathit{succ}(\boldsymbol{x}, \boldsymbol{x'}) &:= \bigvee_i \Big(\bigwedge_{j<i} (x_j = b \land x'_j = a) \land x_i = a \land x'_i = b \land \bigwedge_{j>i} x_i = x'_i \Big) \end{aligned}$$

where $\mathit{less}(\boldsymbol{x}, \boldsymbol{x'})$ says that $r(\boldsymbol{x}) < r(\boldsymbol{x'})$ (since $\boldsymbol{x'}$ has the 'larger' bit at the most significant position where $\boldsymbol{x}$ and $\boldsymbol{x'}$ differ) and $\mathit{succ}(\boldsymbol{x}, \boldsymbol{x'})$ expresses that $r(\boldsymbol{x'}) = r(\boldsymbol{x}) + 1$.

Assume for the moment, that we have available a predicate E of arity $4n + 4$ such that, for pairs (a, b) and (c, d) where (c, d) represents a successor node of (a, b), and for $\boldsymbol{x}, \boldsymbol{x'} \in \{a, b\}^n$ and $\boldsymbol{y}, \boldsymbol{y'} \in \{c, d\}^n$,

$$E\boldsymbol{x}\boldsymbol{x'}ab\boldsymbol{y}\boldsymbol{y'}cd \text{ holds if and only if } r(\boldsymbol{x}) = r(\boldsymbol{y}) \text{ and } r(\boldsymbol{x'}) = r(\boldsymbol{y'}).$$

Under this assumption, we construct φ_n of length $O(n^2)$ containing, besides E, the predicates T, R, U, D, S_1, S_2 with the following intended interpretation:

- Tab means that (a, b) represents a node of the tree;
- Rab means that (a, b) represents the root;
- $U\boldsymbol{x}ab$ is true (at least) for all $\boldsymbol{x} \in \{a, b\}^n$, where (a, b) represents a node of the tree;
- D is the n-ary relation which, on every node (a, b), encodes its depth $D(a, b)$ as described above;

- S_1abcd (resp. S_2abcd) means that (c,d) represents the first (resp. second) successor node of (a,b).

Let φ_n be the conjunction of the sentences

$$\exists\, ab\big(Rab \wedge Tab \wedge (\forall \boldsymbol{x}\,.\,U\boldsymbol{x}ab)\neg D\boldsymbol{x}\big)$$
$$(\forall ab\,.\,Tab)(a \neq b \wedge Ua\cdots aab)$$
$$(\forall \boldsymbol{x}ab\,.\,U\boldsymbol{x}ab) \bigwedge_{i<n} U\boldsymbol{x}ab[x_i/b]$$
$$(\forall ab\,.\,Tab)(\exists cd\, S_1abcd \wedge \exists cd\, S_2abcd)$$
$$\bigwedge_{i=1,2} (\forall abcd\,.\,S_iabcd)(Tcd \wedge \beta(a,b,c,d) \wedge Succ(a,b,c,d))$$

where β will be a formula axiomatizing E and

$$\begin{aligned} Succ(a,b,c,d) := \exists \boldsymbol{x}\boldsymbol{y}\Big(& E\boldsymbol{x}\boldsymbol{x}ab\boldsymbol{y}\boldsymbol{y}cd \wedge \neg D\boldsymbol{x} \wedge D\boldsymbol{y} \wedge \\ & (\forall \boldsymbol{x}'\boldsymbol{y}'\,.\,E\boldsymbol{x}\boldsymbol{x}'ab\boldsymbol{y}\boldsymbol{y}'cd)(less(\boldsymbol{x}',\boldsymbol{x}) \rightarrow (D\boldsymbol{x}' \wedge \neg D\boldsymbol{y}')) \wedge \\ & (\forall \boldsymbol{x}'\boldsymbol{y}'\,.\,E\boldsymbol{x}\boldsymbol{x}'ab\boldsymbol{y}\boldsymbol{y}'cd)(less(\boldsymbol{x},\boldsymbol{x}') \rightarrow (D\boldsymbol{x}' \leftrightarrow D\boldsymbol{y}'))\Big). \end{aligned}$$

Note that the formula $Succ(a,b,c,d)$ states that, for some $i = r(\boldsymbol{x}) = r(\boldsymbol{y})$, the i-th bit of $D(a,b)$ is 0, all bits at positions $j < i$ of $D(a,b)$ are 1, that $D(c,d)$ has a 1 at position i, $0's$ at positions $j < i$, and that the bits of $D(a,b)$ and $D(c,d)$ at positions $j > i$ coincide. This means that $D(c,d) = D(a,b) + 1$.

We therefore see that:

(a) The first conjunct of φ_n expresses that there exists a pair (a,b) with $D(a,b) = 0$; this pair represents the root of the tree.

(b) The second and the third conjunct assert that the components a,b of every node are distinct and axiomatize U by imposing that $U\boldsymbol{x}ab$ holds for every tuple $\boldsymbol{x}$ composed only of a's and b'.

(c) The fourth conjunct of φ_n enforces that for every node (a,b) of the tree there exist two (not necessarily distinct) successor nodes, via the successor predicates S_1, S_2.

(d) The last conjunct expresses that if (c,d) is a successor node of (a,b), then $D(c,d) = D(a,b) + 1$.

It remains to axiomatize E. To simplify notation, we abbreviate the concatenation $\boldsymbol{x}\boldsymbol{x}'$ by $\boldsymbol{u}$ and $\boldsymbol{y}\boldsymbol{y}'$ by $\boldsymbol{v}$, we let $\boldsymbol{a}$ and $\boldsymbol{c}$ be the tuples $(a,a,\dots,a)$ and $(c,c,\dots,c)$ of length $2n$ and R be the set of all substitutions $[u_i/a, v_i/c]$ or $[u_i/b, v_i/d]$ for $i = 0,\dots,2n-1$. Now let

$$\begin{aligned} \beta(a,b,c,d) :=& E\boldsymbol{a}ab\boldsymbol{c}cd \wedge (\forall \boldsymbol{u}\boldsymbol{v}\,.\,E\boldsymbol{u}ab\boldsymbol{v}cd) \bigwedge_{\rho\in R} E\boldsymbol{u}ab\boldsymbol{v}cd[\rho] \wedge \\ & (\forall \boldsymbol{u}\boldsymbol{v}\,.\,E\boldsymbol{u}ab\boldsymbol{v}cd) \bigwedge_{i<2n} \big((u_i = a \wedge v_i = c) \vee (u_i = b \wedge v_i = d)\big) \end{aligned}$$

There are $4n$ substitutions in R, so the length of β is $O(n \log n)$. It should be obvious that β enforces the required properties for E.

We thus have proved that every model $\mathfrak{A} \models \varphi_n$ contains a binary tree $T(\mathfrak{A})$ of depth 2^{2^n}.

The rest of the proof is a fairly straightforward application of the usual techniques for encoding computations by formulae (see [8] for a detailed exposition of the method and numerous applications).

Given any input w for M, we construct a guarded sentence $\mathrm{ACCEPT}_{M,w}$ which is satisfiable if and only if M accepts w. For $|w| = n$, $\mathrm{ACCEPT}_{M,w}$ is the conjunction of φ_n with the sentences

$$(\forall ab \,.\, Rab)\mathrm{INPUT}_{M,w}(a,b))$$
$$(\forall abcd \,.\, S_1abcd)(\mathrm{EXIST}(a,b) \to (\mathrm{NEXT}_1(a,b,c,d) \lor \mathrm{NEXT}_2(a,b,c,d))$$
$$\bigwedge_{i=1,2} (\forall abcd \,.\, S_iabcd)(\mathrm{UNIV}(a,b) \to \mathrm{NEXT}_i(a,b,c,d))$$
$$(\forall ab \,.\, Tab)\neg\mathrm{REJECT}(a,b)$$

where the formulae INPUT, NEXT_i, EXIST, UNIV and REJECT describe the behaviour of M on inputs of length n by formulae of polynomial length in n.

Here is an explicit description (for the suspicious reader): Since φ_n is a conjunct of $\mathrm{ACCEPT}_{M,w}$, every model $\mathfrak{A} \models \mathrm{ACCEPT}_{M,w}$ contains a binary tree $T(\mathfrak{A})$ of double exponential depth, whose nodes are represented by pairs (a,b). To encode on this tree an accepting subtree of the computation tree $\mathcal{T}(M,w)$, we use binary predicates Q_i (for each state q_i of M), and $(n+2)$-ary predicates Z and P_σ (for each symbol σ in the alphabet of M).

We say that $\mathfrak{A}$ imposes the configuration C (with state q_i, head position p and tape inscription y) at the node (a,b) of $T(\mathfrak{A})$ if

(1) $\mathfrak{A} \models Q_iab$;

(2) $\mathfrak{A} \models Z\boldsymbol{u}ab$ for the tuple $\boldsymbol{u} \in \{a,b\}^n$ with $r(\boldsymbol{u}) = p$;

(3) $\mathfrak{A} \models P_\sigma\boldsymbol{u}ab$ for all σ and all $\boldsymbol{u} \in \{a,b\}^n$ such that the symbol of y at position $r(\boldsymbol{u})$ is σ.

(Note that we do *not* require that $\mathfrak{A} \models \neg P_\sigma\boldsymbol{u}ab$ for those σ and $\boldsymbol{u} \in \{a,b\}^n$ for which the symbol of y at position $r(\boldsymbol{u})$ is different from σ. It thus may well be the case that $\mathfrak{A}$ imposes more than one configuration at a node (a,b). Nevertheless the construction ensures that every model of $\mathrm{ACCEPT}_{M,w}$ encodes an accepting computation tree of M on w.)

Assume that q_0 is the initial state, that $\Box$ is the blank symbol and that the input is $w = w_0 \cdots w_{n-1}$. For $i = 0, \ldots, n-1$, let $\boldsymbol{i} \in \{a,b\}^n$ be the tuple that represents i (in the sense that $r(\boldsymbol{i}) = i$). We can then express that the configuration at node (a,b) is the input configuration on input w by the formula

$$\mathrm{INPUT}_{M,x}(a,b) := Q_0ab \land Z\boldsymbol{0}ab \land \bigwedge_{i=0}^{n-1} P_{w_i}\boldsymbol{i}ab \land (\forall \boldsymbol{x} \,.\, U\boldsymbol{x}ab)(\mathit{less}(\boldsymbol{n-1}, \boldsymbol{x}) \to P_\Box\boldsymbol{x}ab).$$

The formulae EXIST, UNIV and REJECT are just disjunctions of the form $\bigvee Q_i ab$ over, respectively, the existential, the universal and the rejecting states q_i. Finally, the formulae $\mathrm{NEXT}_i(a,b,c,d)$ ensures that if C is imposed at node (a,b), then the i-the successor configuration of C is imposed at node (c,d). To express this, we take the conjunction of the formula

$$(\forall \boldsymbol{x}\boldsymbol{x}'\boldsymbol{y}\boldsymbol{y}' \,.\, E\boldsymbol{x}\boldsymbol{y}ab\boldsymbol{y}\boldsymbol{y}'cd)(Z\boldsymbol{x}'ab \wedge \boldsymbol{x} \neq \boldsymbol{x}') \rightarrow \bigwedge_{\sigma}(P_\sigma \boldsymbol{x}ab \rightarrow P_\sigma \boldsymbol{y}cd)$$

(which says that the inscription does not change at cells not currently read) with formulae taking care of the changes induced by transitions $(q_j,\sigma) \mapsto (q_k,\sigma',m)$ (imposing that when M is in state q_j and reads the symbol σ then it should write the symbol σ', shift its head according to $m \in \{-1,0,1\}$ and go into state q_k). For any such transition with $m=1$ (i.e. with head movement to the right) we have a conjunct

$$(\forall \boldsymbol{x}\boldsymbol{x}'\boldsymbol{y}\boldsymbol{y}' \,.\, E\boldsymbol{x}\boldsymbol{y}ab\boldsymbol{y}\boldsymbol{y}'cd)((Q_j ab \wedge Z\boldsymbol{x}ab \wedge P_\sigma \boldsymbol{x}ab \wedge succ(\boldsymbol{y},\boldsymbol{y}') \rightarrow (Q_k cd \wedge P_{\sigma'}\boldsymbol{y}cd \wedge Z\boldsymbol{y}'cd)).$$

Similar conjuncts are added for the transitions with $m=0$ or $m=-1$.

We have to prove that the formula $\mathrm{ACCEPT}_{M,w}$ is satisfiable if and only if the computation tree $\mathcal{T}(M,w)$ contains an accepting subtree, i.e. if and only if $w \in A$.

It should be clear every accepting subtree of $\mathcal{T}(M,w)$ provides a model for $\mathrm{ACCEPT}_{M,w}$. For the converse, suppose that $\mathfrak{A} \models \mathrm{ACCEPT}_{M,w}$ and that $T(\mathfrak{A})$ is a tree of double exponential depth that is encoded in $\mathfrak{A}$. We claim that this tree represents, via the predicates Q_i, Z and P_σ, an accepting of subtree of $\mathcal{T}(M,w)$.

Due to the conjunct

$$(\forall ab \,.\, Rab)\mathrm{INPUT}_{M,w}(a,b))$$

of $\mathrm{ACCEPT}_{M,w}$, $\mathfrak{A}$ imposes the input configuration of M on w at the root of $T(\mathfrak{A})$. Now suppose that $\mathfrak{A}$ imposes the configuration C at a node (a,b). If C is existential, then the clause

$$(\forall abcd \,.\, S_1 abcd)(\mathrm{EXIST}(a,b) \rightarrow (\mathrm{NEXT}_1(a,b,c,d) \vee \mathrm{NEXT}_2(a,b,c,d)))$$

implies that $\mathfrak{A}$ imposes one of the successor configurations at a child (c,d) of (a,b). If C is universal then the clause

$$\bigwedge_{i=1,2} (\forall abcd \,.\, S_i abcd)(\mathrm{UNIV}(a,b) \rightarrow \mathrm{NEXT}_i(a,b,c,d))$$

implies that $\mathfrak{A}$ imposes both successor configurations of C at two children (c,d) of (a,b). And finally the clause

$$(\forall ab \,.\, Tab)\neg\mathrm{REJECT}(a,b)$$

asserts that none of the configurations imposed by $\mathfrak{A}$ at a node of $T(\mathfrak{A})$ is rejecting. But this implies that there exists an accepting subtree of M on w.

We have thus proved that every problem solvable in alternating exponential space, hence every problem solvable in deterministic double exponential time, is polynomial-time reducible to the satisfiability problem for GF. $\square$

Elimination of equality. Note that equality is not really needed in this proof. Indeed we can introduce a new predicate N, replace the inequality $a \neq b$ in the second conjunct of φ_n by $Na \wedge \neg Nb$ and replace in the formulae all equalities $x_i = a$ or $x_i = c$ by Nx_i and all equalities $x_i = b$ or $x_i = d$ by $\neg Nx_i$. Thus, even the satisfiablity problem for GF without equality is complete for deterministic double exponential time.

Together with the upper bound proved before, we have thus determined the complexity of the guarded fragment and the loosely guarded fragment of first-order logic.

Corollary 4.5. *The satisfiability problems for* GF*and* LGF*(with or without equality) are complete for* 2EXPTIME.

Subfragments that are in EXPTIME. For the subfragments of GF and LGF with bounded number of variables or with relations of bounded arity, the complexity is one exponential lower. Indeed, the satisfiability problem for sentences with only k variables can easily be reduced to the case where all relation symbols have arity at most k. But if the arity of all relation symbols is bounded then the number $N(\psi)$ of relevant atomic types of a sentence ψ of length n is bounded by $2^{p(n)}$ for some polynomial p. We can hence use precisely the same arguments as above to show that the satisfiability problem for LGF with bounded number of variables or with relations of bounded arity is decidable in alternating polynomial space and hence in deterministic exponential time. For the completeness proof, the arguments are even simpler than in the case of GF. A configuration of a linearly space bounded alternating Turing machine is encoded by the truth values of n monadic predicates, and the same ideas as in the completeness proof for GF are used to represent an accepting computation tree.

Corollary 4.6. *Let $k \geq 2$ and let X be any of the following formula classes:*

(1) the set of all guarded (or loosely guarded) formulae with at most k variables;

(2) the set of all guarded (or loosely guarded) formulae with predicates of arity at most k.

Then the satisfiability problem for X is EXPTIME*-complete.*

In particular, for any finite vocabulary τ, the satisfiability problem for loosely guarded sentences of vocabuary τ is in EXPTIME.

5 Undecidable extensions of the guarded fragment

In the last section of this paper we show that some natural, modest extensions of the guarded fragment are undecidable for both satisfiability and finite satisfiability.

5.1 Recursive inseparability, conservative reduction classes and domino problems

A stronger variant of the unsolvability of the classical decision problem for first-order logic is Trakhtenbrot's *Inseparability Theorem* [27], which uses the concept of recursive inseparability.

Definition 5.1. Two disjoint sets X, Y are called *recursively inseparable* if there is no recursive set R such that $X \subseteq R$ and $R \cap Y = \varnothing$. In particular, neither X nor Y can then be decidable.

Let X be a class of formulae. We write

- $\mathit{sat}(X)$ for the the set of $\psi \in X$ that are satisfiable;
- $\mathit{fin\text{-}sat}(X)$ for the set of $\psi \in X$ that have a finite model;
- $\mathit{inf\text{-}axioms}(X)$ for $\mathit{sat}(X) - \mathit{fin\text{-}sat}(X)$, the *infinity axioms* of X;
- $\mathit{non\text{-}sat}(X)$ for the set of unsatisfiable $\psi \in X$.

Theorem 5.2 (Trakhtenbrot). *The sets* $\mathit{fin\text{-}sat}(\mathrm{FO})$, $\mathit{inf\text{-}axioms}(\mathrm{FO})$ *and* $\mathit{non\text{-}sat}(\mathrm{FO})$ *are pairwise recursively inseparable.*

Definition 5.3. Let X and Y be formula classes. A *conservative reduction* from X to Y is a recursive function $g : X \to Y$ such that for all $\psi \in X$:

(i) ψ is satisfiable if and only if $g(\psi)$ is satisfiable;

(ii) ψ is finitely satisfiable if and only if $g(\psi)$ is finitely satisfiable.

A formula class X is a *conservative reduction class* if there exists a conservative reduction from the set of all first-order formulae to X.

By Trakhtenbrot's Theorem, the sets $\mathit{non\text{-}sat}(X)$, $\mathit{fin\text{-}sat}(X)$, and $\mathit{inf\text{-}axioms}(X)$ are pairwise recursively inseparable for every conservative reduction class X. Moreover it follows that $\mathit{fin\text{-}sat}(X)$ and $\mathit{non\text{-}sat}(X)$ are hard for the r.e. sets while $\mathit{sat}(X)$ and $\mathit{inf\text{-}axioms}(X)$ are hard for the co-r.e. sets. In particular, all these problems are undecidable and a conservative reduction class necessarily has infinity axioms.

In the case that X is a recursive fragment of first-order logic, it actually suffices to find a *semi-conservative* reduction, i.e. a reduction from FO to X which maps finitely satisfiable formulae to finitely satisfiable ones and unsatisfiable formulae to unsatisfiable ones. A general recursion-theoretic argument then implies that X is a conservative reduction class (see [8, p.37f] for details).

Dominoes. Domino or tiling problems provide a simple and powerful method for proving undecidability results. They were introduced in the early 1960s by Wang as a tool to show the unsolvability of the $\forall\exists\forall$-prefix class in the pure predicate calculus. In the last thirty years they have been used to establish many undecidability results and lower complexity bounds for various systems of propositional logic, for subclasses of first-order logic and for decision problems in mathematical theories (see e.g. [8, 10, 18, 23]).

The original, 'unconstrained' version of a domino problem is given by a finite set of dominoes or tiles, each of them an oriented unit square with coloured edges; the question is whether it is possible to cover the first quadrant in the Cartesian plane by copies of these tiles, without holes and overlaps, such that adjacent dominoes have matching colours on their common edge. The set of tiles is finite, but there are infinitely many copies of each tile available; rotation of the tiles is not allowed. Variants of this problem require that certain places (e.g. the origin, the bottom row or the diagonal) are tiled by specific tiles. A slightly more convenient definition is the following.

Definition 5.4. A *domino system* $\mathcal{D}$ is a triple (D, H, V) where D is a finite set of dominoes and $H, V \subseteq D \times D$ are two binary relations. Let S be any of the spaces $\mathbb{N} \times \mathbb{N}$ or $\mathbb{Z}/s\mathbb{Z} \times \mathbb{Z}/t\mathbb{Z}$ (where $\mathbb{Z}/s\mathbb{Z}$ is $\{0, \ldots, s-1\}$ with the successor modulo s). We say that $\mathcal{D}$ *tiles* S if there exists a *tiling* $\tau : S \to D$ such that for all $(x, y) \in S$:

(i) If $\tau(x, y) = d$ and $\tau(x + 1, y) = d'$ then $(d, d') \in H$;

(ii) if $\tau(x, y) = d$ and $\tau(x, y + 1) = d'$ then $(d, d') \in V$.

In the context of conservative reductions classes we are also interested in periodic solutions of domino problems.

Definition 5.5. A domino system $\mathcal{D}$ is said to admit a *periodic tiling* of any of the spaces S above, if there is a tiling τ of S by $\mathcal{D}$ which has a horizontal and a vertical period $s, t > 0$ respectively. This means that for all points $(x, y) \in S$ we have that

$$\tau(x, y) = \tau(x + s, y) = \tau(x, y + t).$$

A periodic tiling with periods s, t may be pictured as a tiling of a torus $\mathbb{Z}/s\mathbb{Z} \times \mathbb{Z}/t\mathbb{Z}$ obtained from gluing an $s \times t$ rectangle along the edges. In particular $\mathcal{D}$ admits a periodic tiling of $\mathbb{N} \times \mathbb{N}$ iff it admits a tiling of some $\mathbb{Z}/s\mathbb{Z} \times \mathbb{Z}/t\mathbb{Z}$.

Berger [7] proved that the domino problem is undecidable. Gurevich and Koryakov [17] strengthened this to an inseparability result. For a new proof of this theorem we refer to [8, Appendix A]. The proof shows that one can effectively associate with every first-order sentence ψ a domino system $\mathcal{D}$ which tiles $\mathbb{N} \times \mathbb{N}$ periodically if ψ has a finite model, and which admits no tiling of $\mathbb{N} \times \mathbb{N}$ if ψ is unsatisfiable. Together with Gurevich's Theorem on semi-conservative reductions we obtain a simple and powerful method to establish strong forms of undecidability results.

Theorem 5.6 (Semi-conservative reductions from the domino problem). *Let X be a recursive fragment of first-order logic. X is a conservative reduction class if there exists a recursive function that associates with every domino system $\mathcal{D}$ a formula $\psi_{\mathcal{D}} \in X$ such that*

(i) If $\mathcal{D}$ admits a periodic tiling of $\mathbb{N} \times \mathbb{N}$, then $\psi_{\mathcal{D}}$ has a finite model.

(ii) If $\mathcal{D}$ does not tile $\mathbb{N} \times \mathbb{N}$, then $\psi_{\mathcal{D}}$ is not satisfiable.

This is Corollary 3.1.8 in [8, p. 91].

Grids. Two-dimensional grids form the basis of reductions from domino problems. Let $\mathcal{G}_{\mathbb{N}} = (\mathbb{N} \times \mathbb{N}, F, G)$ be the two-dimensional standard grid on $\mathbb{N} \times \mathbb{N}$, with horizontal and vertical successor relations F and G:

$$F = \big\{\big((x, y), (x + 1, y)\big) : x, y \in \mathbb{N}\big\}$$
$$G = \big\{\big((x, y), (x, y + 1)\big) : x, y \in \mathbb{N}\big\}$$

Similarly, $\mathcal{G}_m$ denotes the finite grid on $\mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/m\mathbb{Z}$.

To establish that an extension X of GF is a conservative reduction class, it suffices to characterize grids by an axiom in X, in the sense given by the following Lemma.

Lemma 5.7. *Let $X \subseteq \mathrm{FO}$ be a recursive class of sentences that is closed under conjunction with GF^2-sentences (i.e., if $\psi \in X$ and $\varphi \in \mathrm{GF}^2$ then $\psi \wedge \varphi \in X$). To establish that X is a conservative reduction class, it suffices to exhibit a sentence $\varphi \in X$, containing the binary predicates F, G (and possibly further auxiliary relations) such that*

(i) For all $r \in \mathbb{N}$ there exists a $k \in \mathbb{N}$ such that some expansion $\mathfrak{A}$ of the grid $\mathcal{G}_{k \cdot r}$ is a model of φ.

(ii) If $\mathfrak{A} \models \varphi$, then there exists a homomorphism from $\mathcal{G}_{\mathbb{N}}$ to $\mathfrak{A}$ (i.e. a function $h : \mathbb{N} \times \mathbb{N} \to A$ such that, if $h(i,j) = a$, $h(i+1,j) = b$ and $h(i,j+1) = c$, then $\mathfrak{A} \models Fab$ and $\mathfrak{A} \models Gac$).

Proof. Given a domino system $\mathcal{D} = (D, H, V)$, we construct a sentence $\psi_{\mathcal{D}} \in \mathrm{GF}^2$ whose vocabulary consists of the binary predicates F, G and monadic predicates P_d for $d \in D$ (we assume without loss of generality that the P_d are not used in φ as auxiliary predicates). The sentence $\varphi \wedge \psi_{\mathcal{D}}$ will then provide a semi-conservative reduction from the domino problem to X.

$$\begin{aligned} \psi_{\mathcal{D}} := & \bigwedge_{d \neq d'} \forall x (P_d x \to \neg P_{d'} x) \wedge \\ & (\forall xy \,.\, Fxy) \bigvee_{(d,d') \in H} (P_d x \wedge P_{d'} y) \wedge \\ & (\forall xy \,.\, Gxy) \bigvee_{(d,d') \in V} (P_d x \wedge P_{d'} y). \end{aligned}$$

Obviously $\psi_{\mathcal{D}} \in \mathrm{GF}^2$ and thus $\varphi \wedge \psi_{\mathcal{D}} \in X$. We have to show that

(1) If $\mathcal{D}$ admits a periodic tiling of $\mathbb{N} \times \mathbb{N}$ then $\varphi \wedge \psi_{\mathcal{D}}$ has a finite model.

(2) If $\mathcal{D}$ admits no tiling of $\mathbb{N} \times \mathbb{N}$ then $\varphi \wedge \psi_{\mathcal{D}}$ is unsatisfiable.

Suppose that $\tau : \mathbb{N} \times \mathbb{N} \to D$ is a periodic tiling with horizontal and vertical periods s and t. Let r be the least common multiple of s and t. There is some expansion $\mathfrak{A}$ of a grid $\mathcal{G}_{k \cdot r}$ such that $\mathfrak{A} \models \varphi$. Obviously τ also provides a correct tiling of $\mathbb{Z}/kr\mathbb{Z} \times \mathbb{Z}/kr\mathbb{Z}$. We obtain a finite model of $\varphi \wedge \psi_{\mathcal{D}}$ by further expanding $\mathfrak{A}$ with the relations

$$P_d := \{(u,v) : \tau(u,v) = d\}.$$

This proves (1). For (2), suppose that $\mathfrak{A} \models \varphi \wedge \psi_{\mathcal{D}}$. Then there is a homomorphism $h : \mathcal{G}_{\mathbb{N}} \to \mathfrak{A}$. Define a tiling $\tau : \mathbb{N} \times \mathbb{N} \to D$ by

$$\tau(i,j) := d \ \text{ for the unique } d \text{ such that } \mathfrak{A} \models P_d h(i,j).$$

Since $\mathfrak{A} \models \psi_{\mathcal{D}}$ this mapping is well-defined and provides a correct tiling.

By Theorem 5.6 it follows that X is a conservative reduction class. □

5.2 The guarded fragment with functionality statements or counting

By a functionality statement we mean an assertion functional$[D]$, saying that the binary relation D is the graph of a partial function, i.e. a statement of the form $\forall x \exists^{\leq 1} y Dxy$.

Theorem 5.8. GF *with functionality statements is a conservative reduction class. In fact, this is already the case for the class of formulae* $\psi \wedge \text{functional}[D]$*, where* ψ *is in* GF^3 *and does not contain equality.*

Proof. It suffices to show that we can characterize grids in the sense of Lemma 5.7. The idea is to expand any finite grids $\mathcal{G}_m = (\mathbb{Z}/m\mathbb{Z} \times \mathbb{Z}/m\mathbb{Z}, F, G)$ by the relations

$$\begin{aligned} FG &= \{(x,y,z) : (x,y) \in F \wedge (y,z) \in G\} \\ GF &= \{(x,y,z) : (x,y) \in G \wedge (y,z) \in F\} \\ D &= \{(x,z) : (x,y,z) \in F \text{ for some } y\} = \{(x,z) : (x,y,z) \in G \text{ for some } y\}. \end{aligned}$$

Let φ be the conjunction of the formulae

$$\begin{gathered} \exists x N x \\ (\forall\ x.\, Nx)(\exists y Fxy \wedge \exists y Gxy) \\ (\forall xy\,.\, Fxy)(\exists z Fyz \wedge \exists z FGxyz) \\ (\forall xy\,.\, Gxy)(\exists z Gyz \wedge \exists z GFxyz) \\ (\forall xyz\,.\, FGxyz)(Gyz \wedge Dxz) \\ (\forall xyz\,.\, GFxyz)(Fyz \wedge Dxz) \end{gathered}$$

We claim that the formula $\varphi \wedge \text{functional}[D]$ provides the desired characterization. Indeed, for every finite finite grid $\mathcal{G}_m$, the expansions $(\mathcal{G}_m, FG, GF, D)$ described above is a model of $\varphi \wedge \text{functional}[D]$. On the other side, let $\mathfrak{A}$ be an arbitrary model of $\varphi \wedge \text{functional}[D]$. By the first conjunct of φ, there exists an element a such that $\mathfrak{A} \models Na$. The further conjuncts of φ imply that there exist unary functions f, g on $\mathfrak{A}$ such that, for every element b in the closure of $\{a\}$ under f and g, the following hold:

$$\mathfrak{A} \models F(b, fb) \wedge G(b, gb) \wedge FG(b, fb, gfb) \wedge GF(b, gb, fgb) \wedge D(b, fgb) \wedge D(b, gfb).$$

Denote by B the closure of $\{a\}$ under f and g. Since $\mathfrak{A} \models \text{functional}[D]$, it follows that $gfb = fgb$ for all $b \in B$, i.e. (B, f, g) is isomorphic to the standard grid on $\mathbb{N} \times \mathbb{N}$. The desired homomorphism $h : \mathcal{G}_{\mathbb{N}} = (\mathbb{N} \times \mathbb{N}, F, G) \to \mathfrak{A}$ is now defined by $h(i,j) := f^i g^j a$. $\square$

Corollary 5.9. GF^3 *with counting quantifiers is a conservative reduction class.*

This last result was first observed by Maarten Marx (private communication). Note that this result is optimal with respect to the number of variables since even the unguarded variant of first-order logic with two variables and arbitrary counting quantifiers was proved to be decidable (see [14]).

5.3 The guarded fragment with transitivity statements

Note that the obvious formalization of the transitivity of a binary relation E, namely

$$\forall xyz(Exy \wedge Eyz \rightarrow Exz)$$

is neither guarded nor loosely guarded. But of course, there still could be a different way to express transitivity in one of these fragments.

We show here that this is not the case. In fact it suffices to add simple statements of the form transitive$[E_0, \dots, E_m]$ (saying that the binary relations $E_0, \dots, E_m$ are transitive) to GF^3 to obtain a conservative reduction class.

Theorem 5.10. *The class of formulae* $\psi \wedge \text{transitive}[E_0, \dots, E_m]$ *with* $\psi \in \mathrm{GF}^3$ *is a conservative reduction class.*

Proof. We show that the functionality of some binary relation can be enforced by transitivity statements. More precisely, we will construct a formula $\psi(E_0, E_1, F) \in GF^2$ (which will contain additional predicates besides E_0, E_1 and F) such that $\psi \wedge \text{transitive}[E_0, E_1] \models$ functional$[F]$ and, on the other side, every structure (A, F) with the property that F is the graph of a bipartite partial function can be expanded to a model for $\psi \wedge \text{transitive}[E_0, E_1]$.

To build ψ, we start with the conjuncts

$$\begin{gathered}
\exists xy Fxy \\
(\forall xy \,.\, Fxy)(F_0xy \leftrightarrow \neg F_1xy) \\
(\forall xy \,.\, F_0xy)(Fxy \wedge \forall y \neg F_1xy \wedge \forall x \neg F_0yx) \\
(\forall xy \,.\, F_1xy)(Fxy \wedge \forall y \neg F_0xy \wedge \forall x \neg F_1yx)
\end{gathered}$$

(Note that here $\forall y \neg F_ixy$ is just a shorthand for $(\forall y . Fxy)$*false* which is properly guarded.) These formulae enforce that F is the disjoint union of F_0 and F_1, that F_0 and F_1 alternate on every F-chain, and that no point has at the same time outgoing F_0-edges and outgoing F_1-edges

Further we add, for $i = 0, 1$, the conjuncts

$$\begin{gathered}
(\forall xy \,.\, F_ixy)(E_ixy \wedge E_iyx \wedge E_ixx \wedge E_iyy) \\
(\forall xy \,.\, E_ixy)(x = y \vee F_ixy \vee F_iyx)
\end{gathered}$$

saying that E_i is the reflexive, symmetric closure of F_i.

This completes the construction of ψ. It should be obvious that every structure (A, F) where F is functional and bipartite (i.e. has no odd cycles) can be expanded to a model of $\psi \wedge \text{transitive}[E_0, E_1]$. Conversely, suppose that

$$(A, F, F_0, F_1, E_0, E_1) \models \psi \wedge \text{transitive}[E_0, E_1].$$

Now suppose that F is not functional; then either F_0 or F_1 are not functional since no point has outgoing edges of both kinds. So suppose that F_0ab and F_0ac hold for $b \neq c$.

Clearly b and c are also distinct from a and neither F_0bc nor F_0cb holds (otherwise the structure would contain two successive F_0-edges). Since E_0 is the reflexive and symmetric closure of F_0 the atoms E_0ba and E_0ac are true, but *not* the atom E_0bc. But this contradicts the transitivity of E_0. Thus F has to be functional.

Now one can proceed in the same way as in the proof of Theorem 5.8. □

References

[1] H. ANDRÉKA, J. VAN BENTHEM, AND I. NÉMETI, *Modal Languages and Bounded Fragments of Predicate Logic*, ILLC Research Report ML-96-03 (1996), 59 pages.

[2] H. ANDRÉKA, I. HODKINSON, AND I. NÉMETI, *Finite algebras of relations are representable on finite sets*, preprint (1996), 28 pages.

[3] J. BALCÁZAR, J. DÍAZ, AND J. GABARRÓ, *Structural Complexity II*, Springer 1990.

[4] J. VAN BENTHEM, *Modal Logic an Classical Logic*, Bibliopolis, Napoli 1983.

[5] J. VAN BENTHEM, *Exploring Logical Dynamics*, CSLI-Publications, Stanford 1996.

[6] J. VAN BENTHEM, *Dynamic bits and pieces*, ILLC Research Report (1997).

[7] R. BERGER, *The undecidability of the domino problem*, Mem. AMS, 66 (1966).

[8] E. BÖRGER, E. GRÄDEL, AND Y. GUREVICH, *The Classical Decision Problem*, Springer, 1997.

[9] F. DONNINI, M. LENZERINI, D. NARDI, AND A. SCHAERF, *Reasoning in description logics*, in G. Brewka (Ed.), "Principles of Knowledge Representation", Studies in Logic, Language and Information, CSLI Publications (1996), 193–238.

[10] E. GRÄDEL, *Dominoes and the complexity of subclasses of logical theories*, Annals of Pure and Applied Logic, 43 (1989), pp. 1–30.

[11] E. GRÄDEL, P. KOLAITIS, AND M. VARDI, *On the decision problem for two-variable first-order logic*, Bulletin of Symbolic Logic 3 (1997), 53–69.

[12] E. GRÄDEL AND M. OTTO, *On logcs with two variables*, to appear in Theoretical Computer Science.

[13] E. GRÄDEL, M. OTTO, AND E. ROSEN, *Undecidability results on Two-variable logics*, Proceedings of 14th Symposium on Theoretical Aspects of Computer Science STACS'97, Lecture Notes in Computer Science Nr. 1200, Springer (1997), 249–260. A revised and expanded version will appear in Archive of Mathematical Logic.

[14] ——, *Two-variable logic with counting is decidable*, Proceedings of 12th IEEE Symposium on Logic in Computer Science LICS '97, Warsaw (1997).

[15] E. GRÄDEL AND E. ROSEN, *On preservation theorems for two-variable logic*, submitted for publication.

[16] M. GROHE AND J. MARIÑO, *Definability and descriptive complexity on databases of bounded tree-width*, submitted for publication.

[17] Y. GUREVICH AND I. KORYAKOV, *Remarks on Berger's paper on the domino problem*, Siberian Math. Journal, 13 (1972), pp. 319–321.

[18] D. HAREL, *Recurring dominoes: Making the highly undecidable highly understandable*, Annals of Discrete Mathematics, 24 (1985), pp. 51–72.

[19] ——, *Effective transformations on infinite trees, with applications to high undecidability, dominoes and fairness*, J. ACM, 33 (1986), pp. 224–248.

[20] B. HERWIG, *Extending partial isomorphisms on finite structures*, Combinatorica 15 (1995), 365–371.

[21] B. HERWIG, *Extending partial isomorphisms for the small index property of many ω-categorical structures*, Israel J. Math., submitted.

[22] E HRUSHOWSKI, *Extending partial isomorphisms of graphs*, Combinatorica 12 (1992), 411–416.

[23] A. KAHR, E. MOORE, AND H. WANG, *Entscheidungsproblem reduced to the $\forall\exists\forall$ case*, Proc. Nat. Acad. Sci. U.S.A., 48 (1962), pp. 365–377.

[24] R. LADNER, *The computational complexity of provability in systems of modal logic*, SIAM J. Computing 6 (1977), pp. 467–480.

[25] M. MORTIMER, *On languages with two variables*, Zeitschr. f. math. Logik u. Grundlagen d. Math., 21 (1975), pp. 135–140.

[26] B. REED, *Tree width and tangles: A new connectivity measure and some applications*, in: R.A. Bailey (Ed.), *Surveys in Combinatorics*, Cambridge University Press (1997), 87–162.

[27] B. TRAKHTENBROT, *On recursive separability*, Dokl. Akad. Nauk SSSR, 88 (1953), pp. 953–955. In Russian.

[28] M. VARDI, *Why is modal logic so robustly decidable*, DIMACS Series in Discrete Mathematics and Theoretical Computer Science 31, American Math. Society, 1997, 149–184.