Conservative Extensions in Guarded and Two-Variable Fragments*†

Jean Christoph Jung[1], Carsten Lutz[1], Mauricio Martel[1], Thomas Schneider[1], and Frank Wolter[2]

1 University of Bremen, Bremen, Germany
jeanjung@informatik.uni-bremen.de
2 University of Bremen, Bremen, Germany
clu@informatik.uni-bremen.de
3 University of Bremen, Bremen, Germany
martel@informatik.uni-bremen.de
4 University of Bremen, Bremen, Germany
ts@informatik.uni-bremen.de
5 University of Liverpool, Liverpool, UK
wolter@liverpool.ac.uk

Abstract

We investigate the decidability and computational complexity of (deductive) conservative extensions in fragments of first-order logic (FO), with a focus on the two-variable fragment FO^2 and the guarded fragment GF. We prove that conservative extensions are undecidable in any FO fragment that contains FO^2 or GF (even the three-variable fragment thereof), and that they are decidable and 2ExpTime-complete in the intersection GF^2 of FO^2 and GF.

1998 ACM Subject Classification F.4.1 [Mathematical Logic] Computational Logic

Keywords and phrases Conservative Extensions, Decidable Fragments of First-Order Logic, Computational Complexity

Digital Object Identifier 10.4230/LIPIcs.ICALP.2017.108

1 Introduction

Conservative extensions are a fundamental notion in logic. In mathematical logic, they provide an important tool for relating logical theories, such as theories of arithmetic and theories that emerge in set theory [35, 31]. In computer science, they come up in diverse areas such as software specification [12], higher order theorem proving [15], and ontologies [24]. In these applications, it would be very useful to decide, given two sentences φ_1 and φ_2, whether $\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1. As expected, this problem is undecidable in first-order logic (FO). In contrast, it has been observed in recent years that conservative extensions are decidable in many modal and description logics [13, 26, 27, 7]. This observation is particularly interesting from the viewpoint of ontologies, where conservative extensions have many natural applications including modularity and reuse, refinement, versioning, and forgetting [9, 24].

Regarding decidability, conservative extensions thus seem to behave similarly to the classical satisfiability problem, which is also undecidable in FO while it is decidable for modal

* A full version of the paper is available at https://arxiv.org/abs/1705.10115.
† Funded by DFG grant LU 1417/2.

44th International Colloquium on Automata, Languages, and Programming (ICALP 2017).
Editors: Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl;
Article No. 108; pp. 108:1–108:14
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

and description logics. In the case of satisfiability, the aim to understand the deeper reasons for this discrepancy and to push the limits of decidability to more expressive fragments of FO has sparked a long line of research that identified prominent decidable FO fragments such as the two-variable fragment FO^2 [34, 29], its extension with counting quantifiers C^2 [19], the guarded fragment GF [1], and the guarded negation fragment GNF [4], see also [6, 16, 33, 23] and references therein. These fragments have sometimes been used as a replacement for the modal and description logics that they generalize, and in particular the guarded fragment has been proposed as an ontology language [3]. Motivated by this situation, the aim of the current paper is to study the following two questions:

1. Are conservative extensions decidable in relevant fragments of FO such as FO^2, C^2, GF, and GNF?
2. What are the deeper reasons for decidability of conservative extensions in modal and description logics and how far can the limits of decidability be pushed?

To be more precise, we concentrate on *deductive* conservative extensions, that is, $\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1 if for every sentence ψ formulated in the signature of φ_1, $\varphi_1 \wedge \varphi_2 \models \psi$ implies $\varphi_1 \models \psi$. There is also a *model-theoretic* notion of conservative extension which says that $\varphi_1 \wedge \varphi_2$ is a conservative extension of φ_1 if every model of φ_1 can be extended to a model of φ_2 by interpreting the additional symbols in φ_2. Model-theoretic conservative extensions imply deductive conservative extensions, but the converse fails unless one works with a very expressive logic such as second-order logic [24]. In fact, model-theoretic conservative extensions are undecidable even for some very inexpressive description logics that include neither negation nor disjunction [25]. Deductive conservative extensions, as studied in this paper, are closely related to other important notions in logic, such as uniform interpolation [30, 36, 5]. For example, in logics that enjoy Craig interpolation, a decision procedure for conservative extensions can also be used to decide whether a given sentence φ_2 is a uniform interpolant of a given sentence φ_1 regarding the symbols used in φ_2.

Instead of concentrating only on conservative extensions, we also consider two related reasoning problems that we call Σ-entailment and Σ-inseparability, where Σ denotes a signature. The definitions are as follows: a sentence φ_1 Σ-entails a sentence φ_2 if for every sentence ψ formulated in Σ, $\varphi_2 \models \psi$ implies $\varphi_1 \models \psi$. This can be viewed as a more relaxed notion of conservative extension where it is not required that one sentence actually extends the other as in the conjunction $\varphi_1 \wedge \varphi_2$ used in the definition of conservative extensions. Two sentences φ_1, φ_2 are Σ-inseparable if they Σ-entail each other. We generally prove lower bounds for conservative extensions and upper bounds for Σ-entailment, in this way obtaining the same decidability and complexity results for all three problems.

Our first main result is that conservative extensions are undecidable in FO^2 and (the three-variable fragment of) GF, and in fact in all fragments of FO that contain at least one of the two; note that the latter is not immediate because the separating sentence ψ in the definition of conservative extensions ranges over all sentences from the considered fragment, giving greater separating power when we move to a larger fragment. The proofs are by reductions from the halting problem for two-register machines and a tiling problem, respectively. We note that undecidability of conservative extensions also implies that there is no extension of the logic in question in which consequence is decidable and that has effective uniform interpolation (in the sense that uniform interpolants exist and are computable). We then show as our second main result that, in the two-variable guarded fragment GF^2, Σ-entailment is decidable in 2ExpTime. Regarding the satisfiability problem, GF^2 behaves fairly similarly to modal and description logics. It is thus suprising that deciding Σ-entailment (and conservative extensions) in GF^2 turns out to be much more challenging than in most modal and description logics. There, the main approach to proving decidability of Σ-entailment is to first establish a suitable model-theoretic characterization based on bisimulations which is then used as a foundation for a decision procedure based on tree automata [27, 7]. In GF^2, an analogous characterization in terms of appropriate guarded bisimulation fails. Instead, one has to demand the existence of *k-bounded* (guarded) bisimulations, *for all* k, and while tree automata can easily handle bisimulations, it is not clear how they can deal with such an infinite family of bounded bisimulations. We solve this problem by a very careful analysis of the situation and by providing another characterization that can be viewed as being 'half way' between a model-theoretic characterization and an automata-encoding of Σ-entailment.

We also observe that a 2ExpTime lower bound from [13] for conservative extensions in description logics can be adapted to GF^2, and consequently our upper bound is tight. It is known that GF^2 enjoys Craig interpolation and thus our results are also relevant to deciding uniform interpolants and to a stronger version of conservative extensions in which the separating sentence ψ can also use 'helper symbols' that occur neither in φ_1 nor in φ_2.

2 Preliminaries

We introduce the fragments of classical first-order logic (FO) that are relevant for this paper. We generally admit equality and disallow function symbols and constants. With *FO^2*, we denote the *two-variable fragment of FO*, obtained by fixing two variables x and y and disallowing the use of other variables [34, 29]. In FO^2 and fragments thereof, we generally admit only predicates of arity one and two, which is without loss of generality [17]. In the *guarded fragment of FO*, denoted *GF*, quantification is restricted to the pattern

$$\forall \mathbf{y}(\alpha(\mathbf{x},\mathbf{y}) \to \varphi(\mathbf{x},\mathbf{y})) \quad \exists \mathbf{y}(\alpha(\mathbf{x},\mathbf{y}) \wedge \varphi(\mathbf{x},\mathbf{y}))$$

where $\varphi(\mathbf{x},\mathbf{y})$ is a GF formula with free variables among $\mathbf{x},\mathbf{y}$ and $\alpha(\mathbf{x},\mathbf{y})$ is an atomic formula $R\mathbf{xy}$ or an equality $x = y$ that in either case contains all variables in $\mathbf{x},\mathbf{y}$ [1, 16]. The formula α is called the *guard* of the quantifier. The k-variable fragment of GF, defined in the expected way, is denoted *GF^k*. Apart from the logics introduced so far, in informal contexts we shall also mention several related description logics. Exact definitions are omitted, we refer the reader to [2].

A *signature* Σ is a finite set of predicates. We use GF(Σ) to denote the set of all GF-sentences that use only predicates from Σ (and possibly equality), and likewise for $GF^2(\Sigma)$ and other fragments. We use $\mathsf{sig}(\varphi)$ to denote the set of predicates that occur in the FO formula φ. Note that we consider equality to be a logical symbol, rather than a predicate, and it is thus never part of a signature. We write $\varphi_1 \models \varphi_2$ if φ_2 is a logical consequence of φ_1. The next definition introduces the central notions studied in this paper.

▶ **Definition 1.** Let F be a fragment of FO, φ_1, φ_2 F-sentences and Σ a signature. Then
1. φ_1 Σ-*entails* φ_2, written $\varphi_1 \models_\Sigma \varphi_2$, if for all $F(\Sigma)$-sentences ψ, $\varphi_2 \models \psi$ implies $\varphi_1 \models \psi$;
2. φ_1 and φ_2 are Σ-*inseparable* if φ_1 Σ-entails φ_2 and vice versa;
3. $\varphi_1 \wedge \varphi_2$ is a *conservative extension* of φ_1 if φ_1 $\mathsf{sig}(\varphi_1)$-entails $\varphi_1 \wedge \varphi_2$.

Note that Σ-entailment could equivalently be defined as follows when F is closed under negation: φ_1 Σ-*entails* φ_2 if for all $F(\Sigma)$-sentences ψ, satisfiability of $\varphi_1 \wedge \psi$ implies satisfiability of $\varphi_2 \wedge \psi$. If φ_1 does not Σ-entail φ_2, there is thus an $F(\Sigma)$-sentence ψ such that $\varphi_1 \wedge \psi$ is satisfiable while $\varphi_2 \wedge \psi$ is not. We refer to such ψ as a *witness sentence* for non-Σ-entailment.

▶ **Example 2.** (1) Σ-entailment is a weakening of logical consequence, that is, $\varphi_1 \models \varphi_2$ implies $\varphi_1 \models_\Sigma \varphi_2$ for any Σ. The converse is true when $\mathsf{sig}(\varphi_2) \subseteq \Sigma$.

(2) Consider the GF^2 sentences $\varphi_1 = \forall x \exists y Rxy$ and $\varphi_2 = \forall x(\exists y(Rxy \wedge Ay) \wedge \exists y(Rxy \wedge \neg Ay))$ and let $\Sigma = \{R\}$. Then $\psi = \forall xy(Rxy \rightarrow x = y)$ is a witness for $\varphi_1 \not\models_\Sigma \varphi_2$. If φ_1 is replaced by $\varphi_1' = \forall x \exists y(Rxy \wedge x \neq y)$ we obtain $\varphi_1' \models_\Sigma \varphi_2$ since GF^2 cannot count the number of R-successors.

It is important to note that different fragments F of FO give rise to different notions of Σ-entailment, Σ-inseparability and conservative extensions. For example, if φ_1 and φ_2 belong to GF^2, then they also belong to GF and to FO^2, but it might make a difference whether witness sentences range over all GF^2-sentences, over all GF-sentences, or over all FO^2-sentences. If we want to emphasize the fragment F in which witness sentences are formulated, we speak of $F(\Sigma)$-entailment instead of Σ-entailment and write $\varphi_1 \models_{F(\Sigma)} \varphi_2$, and likewise for $F(\Sigma)$-inseparability and F-conservative extensions.

▶ **Example 3.** Let φ_1', φ_2, and $\Sigma = \{R\}$ be from Example 2 (2). Then φ_1' $\mathrm{GF}^2(\Sigma)$-entails φ_2 but φ_1' does not $\mathrm{FO}(\Sigma)$-entail φ_2; a witness is given by $\forall xy_1y_2((Rxy_1 \wedge Rxy_2) \rightarrow y_1 = y_2)$.

Note that conservative extensions and Σ-inseparability reduce in polynomial time to Σ-entailment (with two calls to Σ-entailment required in the case of Σ-inseparability). Moreover, conservative extensions reduce in polynomial time to Σ-inseparability. We thus state our upper bounds in terms of Σ-entailment and lower bounds in terms of conservative extensions.

There is a natural variation of each of the three notions in Definition 1 obtained by allowing to use additional 'helper predicates' in witness sentences. For a fragment F of FO, F-sentences φ_1, φ_2, and a signature Σ, we say that φ_1 *strongly Σ-entails* φ_2 if φ_1 Σ'-entails φ_2 for any Σ' with $\Sigma' \cap \mathsf{sig}(\varphi_2) \subseteq \Sigma$. Strong Σ-inseparability and strong conservative extensions are defined accordingly. Strong Σ-entailment implies Σ-entailment, but the converse may fail.

▶ **Example 4.** $\mathrm{GF}(\Sigma)$-entailment does not imply strong $\mathrm{GF}(\Sigma)$-entailment. Let φ_1 state that the binary predicate R is irreflexive and symmetric and let φ_2 be the conjunction of φ_1 and $\forall x(Ax \rightarrow \forall y(Rxy \rightarrow \neg Ay)) \wedge \forall x(\neg Ax \rightarrow \forall y(Rxy \rightarrow Ay))$. Thus, an $\{R\}$-structure satisfying φ_1 can be extended to a model of φ_2 if it contains no R-cycles of odd length. Now observe that any satisfiable $\mathrm{GF}(\{R\})$ sentence is satisfiable in a forest $\{R\}$-structure (see Section 4 for a precise definition). Hence, if a $\mathrm{GF}(\{R\})$-sentence is satisfiable in an irreflexive and symmetric structure then it is satisfiable in a structure without odd cycles and so φ_1 $\mathrm{GF}(\{R\})$-entails φ_2. In contrast, for the fresh ternary predicate Q and $\psi = \exists x_1x_2x_3(Qx_1x_2x_3 \wedge Rx_1x_2 \wedge Rx_2x_3 \wedge Rx_3x_1)$ we have $\varphi_2 \models \neg\psi$ but $\varphi_1 \not\models \neg\psi$ and so ψ witnesses that φ_1 does not $\mathrm{GF}(\{R, Q\})$-entail φ_2.

The example above is inspired by proofs that GF does not enjoy Craig interpolation [21, 11]. This is not accidental, as we explain next. Recall that a fragment F of FO *has Craig interpolation* if for all F-sentences ψ_1, ψ_2 with $\psi_1 \models \psi_2$ there exists an F-sentence ψ (called an *F-interpolant for ψ_1, ψ_2*) such that $\psi_1 \models \psi \models \psi_2$ and $\mathsf{sig}(\psi) \subseteq \mathsf{sig}(\psi_1) \cap \mathsf{sig}(\psi_2)$. F *has uniform interpolation* if one can always choose an F-interpolant that does not depend on ψ_2, but only on ψ_1 and $\mathsf{sig}(\psi_1) \cap \mathsf{sig}(\psi_2)$. Thus, given ψ_1, ψ and Σ with $\psi_1 \models \psi$ and $\mathsf{sig}(\psi) \subseteq \Sigma$, then ψ is a *uniform $F(\Sigma)$-interpolant of* ψ_1 iff ψ strongly $F(\Sigma)$-entails ψ_1. Both Craig interpolation and uniform interpolation have been investigated extensively, for example for intuitionistic logic [30], modal logics [36, 10, 28], guarded fragments [11], and description logics [27]. The following observation summarizes the connection between (strong) Σ-entailment and interpolation.

▶ **Theorem 5.** *Let F be a fragment of FO that enjoys Craig interpolation. Then $F(\Sigma)$-entailment implies strong $F(\Sigma)$-entailment. In particular, if $\varphi_2 \models \varphi_1$ and $\mathsf{sig}(\varphi_1) \subseteq \Sigma$, then φ_1 is a uniform $F(\Sigma)$-interpolant of φ_2 iff φ_1 $F(\Sigma)$-entails φ_2.*

Proof. Assume φ_1 does not strongly $F(\Sigma)$-entail φ_2. Then there is an F-sentence ψ with $\mathsf{sig}(\psi) \cap \mathsf{sig}(\varphi_2) \subseteq \Sigma$ such that $\varphi_2 \models \psi$ and $\varphi_1 \wedge \neg\psi$ is satisfiable. Let χ be an interpolant for φ_2 and ψ in F. Then $\neg\chi$ witnesses that φ_1 does not $F(\Sigma)$-entail φ_2. ◀

The *uniform interpolant recognition problem for* F is the problem to decide whether a sentence ψ is a uniform $F(\Sigma)$-interpolant of a sentence ψ'. It follows from Theorem 5 that in any fragment F of FO that enjoys Craig interpolation, this problem reduces in polynomial time to Σ-inseparability in F and that, conversely, conservative extension in F reduces in polynomial time to the uniform interpolant recognition problem in F. Neither GF nor FO^2 nor description logics with role inclusions enjoy Craig interpolation [21, 8, 24], but GF^2 does [21]. Thus, our decidability and complexity results for Σ-entailment in GF^2 also apply to strong Σ-entailment and the uniform interpolant recognition problem.

3 Undecidability

We prove that conservative extensions are undecidable in GF^3 and in FO^2, and consequently so are Σ-entailment and Σ-inseparability (as well as strong Σ-entailment and the uniform interpolant recognition problem). These results hold already without equality and in fact apply to all fragments of FO that contain at least one of GF^3 and FO^2 such as the guarded negation fragment [4] and the two-variable fragment with counting quantifiers [19].

We start with the case of GF^3, using a reduction from the halting problem of two-register machines. A (deterministic) *two-register machine (2RM)* is a pair $M = (Q, P)$ with $Q = q_0, \dots, q_\ell$ a set of *states* and $P = I_0, \dots, I_{\ell-1}$ a sequence of *instructions*. By definition, q_0 is the *initial state*, and q_ℓ the *halting state*. For all $i < \ell$,

- either $I_i = +(p, q_j)$ is an *incrementation instruction* with $p \in \{0, 1\}$ a register and q_j the subsequent state;
- or $I_i = -(p, q_j, q_k)$ is a *decrementation instruction* with $p \in \{0, 1\}$ a register, q_j the subsequent state if register p contains 0, and q_k the subsequent state otherwise.

A *configuration* of M is a triple (q, m, n), with q the current state and $m, n \in \mathbb{N}$ the register contents. We write $(q_i, n_1, n_2) \Rightarrow_M (q_j, m_1, m_2)$ if one of the following holds:

- $I_i = +(p, q_j)$, $m_p = n_p + 1$, and $m_{1-p} = n_{1-p}$;
- $I_i = -(p, q_j, q_k)$, $n_p = m_p = 0$, and $m_{1-p} = n_{1-p}$;
- $I_i = -(p, q_k, q_j)$, $n_p > 0$, $m_p = n_p - 1$, and $m_{1-p} = n_{1-p}$.

The *computation* of M on input $(n, m) \in \mathbb{N}^2$ is the unique longest configuration sequence $(p_0, n_0, m_0) \Rightarrow_M (p_1, n_1, m_1) \Rightarrow_M \cdots$ such that $p_0 = q_0$, $n_0 = n$, and $m_0 = m$. The halting problem for 2RMs is to decide, given a 2RM M, whether its computation on input $(0, 0)$ is finite (which implies that its last state is q_ℓ).

We show how to convert a given 2RM M into GF^3-sentences φ_1 and φ_2 such that M halts on input $(0, 0)$ iff $\varphi_1 \wedge \varphi_2$ is not a conservative extension of φ_1. Let $M = (Q, P)$ with $Q = q_0, \dots, q_\ell$ and $P = I_0, \dots, I_{\ell-1}$. We assume w.l.o.g. that $\ell \geq 1$ and that if $I_i = -(p, q_j, q_k)$, then $q_j \neq q_k$. In φ_1, we use the following set Σ of predicates:

- a binary predicate N connecting a configuration to its successor configuration;
- binary predicates R_1 and R_2 that represent the register contents via the length of paths;
- unary predicates $q_0, \dots, q_\ell$ representing the states of M;
- a unary predicate S denoting points where a computation starts.

We define φ_1 to be the conjunction of several GF^2-sentences. First, we say that there is a point where the computation starts:[1]

$$\exists x Sx \wedge \forall x(Sx \rightarrow (q_0 x \wedge \neg\exists y\, R_0 xy \wedge \neg\exists y\, R_1 xy))$$

And second, we add that whenever M is not in the final state, there is a next configuration. For $0 \leq i < \ell$:

$$\begin{aligned} \forall x(q_i x \rightarrow \exists y(Nxy \wedge q_j y)) &\quad \text{if } I_i = +(p, q_j) \\ \forall x((q_i x \wedge \neg\exists y R_p xy) \rightarrow \exists y(Nxy \wedge q_j y)) &\quad \text{if } I_i = -(p, q_j, q_k) \\ \forall x((q_i x \wedge \exists y R_p xy) \rightarrow \exists y(Nxy \wedge q_k y)) &\quad \text{if } I_i = -(p, q_j, q_k) \end{aligned}$$

The second sentence φ_2 is constructed so as to express that either M does not halt or the representation of the computation of M contains a defect, using the following additional predicates:

- a unary predicate P used to represent that M does not halt;
- binary predicates $D_p^+, D_p^-, D_p^=$ used to describe defects in incrementing, decrementing, and keeping register $p \in \{0, 1\}$;
- ternary predicates $H_1^+, H_2^+, H_1^-, H_2^-, H_1^=, H_2^=$ used as guards for existential quantifiers.

In fact, φ_2 is the disjunction of two sentences. The first sentence says that the computation does not terminate:

$$\exists x\, (Sx \wedge Px) \wedge \forall x\, (Px \rightarrow \exists y\, (Nxy \wedge Py))$$

while the second says that registers are not updated properly:

$$\begin{aligned} \exists x \exists y \big(Nxy \wedge \big(&\bigvee_{I_i = +(p, q_j)} (q_i x \wedge q_j y \wedge (D_p^+ xy \vee D_{1-p}^= xy)) \\ &\vee \bigvee_{I_i = -(p, q_j, q_k)} (q_i x \wedge q_k y \wedge (D_p^- xy \vee D_{1-p}^= xy)) \\ &\vee \bigvee_{I_i = -(p, q_j, q_k)} (q_i x \wedge q_j y \wedge (D_p^= xy \vee D_{1-p}^= xy))\big)\big) \\ \wedge \forall x \forall y\, (D_p^+ xy \rightarrow (&\neg\exists z\, R_p yz \vee (\neg\exists z\, R_p xz \wedge \exists z\, (R_p yz \wedge \exists x R_p zx)) \\ &\vee \exists z (H_1^+ xyz \wedge R_p xz \wedge \exists x (H_2^+ xzy \wedge R_p yx \wedge D_p^+ zx)). \end{aligned}$$

In this second sentence, additional conjuncts that implement the desired behaviour of $D_p^=$ and D_p^- are also needed; they are constructed analogously to the last three lines above (but using guards H_j^- and $H_j^=$), details are omitted. The following is proved in the appendix of the full version of this paper.

▶ **Lemma 6.**

1. *If M halts, then $\varphi_1 \wedge \varphi_2$ is not a GF^2-conservative extension of φ_1.*
2. *If there exists a Σ-structure that satisfies φ_1 and cannot be extended to a model of φ_2 (by interpreting the predicates in $\mathsf{sig}(\varphi_2) \setminus \mathsf{sig}(\varphi_1)$), then M halts.*

In the proof of Point 1, the sentence that witnesses non-conservativity describes a halting computation of M, up to global $\mathrm{GF}^2(\Sigma)$-bisimulations. This can be done using only two variables. The following result is an immediate consequence of Lemma 6.

[1] The formulas that are not syntactically guarded can easily be rewritten into such formulas.

▶ **Theorem 7.** *In any fragment of FO that extends the three-variable guarded fragment GF^3, the following problems are undecidable: conservative extensions, Σ-inseparability, Σ-entailment, and strong Σ-entailment.*

Since Point 1 of Lemma 6 ensures GF^2-witnesses, Theorem 7 can actually be strengthened to say that GF^2-conservative extensions of GF^3-sentences are undecidable.

Our result for FO^2 is proved by a reduction of a tiling problem that asks for the tiling of a rectangle (of any size) such that the borders are tiled with certain distinguished tiles. Because of space limitations, we defer details to the appendix of the full version and state only the obtained result.

▶ **Theorem 8.** *In any fragment of FO that extends FO^2, the following problems are undecidable: conservative extensions, Σ-inseparability, Σ-entailment, and strong Σ-entailment.*

It is interesting to note that the proof of Theorem 8 also shows that FO^2-conservative extensions of $\mathcal{ALC}$-TBoxes are undecidable while it follows from our results below that GF^2-conservative extensions of $\mathcal{ALC}$-TBoxes are decidable.

4 Characterizations

The undecidability results established in the previous section show that neither the restriction to two variables nor guardedness alone are sufficient for decidability of conservative extensions and related problems. In the remainder of the paper, we show that adopting both restrictions simultaneously results in decidability of Σ-entailment (and thus also of conservative extensions and of inseparability). We proceed by first establishing a suitable model-theoretic characterization and then use it as the foundation for a decision procedure based on tree automata. We in fact establish two versions of the characterization, the second one building on the first one.

We start with some preliminaries. An *atomic 1-type for* Σ is a maximal satisfiable set τ of atomic $\mathrm{GF}^2(\Sigma)$-formulas and their negations that use the variable x, only. We use $\mathrm{at}^{\Sigma}_{\mathfrak{A}}(a)$ to denote the atomic 1-type for Σ realized by the element a in the structure $\mathfrak{A}$. An *atomic 2-type for* Σ is a maximal satisfiable set τ of atomic $\mathrm{GF}^2(\Sigma)$-formulas and their negations that use the variables x and y, only, and contains $\neg(x = y)$. We say that τ is *guarded* if it contains an atom of the form Rxy or Ryx, R a predicate symbol. We use $\mathrm{at}^{\Sigma}_{\mathfrak{A}}(a, b)$ to denote the atomic 2-type for Σ realized by the elements a, b in the structure $\mathfrak{A}$. A relation $\sim\, \subseteq A \times B$ is a *$GF^2(\Sigma)$-bisimulation between* $\mathfrak{A}$ *and* $\mathfrak{B}$ if the following conditions hold whenever $a \sim b$:

1. $\mathrm{at}^{\Sigma}_{\mathfrak{A}}(a) = \mathrm{at}^{\Sigma}_{\mathfrak{B}}(b)$;
2. for every $a' \neq a$ such that $\mathrm{at}^{\Sigma}_{\mathfrak{A}}(a, a')$ is guarded, there is a $b' \neq b$ such that $\mathrm{at}^{\Sigma}_{\mathfrak{A}}(a, a') = \mathrm{at}^{\Sigma}_{\mathfrak{B}}(b, b')$ and $a' \sim b'$ (forth condition);
3. for every $b' \neq b$ such that $\mathrm{at}^{\Sigma}_{\mathfrak{B}}(b, b')$ is guarded, there is an $a' \neq a$ such that $\mathrm{at}^{\Sigma}_{\mathfrak{A}}(a, a') = \mathrm{at}^{\Sigma}_{\mathfrak{B}}(b, b')$ and $a' \sim b'$ (back condition).

We write $(\mathfrak{A}, a) \sim_{\Sigma} (\mathfrak{B}, b)$ and say that $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ are *$GF^2(\Sigma)$-bisimilar* if there is a $\mathrm{GF}^2(\Sigma)$-bisimulation $\sim$ between $\mathfrak{A}$ and $\mathfrak{B}$ with $a \sim b$. If the domain and range of $\sim$ coincide with A and B, respectively, then $\sim$ is called a *global $GF^2(\Sigma)$-bisimulation*.

We next introduce a bounded version of bisimulations. For $k \geq 0$, we write $(\mathfrak{A}, a) \sim^k_{\Sigma} (\mathfrak{B}, b)$ and say that $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ are *k-$GF^2(\Sigma)$-bisimilar* if there is a $\sim\, \subseteq A \times B$ such that the first condition for bisimulations holds and the back and forth conditions can be iterated up to k times starting from a and b; a formal definition is in the appendix of the full version. It is straightforward to show the following link between k-GF^2-bisimilarity and GF^2-sentences of guarded quantifier depth k (defined in the obvious way).

▶ **Lemma 9.** *Let $\mathfrak{A}$ and $\mathfrak{B}$ be structures, Σ a signature, and $k \geq 0$. Then the following conditions are equivalent:*

1. *for all $a \in A$ there exists $b \in B$ with $(\mathfrak{A}, a) \sim^k_\Sigma (\mathfrak{B}, b)$ and vice versa;*
2. *$\mathfrak{A}$ and $\mathfrak{B}$ satisfy the same $GF^2(\Sigma)$-sentences of guarded quantifier depth at most k.*

The corresponding lemma for $\mathrm{GF}^2(\Sigma)$-sentences of unbounded guarded quantifier depth and $\mathrm{GF}^2(\Sigma)$-bisimulations holds if $\mathfrak{A}$ and $\mathfrak{B}$ satisfy certain saturation conditions (for example, if $\mathfrak{A}$ and $\mathfrak{B}$ are ω-saturated). It can then be proved that an FO-sentence φ is equivalent to a GF^2 sentence iff its models are preserved under global $\mathrm{GF}^2(\mathsf{sig}(\varphi))$-bisimulations [18, 14]. In modal and description logic, global Σ-bisimulations can often be used to characterize Σ-entailment in the following natural way [27]: φ_1 Σ-entails φ_2 iff every for every (tree) model $\mathfrak{A}$ of φ_1, there exists a (tree) model $\mathfrak{B}$ of φ_2 that is globally Σ-bisimilar to $\mathfrak{A}$. Such a characterization enables decision procedures based on tree automata, but does not hold for GF^2.

▶ **Example 10.** Let $\varphi_1 = \forall x \exists y Rxy$ and let $\varphi_2 = \varphi_1 \wedge \exists x Bx \wedge \forall x(Bx \to \exists y(Ryx \wedge By))$. Let $\mathfrak{A}$ be the model of φ_1 that consists of an infinite R-path with an initial element. Then there is no model of φ_2 that is globally $\mathrm{GF}^2(\{R\})$-bisimilar to $\mathfrak{A}$ since any such model has to contain an infinite R-path with no initial element. Yet, φ_2 is a conservative extension of φ_1 which can be proved using Theorem 11 below.

We give our first characterization theorem that uses unbounded bisimulations in one direction and bounded bisimulations in the other.

▶ **Theorem 11.** *Let φ_1, φ_2 be GF_2-sentences and Σ a signature. Then $\varphi_1 \models_\Sigma \varphi_2$ iff for every model $\mathfrak{A}$ of φ_1 of finite outdegree, there is a model $\mathfrak{B}$ of φ_2 such that*

1. *for every $a \in A$ there is a $b \in B$ such that $(\mathfrak{A}, a) \sim_\Sigma (\mathfrak{B}, b)$*
2. *for every $b \in B$ and every $k \geq 0$, there is an $a \in A$ such that $(\mathfrak{A}, a) \sim^k_\Sigma (\mathfrak{B}, b)$.*

The direction ($\Leftarrow$) follows from Lemma 9 and ($\Rightarrow$) can be proved using compactness and ω-saturated structures. Because of the use of k-bounded bisimulations (for unbounded k), it is not clear how to use Theorem 11 to find a decision procedure based on tree automata. In the following, we formulate a more 'operational' but also more technical characterization that no longer mentions bounded bisimulations. It additionally refers to forest models $\mathfrak{A}$ of φ_1 (of finite outdegree) instead of unrestricted models, but we remark that Theorem 11 also remains true under this modification.

A structure $\mathfrak{A}$ is a *forest* if its Gaifman graph is a forest. Thus, a forest admits cycles of length one and two, but not of any higher length. A (Σ-)*tree* in a forest structure $\mathfrak{A}$ is a maximal (Σ)-connected substructure of $\mathfrak{A}$. When working with forest structures $\mathfrak{A}$, we will typically view them as directed forests rather than as undirected ones. This can be done by choosing a root for each tree in the Gaifman graph of $\mathfrak{A}$, thus giving rise to notions such as successor, descendant, etc. Which node is chosen as the root will always be irrelevant. Note that the direction of binary relations does not need to reflect the successor relation. When speaking of a *path* in a forest structure $\mathfrak{A}$, we mean a path in the directed sense; when speaking of a *subtree*, we mean a tree that is obtained by choosing a root a and restricting the structure to a and its descendants. We say that $\mathfrak{A}$ is *regular* if it has only finitely many subtrees, up to isomorphism.

To see how we can get rid of bounded bisimulations, reconsider Theorem 11. The characterization is still correct if we pull out the quantification over k in Point 2 so that the theorem reads '...iff for every model $\mathfrak{A}$ of φ_1 of finite outdegree and every $k \geq 0$, there is...'. In fact, this modified version of Theorem 11 is even closer to the definition of Σ-entailment. It also suggests that we add a marking $A_\perp \subseteq A$ of elements in $\mathfrak{A}$, representing 'break-off points' for bisimulations, and then replace k-bisimulations with bisimulations that stop whenever they have encountered the *second* marked element on the same path—in this way, the distance between marked elements (roughly) corresponds to the bound k in k-bisimulations. However, we would need a marking $A_\perp$, for any $k \geq 0$, such that there are infinitely many markers on any infinite path and the distance between any two markers in a tree is at least k. It is easy to see that such a marking may not exist, for example when $k = 3$ and $\mathfrak{A}$ is the infinite full binary tree. We solve this problem as follows. First, we only demand that the distance between any two markers *on the same path* is at least k. And second, we use the markers only when following bisimulations upwards in a tree while downwards, we use unbounded bisimulations. This does not compromise correctness of the characterization.

We next introduce a version of bisimulations that implement the ideas just explained. Let $\mathfrak{A}$ and $\mathfrak{B}$ be forest models, Σ a signature, and $A_\perp \subseteq A$. Two relations $\sim_\Sigma^{A_\perp,0}, \sim_\Sigma^{A_\perp,1} \subseteq A \times B$ form an *$A_\perp$-delimited GF$^2(\Sigma)$-bisimulation* between $\mathfrak{A}$ and $\mathfrak{B}$ if the following conditions are satisfied:

1. if $(\mathfrak{A}, a) \sim_\Sigma^{A_\perp,0} (\mathfrak{B}, b)$, then $\mathrm{at}_\mathfrak{A}^\Sigma(a) = \mathrm{at}_\mathfrak{B}^\Sigma(b)$ and
   a. for every $a' \neq a$ with $\mathrm{at}_\mathfrak{A}^\Sigma(a, a')$ guarded, there is a $b' \neq b$ such that $(\mathfrak{A}, a') \sim_\Sigma^{A_\perp,i} (\mathfrak{B}, b')$ where $i = 1$ if a' is the predecessor of a and $a' \in A_\perp$, and $i = 0$ otherwise;
   b. for every $b' \neq b$ with $\mathrm{at}_\mathfrak{B}^\Sigma(b, b')$ guarded, there is an $a' \neq a$ such that $(\mathfrak{A}, a') \sim_\Sigma^{A_\perp,i} (\mathfrak{B}, b')$ where $i = 1$ if a' is the predecessor of a and $a' \in A_\perp$, and $i = 0$ otherwise;
2. if $(\mathfrak{A}, a) \sim_\Sigma^{A_\perp,1} (\mathfrak{B}, b)$ and the predecessor of a in $\mathfrak{A}$ is not in $A_\perp$, then $\mathrm{at}_\mathfrak{A}^\Sigma(a) = \mathrm{at}_\mathfrak{B}^\Sigma(b)$ and
   a. for every $a' \neq a$ with $\mathrm{at}_\mathfrak{A}^\Sigma(a, a')$ guarded, there is a $b' \neq b$ such that $(\mathfrak{A}, a') \sim_\Sigma^{A_\perp,i} (\mathfrak{B}, b')$ where $i = 0$ if a is the predecessor of a' and $a \in A_\perp$, and $i = 1$ otherwise;
   b. for every $b' \neq b$ with $\mathrm{at}_\mathfrak{B}^\Sigma(b, b')$ guarded, there is an $a' \neq a$ such that $(\mathfrak{A}, a') \sim_\Sigma^{A_\perp,i} (\mathfrak{B}, b')$ where $i = 0$ if a is the predecessor of a' and $a \in A_\perp$, and $i = 1$ otherwise.

Then $(\mathfrak{A}, a)$ and $(\mathfrak{B}, b)$ are *$A_\perp$-delimited GF$^2(\Sigma)$-bisimilar*, in symbols $(\mathfrak{A}, a) \sim_\Sigma^{A_\perp} (\mathfrak{B}, b)$, if there exists an $A_\perp$-delimited GF$^2(\Sigma)$-bisimulation $\sim_\Sigma^{A_\perp,0}, \sim_\Sigma^{A_\perp,1}$ between $\mathfrak{A}$ and $\mathfrak{B}$ such that $(\mathfrak{A}, a) \sim_\Sigma^{A_\perp,0} (\mathfrak{B}, b)$.

Let φ be a GF^2-sentence. We use $\mathsf{cl}(\varphi)$ to denote the set of all subformulas of φ closed under single negation and renaming of free variables (using only the available variables x and y). A *1-type for* φ is a subset $t \subseteq \mathsf{cl}(\varphi)$ that contains only formulas of the form $\psi(x)$ and such that $\varphi \wedge \exists x \bigwedge t(x)$ is satisfiable. For a model $\mathfrak{A}$ of φ and $a \in A$, we use $\mathsf{tp}_\mathfrak{A}(a)$ to denote the 1-type $\{\psi(x) \in \mathsf{cl}(\varphi) \mid \mathfrak{A} \models \psi(a)\}$, assuming that φ is understood from the context. We say that the 1-type t is *realized* in $\mathfrak{A}$ if there is an $a \in A$ with $\mathsf{tp}_\mathfrak{A}(a) = t$. We are now ready to formulate our final characterizations.

▶ **Theorem 12.** *Let φ_1, φ_2 be GF^2-sentences and Σ a signature. Then $\varphi_1 \models_\Sigma \varphi_2$ iff for every regular forest model $\mathfrak{A}$ of φ_1 that has finite outdegree and for every set $A_\perp \subseteq A$ with $A_\perp \cap \rho$ infinite for any infinite Σ-path ρ in $\mathfrak{A}$, there is a model $\mathfrak{B}$ of φ_2 such that*
1. *for every $a \in A$, there is a $b \in B$ such that $(\mathfrak{A}, a) \sim_\Sigma (\mathfrak{B}, b)$;*
2. *for every 1-type t for φ_2 that is realized in $\mathfrak{B}$, there are $a \in A$ and $b \in B$ such that $\mathsf{tp}_\mathfrak{B}(b) = t$ and $(\mathfrak{A}, a) \sim_\Sigma^{A_\perp} (\mathfrak{B}, b)$.*

Regularity and finite outdegree are used in the proof of Theorem 12 given in the appendix of the full version, but it follows from the automata constructions below that the theorem is still correct when these qualifications are dropped.

5 Decidability and Complexity

We show that Σ-entailment in GF^2 is decidable and 2ExpTime-complete, and thus so are conservative extensions and Σ-inseparability. The upper bound is based on Theorem 12 and uses alternating parity automata on infinite trees. Since Theorem 12 does not provide us with an obvious upper bound on the outdegree of the involved tree models, we use alternating tree automata which can deal with trees of any finite outdegree, similar to the ones introduced by Wilke [37], but with the capability to move both downwards and upwards in the tree.

A *tree* is a non-empty (and potentially infinite) set of words $T \subseteq (\mathbb{N} \setminus 0)^*$ closed under prefixes. We generally assume that trees are finitely branching, that is, for every $w \in T$, the set $\{i \mid w \cdot i \in T\}$ is finite. For any $w \in (\mathbb{N} \setminus 0)^*$, as a convention we set $w \cdot 0 := w$. If $w = n_0 n_1 \cdots n_k$, we additionally set $w \cdot -1 := n_0 \cdots n_{k-1}$. For an alphabet Θ, a *Θ-labeled tree* is a pair (T, L) with T a tree and $L : T \to \Theta$ a node labeling function.

A *two-way alternating tree automata (2ATA)* is a tuple $\mathcal{A} = (Q, \Theta, q_0, \delta, \Omega)$ where Q is a finite set of *states*, Θ is the *input alphabet*, $q_0 \in Q$ is the *initial state*, δ is a *transition function* as specified below, and $\Omega : Q \to \mathbb{N}$ is a *priority function*, which assigns a priority to each state. The transition function maps a state q and some input letter $\theta \in \Theta$ to a *transition condition* $\delta(q, \theta)$ which is a positive Boolean formula over the truth constants true and false and transitions of the form q, $\langle - \rangle q$, $[-]q$, $\Diamond q$, $\Box q$ where $q \in Q$. The automaton runs on Θ-labeled trees. Informally, the transition q expresses that a copy of the automaton is sent to the current node in state q, $\langle - \rangle q$ means that a copy is sent in state q to the predecessor node, which is then required to exist, $[-]q$ means the same except that the predecessor node is not required to exist, $\Diamond q$ means that a copy is sent in state q to some successor, and $\Box q$ that a copy is sent in state q to all successors. The semantics is defined in terms of runs in the usual way, we refer to the appendix of the full version for details. We use $L(\mathcal{A})$ to denote the set of all Θ-labeled trees accepted by $\mathcal{A}$. It is standard to verify that 2ATAs are closed under complementation and intersection. We show in the appendix that the emptiness problem for 2ATAs can be solved in time exponential in the number of states.

We aim to show that given two GF^2-sentences φ_1 and φ_2 and a signature Σ, one can construct a 2ATA $\mathcal{A}$ such that $L(\mathcal{A}) = \emptyset$ iff $\varphi_1 \models_{\mathrm{GF}^2(\Sigma)} \varphi_2$. The number of states of the 2ATA $\mathcal{A}$ is polynomial in the size of φ_1 and exponential in the size of φ_2, which yields the desired 2ExpTime upper bounds.

Let φ_1, φ_2, and Σ be given. Since the logics we are concerned with have Craig interpolation, we can assume w.l.o.g. that $\Sigma \subseteq \mathsf{sig}(\varphi_1)$. With Θ, we denote the set of all pairs (τ, M) where τ is an atomic 2-type for $\mathsf{sig}(\varphi_1)$ and $M \in \{0, 1\}$. For $p = (\tau, M) \in \Theta$, we use p^1 to denote τ and p^2 to denote M. A Θ-labeled tree (T, L) represents a forest structure $\mathfrak{A}_{(T,L)}$ with universe $A_{(T,L)} = T$ and where $w \in A^{\mathfrak{A}_{(T,L)}}$ if $A(y) \in L(w)$ and $(w, w') \in R^{\mathfrak{A}_{(T,L)}}$ if one of the following conditions is satisfied: (1) $w = w'$ and $Ryy \in L(w)^1$; (2) w' is a successor of w and $Rxy \in L(w')^1$; (3) w is a successor of w' and $Ryx \in L(w)^1$. Thus, the atoms in a node label that involve only the variable y describe the current node, the atoms that involve both variables x and y describe the connection between the predecessor and the current node, and the atoms that involve only the variable x are ignored. The M-components of node labels are used to represent a set of markers $A_\perp = \{w \in A_{(T,L)} \mid L(w)^2 = 1\}$. It is easy to see that, conversely, for every tree structure $\mathfrak{A}$ over Σ, there is a Θ-labeled tree (T, L) such that $\mathfrak{A}_{(T,L)} = \mathfrak{A}$.

To obtain the desired 2ATA $\mathcal{A}$, we construct three 2ATAs $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$ and then define $\mathcal{A}$ so that it accepts $L(\mathcal{A}_1) \cap \overline{L(\mathcal{A}_2)} \cap L(\mathcal{A}_3)$. The 2ATA $\mathcal{A}_3$ only makes sure that the set $A_\perp \subseteq A_{(T,L)}$ is such that for any infinite Σ-path ρ, $A_\perp \cap \rho$ is infinite (as required by

Theorem 12), we omit details. We construct $\mathcal{A}_1$ so that it accepts a Θ-labeled tree (T, L) iff $\mathfrak{A}_{(T,L)}$ is a model of φ_1. The details of the construction, which is fairly standard, can be found in the appendix. The number of states of $\mathcal{A}_1$ is polynomial in the size of φ_1 and independent of φ_2. The most interesting automaton is $\mathcal{A}_2$.

▶ **Lemma 13.** *There is a 2ATA $\mathcal{A}_2$ that accepts a Θ-labeled tree (T, L) iff there is a model $\mathfrak{B}$ of φ_2 s.t. Conditions 1 and 2 from Theorem 12 are satisfied when $\mathfrak{A}$ is replaced with $\mathfrak{A}_{(T,L)}$.*

The general idea of the construction of $\mathcal{A}_2$ is to check the existence of the desired model $\mathfrak{B}$ of φ_2 by verifying that there is a set of 1-types for φ_2 from which $\mathfrak{B}$ can be assembled, represented via the states that occur in a successful run. Before we can give details, we introduce some preliminaries.

A 0-*type* s *for* φ_2 is a maximal set of sentences $\psi() \in \mathsf{cl}(\varphi_2)$ such that $\varphi_2 \wedge s$ is satisfiable. A 2-*type* λ *for* φ_2 is a maximal set of formulas $\psi(x, y) \in \mathsf{cl}(\varphi_2)$ that contains $\neg(x = y)$ and such that $\varphi_2 \wedge \exists xy\, \lambda(x, y)$ is satisfiable. If a 2-type λ contains the atom Rxy or Ryx for at least one binary predicate R, then it is *guarded*. If additionally $R \in \Sigma$, then it is Σ-*guarded*. Note that each 1-type contains a (unique) 0-type and each 2-type contains two (unique) 1-types. Formally, we use λ_x to denote the 1-type obtained by restricting the 2-type λ to the formulas that do not use the variable y, and likewise for λ_y and the variable x. We use TP_n to denote the set of n-types for φ_2, $n \in \{0, 1, 2\}$. For $t \in \mathsf{TP}_1$ and a $\lambda \in \mathsf{TP}_2$, we say that λ is *compatible with* t and write $t \approx \lambda$ if the sentence $\varphi_2 \wedge \exists xy(t(x) \wedge \lambda(x, y))$ is satisfiable; for $t \in \mathsf{TP}_1$ and $T \subseteq \mathsf{TP}_2$ a set of guarded 2-types, we say that T is a *neighborhood for* t and write $t \approx T$ if the sentence

$$\varphi_2 \wedge \exists x\big(t(x) \wedge \bigwedge_{\lambda \in T} \exists y\, \lambda(x, y) \wedge \forall y \bigvee_{R \in \mathsf{sig}(\varphi_2)} ((Rxy \vee Ryx) \rightarrow \bigvee_{\lambda \in T} \lambda(x, y)\big)\big)$$

is satisfiable. Note that each of the mentioned sentences is formulated in GF^2 and at most single exponential in size (in the size of φ_1 and φ_2), thus satisfiability can be decided in 2ExpTime.

To build the automaton $\mathcal{A}_2$ from Lemma 13, set $\mathcal{A}_2 = (Q_2, \Theta, q_0, \delta_2, \Omega_2)$ where Q_2 is

$$\{q_0, q_\perp\} \cup \mathsf{TP}_0 \cup \{t, t^?, t_\uparrow, t_\downarrow, t_\&, t^i, t^i_\uparrow, t^i_\downarrow \mid t \in \mathsf{TP}_1,\ i \in \{0, 1\}\} \cup$$
$$\{\lambda, \lambda_\uparrow, \lambda^i, \lambda^i_\uparrow \mid \lambda \in \mathsf{TP}_2,\ i \in \{0, 1\}\},$$

Ω_2 assigns two to all states except for those of the form $t^?$, to which it assigns one.

The automaton begins by choosing the 0-type s realized in the forest model $\mathfrak{B}$ of φ_2 whose existence it aims to verify. For every $\exists x \varphi(x) \in s$, it then chooses a 1-type t in which $\varphi(x)$ is realized in $\mathfrak{B}$ and sends off a copy of itself to find a node where t is realized. To satisfy Condition 1 of Theorem 12, at each node it further chooses a 1-type that is compatible with s, to be realized at that node. This is implemented by the following transitions:

$$\begin{aligned}
\delta_2(q_0, \sigma) &= \bigvee_{s \in \mathsf{TP}_0} (s \wedge \bigwedge_{\exists x\, \varphi(x) \in s} \bigvee_{\substack{t \in \mathsf{TP}_1 \mid \\ s \cup \{\varphi(x)\} \subseteq t}} t^?) \\
\delta_2(s, \sigma) &= \Box s \wedge \bigvee_{t \in \mathsf{TP}_1, s \subseteq t} t \\
\delta_2(t^?, \sigma) &= \langle -1 \rangle t^? \vee \Diamond t^? \vee t^0
\end{aligned}$$

where s ranges over TP_0. When a state of the form t is assigned to a node w, this is an obligation to prove that there is a $\mathrm{GF}^2(\Sigma)$-bisimulation between the element w in $\mathfrak{A}_{(T,L)}$ and an element b of type t in $\mathfrak{B}$. A state of the form t^0 represents the obligation to verify that there is an $A_\perp$-delimited $\mathrm{GF}^2(\Sigma)$-bisimulation between w and an element of type t in $\mathfrak{B}$. We first verify that the former obligations are satisfied. This requires to follow all successors of w and to guess types of successors of b to be mapped there, satisfying the back condition of bisimulations. We also need to guess successors of b in $\mathfrak{B}$ (represented as a neighborhood for t) to satisfy the existential demands of t and then select successors of a to which they are mapped, satisfying the "back" condition of bisimulations. Whenever we decide to realize a 1-type t in $\mathfrak{B}$ that does not participate in the bisimulation currently being verified, we also send another copy of the automaton in state $t^?$ to guess an $a \in A_{(T,L)}$ that we can use to satisfy Condition 2 from Theorem 12:

$$\delta_2(t,(\tau,M)) = t_\uparrow \wedge \Box t_\downarrow \wedge \bigvee_{T \mid t \approx T} \bigwedge_{\lambda \in T} (\Diamond\lambda \vee \lambda_\uparrow) \quad \text{if } \tau_y =_\Sigma t$$

$$\begin{aligned}
\delta_2(t,(\tau,M)) &= \mathsf{false} && \text{if } \tau_y \neq_\Sigma t\\
\delta_2(t_\downarrow,(\tau,M)) &= \mathsf{true} && \text{if } \tau \text{ is not } \Sigma\text{-guarded}\\
\delta_2(t_\downarrow,(\tau,M)) &= \bigvee_{\lambda \mid t \approx \lambda \wedge \tau =_\Sigma \lambda} \lambda_y && \text{if } \tau \text{ is } \Sigma\text{-guarded}\\
\delta_2(t_\uparrow,(\tau,M)) &= \mathsf{true} && \text{if } \tau \text{ is not } \Sigma\text{-guarded}\\
\delta_2(t_\uparrow,(\tau,M)) &= \bigvee_{\lambda \mid t \approx \lambda \wedge \tau =_\Sigma \lambda^-} [-1]\lambda_y && \text{if } \tau \text{ is } \Sigma\text{-guarded}\\
\delta_2(\lambda,(\tau,M)) &= \lambda_y && \text{if } \lambda \text{ is } \Sigma\text{-guarded and } \tau =_\Sigma \lambda\\
\delta_2(\lambda,(\tau,M)) &= \mathsf{false} && \text{if } \lambda \text{ is } \Sigma\text{-guarded and } \tau \neq_\Sigma \lambda\\
\delta_2(\lambda,(\tau,M)) &= \lambda_y^? && \text{if } \lambda \text{ is not } \Sigma\text{-guarded}\\
\delta_2(\lambda_\uparrow,(\tau,M)) &= \langle -1\rangle\lambda_y && \text{if } \lambda \text{ is } \Sigma\text{-guarded and } \tau =_\Sigma \lambda^-\\
\delta_2(\lambda_\uparrow,(\tau,M)) &= \mathsf{false} && \text{if } \lambda \text{ is } \Sigma\text{-guarded and } \tau \neq_\Sigma \lambda^-\\
\delta_2(\lambda_\uparrow,(\tau,M)) &= \lambda_y^? && \text{if } \lambda \text{ is not } \Sigma\text{-guarded}
\end{aligned}$$

where $\tau_y =_\Sigma t$ means that the atoms in τ that mention only y are identical to the Σ-relational atoms in t (up to renaming x to y), $\tau =_\Sigma \lambda$ means that the restriction of λ to Σ-atoms is exactly τ, and λ^- is obtained from λ by swapping x and y. We need further transitions to satisfy the obligations represented by states of the form t^0, which involves checking $A_\perp$-delimited bisimulations. Details are given in the appendix where also the correctness of the construction is proved.

▶ **Theorem 14.** *In GF^2, Σ-entailment and conservative extensions can be decided in time $2^{2^{p(|\varphi_2|\cdot\log|\varphi_1|)}}$, for some polynomial p. Moreover, Σ-inseparability is in* 2ExpTime.

Note that the time bound for conservative extensions given in Theorem 14 is double exponential only in the size of φ_2 (that is, the extension). In ontology engineering applications, φ_2 will often be small compared with φ_1.

A matching lower bound is proved using a reduction of the word problem of exponentially space-bounded alternating Turing machines, see the appendix for details. The construction is inspired by the proof from [13] that conservative extensions in the description logic $\mathcal{ALC}$ are 2ExpTime-hard, but the lower bound does not transfer directly since we are interested here in witness sentences that are formulated in GF^2 rather than in $\mathcal{ALC}$.

▶ **Theorem 15.** *In any fragment of FO that contains GF^2, the problems Σ-entailment, Σ-inseparability, conservative extensions, and strong Σ-entailment are* 2ExpTime*-hard.*

6 Conclusion

We have shown that conservative extensions are undecidable in (extensions of) GF and FO^2, and that they are decidable and 2ExpTime-complete in GF^2. It thus appears that decidability of conservative extensions is linked even more closely to the tree model property than decidability of the satisfiability problem: apart from cycles of length at most two, GF^2 enjoys a 'true' tree model property while GF only enjoys a bounded treewidth model property and FO^2 has a rather complex regular model property that is typically not even made explicit. As future work, it would be interesting to investigate whether conservative extensions remain decidable when guarded counting quantifiers, transitive relations, equivalence relations, or fixed points are added, see e.g. [32, 22, 20]. It would also be interesting to investigate a finite model version of conservative extensions.

References

1 Hajnal Andréka, István Németi, and Johan van Benthem. Modal languages and bounded fragments of predicate logic. *J. Philosophical Logic*, 27(3):217–274, 1998.

2 Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, 2003. (2nd edition, 2007).

3 Vince Bárány, Georg Gottlob, and Martin Otto. Querying the guarded fragment. *Logical Methods in Computer Science*, 10(2), 2014.

4 Vince Bárány, Balder ten Cate, and Luc Segoufin. Guarded negation. *J. ACM*, 62(3):22:1–22:26, 2015.

5 Michael Benedikt, Balder ten Cate, and Michael Vanden Boom. Interpolation with decidable fixpoint logics. In *Proc. of LICS*, pages 378–389. IEEE Computer Society, 2015.

6 Egon Börger, Erich Grädel, and Yuri Gurevich. *The Classical Decision Problem*. Perspectives in Mathematical Logic. Springer, 1997.

7 Elena Botoeva, Boris Konev, Carsten Lutz, Vladislav Ryzhikov, Frank Wolter, and Michael Zakharyaschev. Inseparability and conservative extensions of description logic ontologies: A survey. In *Proc. of Reasoning Web*, volume 9885 of *LNCS*, pages 27–89. Springer, 2016.

8 Stephen D. Comer. Classes without the amalgamation property. *Pacific Journal of Mathematics*, 28:309–318, 1969.

9 Bernardo Cuenca Grau, Ian Horrocks, Yevgeny Kazakov, and Ulrike Sattler. Modular reuse of ontologies: Theory and practice. *Journal of Artificial Intelligence Research (JAIR)*, 31:273–318, 2008.

10 Giovanna D'Agostino and Marco Hollenberg. Logical questions concerning the μ-calculus: Interpolation, Lyndon and ł oś-Tarski. *J. Symb. Log.*, 65(1):310–332, 2000.

11 Giovanna D'Agostino and Giacomo Lenzi. Bisimulation quantifiers and uniform interpolation for guarded first order logic. *Theor. Comput. Sci.*, 563:75–85, 2015.

12 Răzvan Diaconescu, Joseph A. Goguen, and Petros Stefaneas. Logical support for modularisation. In Gerard Huet and Gordon Plotkin, editors, *Logical Environments*, pages 83–130, 1993.

13 Silvio Ghilardi, Carsten Lutz, and Frank Wolter. Did I damage my ontology? A case for conservative extensions in description logic. In *Proc. of KR*, pages 187–197. AAAI Press, 2006.

14 Valentin Goranko and Martin Otto. Model theory of modal logic. In Patrick Blackburn, Johan van Benthem, and Frank Wolter, editors, *Handbook of Modal Logic*, pages 249–330. Elsevier, 2006.

15 Michael J. C. Gordon and Thomas F. Melham, editors. *Introduction to HOL: A theorem proving environment for higher order logic.* Cambridge University Press, 1993.

16 Erich Grädel. On the restraining power of guards. *J. Symb. Log.*, 64(4):1719–1742, 1999.

17 Erich Grädel, Phokion G. Kolaitis, and Moshe Y. Vardi. On the decision problem for two-variable first-order logic. *Bulletin of Symbolic Logic*, 3(1):53–69, 1997.

18 Erich Grädel and Martin Otto. The freedoms of (guarded) bisimulation. In Alexandru Baltag and Sonja Smets, editors, *Johan van Benthem on Logic and Information Dynamics*, pages 3–31. Springer, 2014.

19 Erich Grädel, Martin Otto, and Eric Rosen. Two-variable logic with counting is decidable. In *Proc. of LICS*, pages 306–317. IEEE Computer Society, 1997.

20 Erich Grädel and Igor Walukiewicz. Guarded fixed point logic. In *Proc. of LICS*, pages 45–54. IEEE Computer Society, 1999.

21 Eva Hoogland and Maarten Marx. Interpolation and definability in guarded fragments. *Studia Logica*, 70(3):373–409, 2002.

22 Emanuel Kieronski. On the complexity of the two-variable guarded fragment with transitive guards. *Inf. Comput.*, 204(11):1663–1703, 2006.

23 Emanuel Kieronski, Jakub Michaliszyn, Ian Pratt-Hartmann, and Lidia Tendera. Two-variable first-order logic with equivalence closure. *SIAM J. Comput.*, 43(3):1012–1063, 2014.

24 Boris Konev, Carsten Lutz, Dirk Walther, and Frank Wolter. Formal properties of modularisation. In Heiner Stuckenschmidt, Christine Parent, and Stefano Spaccapietra, editors, *Modular Ontologies: Concepts, Theories and Techniques for Knowledge Modularization*, volume 5445 of *LNCS*, pages 25–66. Springer, 2009.

25 Boris Konev, Carsten Lutz, Dirk Walther, and Frank Wolter. Model-theoretic inseparability and modularity of description logic ontologies. *Artificial Intelligence*, 203:66–103, 2013.

26 Carsten Lutz, Dirk Walther, and Frank Wolter. Conservative extensions in expressive description logics. In *IJCAI*, pages 453–458, 2007.

27 Carsten Lutz and Frank Wolter. Foundations for uniform interpolation and forgetting in expressive description logics. In *Proc. of IJCAI*, pages 989–995. IJCAI/AAAI, 2011.

28 Johannes Marti, Fatemeh Seifan, and Yde Venema. Uniform interpolation for coalgebraic fixpoint logic. In *CALCO*, volume 35 of *LIPIcs*, pages 238–252. Schloss Dagstuhl, 2015.

29 Michael Mortimer. On languages with two variables. *Math. Log. Q.*, 21(1):135–140, 1975.

30 Andrew M. Pitts. On an interpretation of second-order quantification in first-order intuitionistic propositional logic. *J. of Symbolic Logic*, 57, 1992.

31 Stephen Pollard. *Philosophical Introduction to Set Theory.* University of Notre Dame Press, 1990.

32 Ian Pratt-Hartmann. Complexity of the guarded two-variable fragment with counting quantifiers. *J. Log. Comput.*, 17(1):133–155, 2007.

33 Ian Pratt-Hartmann. Data-complexity of the two-variable fragment with counting quantifiers. *Inf. Comput.*, 207(8):867–888, 2009.

34 Dana Scott. A decision method for validity of sentences in two variables. *Journal of Symbolic Logic*, 27:1962, 1962.

35 Stephen G. Simpson. *Subsystems of Second Order Arithmetic.* Cambridge University Press, 2009.

36 Albert Visser. Uniform interpolation and layered bisimulation. In *Gödel'96 (Brno, 1996)*, volume 6 of *Lecture Notes in Logic*, pages 139–164. Springer, 1996.

37 Thomas Wilke. Alternating tree automata, parity games, and modal μ-calculus. *Bulletin of the Belgian Mathematical Society*, 8(2), 2001.