Finite Satisfiability of Unary Negation Fragment with Transitivity

Daniel Danielski
University of Wrocław, Poland

Emanuel Kieroński
University of Wrocław, Poland
kiero@cs.uni.wroc.pl

Abstract
We show that the finite satisfiability problem for the unary negation fragment with an arbitrary number of transitive relations is decidable and $2\text{-ExpTime}$-complete. Our result actually holds for a more general setting in which one can require that some binary symbols are interpreted as arbitrary transitive relations, some as partial orders and some as equivalences. We also consider finite satisfiability of various extensions of our primary logic, in particular capturing the concepts of nominals and role hierarchies known from description logic. As the unary negation fragment can express unions of conjunctive queries, our results have interesting implications for the problem of finite query answering, both in the classical scenario and in the description logics setting.

2012 ACM Subject Classification Theory of computation → Logic

Keywords and phrases unary negation fragment, transitivity, finite satisfiability, finite open-world query answering, description logics

Funding Supported by Polish National Science Centre grant No 2016/21/B/ST6/01444.

1 Introduction

Decidable fragments and unary negation. Searching for attractive fragments of first-order logic is an important theme in theoretical computer science. Successful examples of such fragments, with numerous applications, are modal and description logics. They have their own syntax, but naturally translate to first-order logic, via the standard translation. Several seminal decidable fragments of first-order logic were identified by preserving one particular restriction obeyed by this translation and dropping all the others. Important examples of such fragments are two-variable logic, $\text{FO}^2$, [25], the guarded fragment, $\text{GF}$, [2], and the fluted fragment, $\text{FF}$, [24, 22]. They restrict, respectively, the number of variables, the quantification pattern and the order of variables in which they appear as arguments of predicates. A more recent proposal [27] is the unary negation fragment, UNFO. This time we restrict the use of negations, allowing them only in front of subformulas with at most one free variable. UNFO turns out to retain many good algorithmic and model theoretic properties of modal logic, including the finite model property, a tree-like model property and the decidability of the satisfiability problem. We remark here that UNFO and GF have a common decidable generalization, the guarded negation fragment, $\text{GNFO}$, [5].

To justify the attractiveness of UNFO let us look at one of the crucial problems in database theory, open-world query answering. Given an (incomplete) set of facts $\mathcal{D}$, a set of constraints $\mathcal{T}$ and a query $q$, check if $\mathcal{D} \land \mathcal{T}$ entails $q$. Generally, this problem is undecidable, and to make it decidable one needs to restrict the class of queries and constraints. Widely investigated class of queries are (unions of) conjunctive queries—(disjunctions of) sentences of the form $\exists \bar{x}\psi(\bar{x})$ where $\psi$ is a conjunction of atoms. An important class of constraints are tuple generating dependencies, TGDs, of the form $\forall \bar{x}\bar{y}(\psi(\bar{x}, \bar{y}) \rightarrow \exists \bar{z}\psi'(\bar{y}, \bar{z}))$, where $\psi$ and $\psi'$ are, again, conjunctions of atoms. Conjunctive query answering against arbitrary
Finite Satisfiability of Unary Negation Fragment with Transitivity

TGDs is still undecidable (see, e.g., [6]), so TGDs need to be restricted further. Several classes of TGDs making the problem decidable have been proposed. One interesting such class are frontier-one TGDs, in which the frontier of each dependency, $\bar{y}$, consists just of a single variable [4]. Frontier-one TGDs are a special case of frontier-guarded TGDs [3]. Checking whether $\mathcal{D}$ and $\mathcal{T}$ entail $q$ boils down to verifying (un)satisfiability of the formula $\mathcal{D} \land \mathcal{T} \land \neg q$. It turns out that if $\mathcal{T}$ is a conjunction of frontier-one TGDs and $q$ is a disjunction of conjunctive queries then the resulting formula belongs to UNFO.

Transitivity. A serious weakness of the expressive power of UNFO is that it cannot express transitivity of a binary relation, nor related properties like being an equivalence, a partial order or a linear order. This limitation becomes particularly important when database or knowledge representation applications are considered, as transitivity is a natural property in many real-life situations. Just consider relations like greater-than or part-of. This weakness is shared by $\text{FO}^2$, $\text{GF}$ and $\text{FF}$. Thus, it is natural to think about their extensions, in which some distinguished binary symbols may be explicitly required to be interpreted as transitive relations. It turns out that $\text{FO}^2$, $\text{GF}$ and $\text{FF}$ do not cope well with transitivity, and the satisfiability problems for the obtained extensions are undecidable [15, 13, 23] (see also [10, 18, 17]). Some positive results were obtained for $\text{FO}^2$, $\text{GF}$ and $\text{FF}$ only when one transitive relation is available [21, 18, 23] or when some further syntactic restrictions are imposed [26].

UNFO is an exception here, since its satisfiability problem remains decidable in the presence of arbitrarily many transitive relations. This has been explicitly stated in [16], as a corollary from a stronger result that UNFO is decidable when extended by regular path expressions. Independently, the decidability of UNFO with transitivity, $\text{UNFO} + S$, follows from [1], which deals with the decidability of a richer logic, the guarded negation fragment with transitive relations restricted to non-guard positions, which embeds $\text{UNFO} + S$. From both papers the $2\text{-ExpTime}$-completeness of $\text{UNFO} + S$ can be inferred.

Our main results. A problem related to satisfiability is finite satisfiability, in which we ask about the existence of finite models. In computer science, the importance of decision procedures for finite satisfiability arises from the fact that most objects about which we may want to reason using logic, e.g., databases, are finite. Thus the ability of solving only general satisfiability may not be fully satisfactory. Both the above-mentioned decidability results implying the decidability of $\text{UNFO} + S$ are obtained by employing tree-like model properties of the logics and then using automata techniques. Since tree-like unravelings of models are infinite, this approach works only for general satisfiability, and gives little insight into the decidability/complexity of finite satisfiability. In this paper we consider the finite satisfiability problem for $\text{UNFO} + S$. Actually, we made a step in this direction already in our previous paper [7] (see [8] for its longer version) where we proved a related result that UNFO with equivalence relations, $\text{UNFO} + \text{EQ}$, has the finite model property and thus that its satisfiability and finite satisfiability problems coincide, both being $2\text{-ExpTime}$-complete. Some ideas developed in [7] are extended and applied also here, even though $\text{UNFO} + S$ does not have the finite model property which becomes evident when looking at the following formula with transitive $T$, $\forall x \exists y Txy \land \forall x \neg Txx$, satisfiable only in infinite models.

Our main contribution is demonstrating the decidability of finite satisfiability for $\text{UNFO} + S$ and establishing its $2\text{-ExpTime}$-completeness. En route we obtain a triply exponential bound on the size of minimal models of finitely satisfiable $\text{UNFO} + S$ formulas. Actually, our results hold for a more general setting, in which some relations may be required to be interpreted as equivalences, some as partial orders, and some just as arbitrary transitive relations. Returning to database motivations, we get this way the decidability of the finite open-world query
answering for unions of conjunctive queries against frontier-one TGDs with equivalences, partial orders and arbitrary transitive relations. By finite open-world query answering we mean the question if for given $D$, $T$ and $q$, $D$ and $T$ entail $q$ over finite structures.

To the best of our knowledge, UNFO+ is the first logic which allows one to use arbitrarily many transitive relations, and, at the same time, to speak non-trivially about relations of arbitrary arities, whose finite satisfiability problem is shown decidable. In the case of related logics of this kind, like the guarded fragment with transitive guards [26], and the guarded negation fragment with transitive relations outside guards [11], the decidability was shown only for general satisfiability, and its finite version is open. (Finite satisfiability was shown decidable only for the two-variable guarded fragment with transitive guards [20]).

We believe that moving from UNFO+EQ from [7] to UNFO+ is an important improvement. Besides the fact that this requires strengthening our techniques and employing some new ideas, general transitive relations have stronger motivations than equivalences. In particular, it opens natural connections to the realm of description logics, DLs.

**UNFO and expressive description logics.** UNFO, via the above-mentioned standard translation, embeds the DL $ALC$, as well as its extension by inverse roles ($I$) and role intersections ($\sqcap$). Thus, having the ability of expressing conjunctive queries, we can use our results to solve the so-called (finite) ontology mediated query answering problem, (F)OMQA, for some DLs. This problem is a counterpart of (finite) open-world query answering: given a conjunctive query (or a union of conjunctive queries) and a knowledge base specified in a DL, check whether the query holds in every (finite) model of this knowledge base.

While there are quite a lot of results for OMQA, not much is known about FOMQA. In particular, for DLs with transitive roles ($S$) the only positive results we are aware of are the ones obtained recently in [12], where the decidability and 2-ExpTime-completeness of FOMQA for the logics $SOI$, $STF$ and $SOF$ is shown. This is orthogonal to our results described above, since UNFO+S captures neither nominals ($O$) nor functional roles ($F$). On the other hand, we are able to express any positive boolean combinations of roles, including their intersection ($\sqcap$), which allows us to solve FOMQA, e.g., for the logic $ST^{\sqcap}$. Moreover we can use non-trivially relations of arity greater than two.

It is an interesting question if our decidability result can be extended to capture some more expressive DLs. Unfortunately, we cannot hope for number restrictions ($Q$ or $N$) or even functional roles ($F$), as satisfiability and finite satisfiability of UNFO (even without transitive relations) and two binary functional relations are undecidable. This is implicit in [27] (see Appendix A for an explicit proof). On the positive side, we show the decidability and 2-ExpTime-completeness of finite satisfiability of UNFO+$SOH$, extending UNFO+S by constants (corresponding to nominals ($O$)) and inclusions of binary relations (capturing role hierarchies ($H$)). This is sufficient, in particular, to imply the decidability of FOMQA for the description logic $SHOT^{\sqcap}$, which, up to our knowledge, is a new result.

**Towards guarded negation fragment.** We propose also another decidable extension of our basic logic, the one-dimensional base-guarded negation fragment with transitive relations on non-guard positions, BGNFO1+S. This is a non-trivial fragment of the already mentioned logic from [11]. After some rather easy adjustments, our constructions cover this bigger logic, however, it becomes undecidable when extended with inclusions of binary relations.

**Organization of the paper.** The rest of this paper is organized as follows. Section 2 contains definitions, basic facts and a high-level description of our decidability proof. As our constructions are rather complex, in the main body of the paper, Section 3 we explicitly process the restricted, two-variable case of our logic, for which our ideas can be presented more transparently. In Section 4 we just formulate the remaining results, leaving the details
Finite Satisfiability of Unary Negation Fragment with Transitivity

for the Appendix, which also contains the missing proofs from Sections 2 and 3. In Section 5 we conclude the paper.

2 Preliminaries

2.1 Logics, structures, types and functions

We employ standard terminology and notation from model theory. We refer to structures using Fraktur capital letters, and their domains using the corresponding Roman capitals. For a structure $\mathfrak{A}$ and $\mathfrak{A}' \subseteq \mathfrak{A}$ we use $\mathfrak{A}[\mathfrak{A}']$ or $\mathfrak{A}'$ to denote the restriction of $\mathfrak{A}$ to $\mathfrak{A}'$.

The unary negation fragment of first-order logic, UNFO, is defined by the following grammar [27]: $\varphi = B \bar{x} | x = y | \varphi \land \varphi | \varphi \lor \varphi | \exists x \varphi | \neg \varphi(x)$, where, in the first clause, $B$ represents any relational symbol, and, in the last clause, $\varphi$ has no free variables besides (at most) $x$. An example formula not expressible in UNFO is $x \neq y$. We formally do not have universal quantification. However we allow ourselves to use $\forall \bar{x} \neg \varphi$ as an abbreviation for $\neg \exists \bar{x} \varphi$, for an UNFO formula $\varphi$. Note that frontier-one TGDs $\forall \bar{x} \bar{y} (\psi(\bar{x}, \bar{y}) \rightarrow \exists \bar{z} \psi'(y, \bar{z}))$ are in UNFO as they can be rewritten as $\neg \exists \bar{x} \bar{y} (\psi(\bar{x}, \bar{y}) \land \neg \exists \bar{z} \psi'(y, \bar{z}))$.

We mostly work with purely relational signatures (admitting constants only in some extensions of our main results) of the form $\sigma = \sigma_{\text{base}} \cup \sigma_{\text{dist}}$, where $\sigma_{\text{base}}$ is the base signature, and $\sigma_{\text{dist}}$ is the distinguished signature. We assume that $\sigma_{\text{dist}} = \{T_1, \ldots, T_{2k}\}$, with all the $T_u$ binary, and intension that $T_{2u}$ is interpreted as the inverse of $T_{2u-1}$. For every $1 \leq u \leq k$ we sometimes write $T_{2u-1}$ for $T_{2u-1}$, and $T_{2u-1}$ for $T_{2u}$. We say that a subset $\mathcal{E}$ of $\sigma_{\text{dist}}$ is closed under inverses if, for every $1 \leq u \leq 2k$, we have $T_u \in \mathcal{E}$ iff $T_u^{-1} \in \mathcal{E}$. Note that $\mathcal{E}$ is closed under inverses iff $\sigma_{\text{dist}} \mathcal{E}$ is closed under inverses. Given a formula $\varphi$ we denote by $\sigma_{\varphi}$ the signature induced by $\varphi$, i.e., the minimal signature, with its distinguished part closed under inverses, containing all symbols from $\varphi$.

The unary negation fragment with transitive relations, UNFO+$\mathcal{S}$, is defined by the same grammar as UNFO, however when satisfiability of its formulas is considered, we restrict the class of admissible models to those that interpret all symbols from $\sigma_{\text{dist}}$ as transitive relations and, additionally, for each $u$, interpret $T_{2u}$ as the inverse of $T_{2u-1}$. The latter condition is intended to simplify the presentation, and is imposed without loss of generality. In our constructions we sometimes consider some auxiliary structures in which symbols from $\sigma_{\text{dist}}$ are not necessarily interpreted as transitive relations (but the pairs $T_{2u-1}, T_{2u}$ are always interpreted as inverses of each other).

An (atomic) $k$-type over a signature $\sigma$ is a maximal satisfiable set of literals (atoms and negated atoms) over $\sigma$ with variables $x_1, \ldots, x_k$. We often identify a $k$-type with the conjunction of its elements. We are mostly interested in 1- and 2-types. Given a $\sigma$-structure $\mathfrak{A}$ and $a, b \in A$ we denote by $\text{atp}^\mathfrak{A}(a)$ the 1-type realized by $a$, that is the unique 1-type $\alpha(x_1)$ such that $\mathfrak{A} \models \alpha(a)$, and by $\text{atp}^\mathfrak{A}(a, b)$ the unique 2-type $\beta(x_1, x_2)$ such that $\mathfrak{A} \models \beta(a, b)$.

We use various functions in our paper. Given a function $f : A \to B$ we denote by $\text{Rng} f$ its range, by $\text{Dom} f$ its domain, and by $f|_{A_0}$ the restriction of $f$ to $A_0 \subseteq A$.

2.2 Normal form, witnesses and basic facts

We say that an UNFO+$\mathcal{S}$ formula is in Scott-normal form if it is of the shape

$$\forall x_1, \ldots, x_t \neg \varphi_0(\bar{x}) \land \bigwedge_{i=1}^{m} \forall x \exists y \varphi_i(x, y)$$

(1)

where each $\varphi_i$ is a UNFO+$\mathcal{S}$ quantifier-free formula and $\varphi_0$ is additionally in negation normal form (NNF). A similar normal form for UNFO was introduced in the bachelor’s
thesis \[9\]. By a straightforward adaptation of Scott’s translation for \(\text{FO}^2\) \[25\] one can translate in polynomial time any UNFO\(+\mathcal{S}\) formula to a formula in normal form, in such a way that both are satisfiable over the same domains. This allows us, when dealing with decidability/complexity issues for UNFO\(+\mathcal{S}\), or when considering the size of minimal finite models of formulas, to restrict attention to normal form formulas.

Given a structure \(\mathfrak{A}\), a normal form formula \(\varphi\) as in \([1]\) and elements \(a,\bar{b}\) of \(A\) such that \(\mathfrak{A} \models \varphi_i(a,\bar{b})\) we say that the elements of \(\bar{b}\) are \textit{witnesses} for \(a\) and \(\varphi_i\) and that \(\mathfrak{A}^i\{a,\bar{b}\}\) is a witness structure for \(a\) and \(\varphi_i\). Fix an element \(a\). For every \(\varphi_i\) choose a witness structure \(\mathfrak{M}_i\). Then the structure \(\mathfrak{M} = \mathfrak{A}^i\{W_1 \cup \ldots \cup W_m\}\) is called a \(\varphi\)-witness structure for \(a\).

We are going to present a construction which given an arbitrary finite model of a normal form UNFO\(+\mathcal{S}\) formula \(\varphi\) builds a finite model of \(\varphi\) of a bounded size. The construction goes via several intermediate steps in which some tree-like models are produced. To argue that they are still models of \(\varphi\) we use the following basic observation (we recall that \(t\) is the number of variables of the \(\forall\)-conjunct of \(\varphi\)).

\begin{lemma}
Let \(\mathfrak{A}\) be a model of a normal form UNFO\(+\mathcal{S}\) formula \(\varphi\). Let \(\mathfrak{A}'\) be a structure in which all symbols from \(\sigma_{\varphi}\) are interpreted as transitive relations, such that
\begin{enumerate}
\item[(a1)] for every \(a' \in A'\) there is a \(\varphi\)-witness structure for \(a'\) in \(\mathfrak{A}'\),
\item[(a2)] for every tuple \(a'_1, \ldots, a'_r \in A'\) there is a homomorphism \(h: \mathfrak{A}'^i\{a'_1, \ldots, a'_r\} \to \mathfrak{A}\) which preserves \(1\)-types of elements.
\end{enumerate}
Then \(\mathfrak{A}' \models \varphi\).
\end{lemma}

2.3 Plan of the small model construction

Our main goal is to show that finite satisfiability of UNFO\(+\mathcal{S}\) formulas can be checked in \(2\)-\text{ExpTime}. To this end we will introduce a natural notion of tree-like structures and a measure associating with transitive paths of such structures their so-called ranks. Intuitively, for a transitive relation \(T_i\) and a \(T_i\)-path \(\pi\), the \(T_i\)-rank of \(\pi\) is the number of one-directional \(T_i\)-edges in \(\pi\) (a precise definition is given in Section 3.1). Then we show that having the following forms of models is equivalent for a normal form formula \(\varphi\):

\begin{enumerate}
\item [(f1)] finite;
\item [(f2)] tree-like, with bounded ranks of transitive paths;
\item [(f3)] tree-like, with ranks of transitive paths bounded doubly exponentially in \(|\varphi|\);
\item [(f4)] tree-like, with ranks of paths bounded doubly exponentially in \(|\varphi|\), and regular (with doubly exponentially many non-isomorphic subtrees);
\item [(f5)] finite of size triply exponential in \(|\varphi|\).
\end{enumerate}

We will make the following steps: (f1) \(\Rightarrow\) (f2), (f2) \(\Rightarrow\) (f3), (f3) \(\Rightarrow\) (f4), (f4) \(\Rightarrow\) (f5). The step closing the circle, (f5) \(\Rightarrow\) (f1) is trivial. In the two-variable case, we will omit the form (f4) and directly show (f3) \(\Rightarrow\) (f5). Our \(2\)-\text{ExpTime}-algorithm will look for models of the form (f3). Showing transitions leading from (f3) to (f5) justifies that its answers coincide indeed with the existence of finite models.

This scheme is similar to the one we used to show the finite model property for UNFO\(+\mathcal{EQ}\) in \([7]\). In the main part of the construction from \([7]\) we build bigger and bigger substructures in which some equivalence relations are total. The induction goes, roughly speaking, by the number of non-total equivalences in the substructure. Here we extend this approach to handle one-way transitive connections. It may be useful to briefly compare the case of UNFO\(+\mathcal{S}\) and the case of UNFO\(+\mathcal{EQ}\).

First of all, if a given formula \(\varphi\) is from UNFO\(+\mathcal{EQ}\) then we can start our constructions leading to a small finite model of \(\varphi\) from its arbitrary model, while if \(\varphi\) is in UNFO\(+\mathcal{S}\) we
start from a finite model of $\varphi$. A very simple step (f1) $\rightsquigarrow$ (f2) in both papers is, essentially, identical. The counterpart of step (f3) $\rightsquigarrow$ (f4) in the case of equivalences is slightly simpler, but the main differences lie in steps (f2) $\rightsquigarrow$ (f3) and (f4) $\rightsquigarrow$ (f5). The former, clearly, is not present at all in [7]. While the general idea in this step is quite standard, as we just use a kind of tree pruning, the details are rather delicate due to possible interactions among different transitive relations, and this step is, by no means, trivial. We refine here, in particular, the apparatus of declarations introduced in [7]. Regarding step (f4) $\rightsquigarrow$ (f5), the main construction there, in its single inductive step, has two phases: building the so-called components and then arranging them into a bigger structure. It is this first phase which is more complicated than in the corresponding step in [7]. Having components prepared we join them similarly as in [7].

3 The two-variable case

As in the case of unbounded number of variables we can restrict attention to normal form formulas, which in the two-variable case simplify to the standard Scott-normal form [25]:

$$\forall xy \neg \varphi_0(x, y) \land \bigwedge_{i=1}^m \exists x \exists y \varphi_i(x, y),$$

where all $\varphi_i$ are quantifier-free UNFO$^2$+$S$ formulas (in this restricted case it is not important whether $\varphi_0$ is in NNF or not). As is typical for two-variable logics we assume that formulas do not use relational symbols of arity greater than 2 (cf. [14]).

3.1 Tree pruning in the two-variable case

We use a standard notion of a (finite or infinite) rooted tree and related terminology. Additionally, any set consisting of a node and all its children is called a family. Any node $b$, except for the root and the leaves, belongs to two families: the one containing its parent, and the one containing its children, the latter called the downward family of $b$.

We say that a structure $\mathfrak{A}$ over a signature consisting of unary and binary symbols is a light tree-like structure if its nodes can be arranged into a rooted tree in such a way that if $\mathfrak{A} \models Baa'$ for some non-transitive relation symbol $B$ then one of three conditions holds: $a = a'$, $a$ is the parent of $a'$ or $a$ is a child of $a'$, and if $\mathfrak{A} \models T_u a a'$ for some $T_u$ then either $a = a'$ or there is a sequence of distinct nodes $a = a_0, a_1, \ldots, a_k = a'$ such that $a_i$ and $a_{i+1}$ are joined by an edge of the tree and $\mathfrak{A} \models T_{u_i} a_i a_{i+1}$. In other words, distant nodes in a light tree-like structure can be joined only by transitive connections, moreover, these transitive connections are just the transitive closures of connections inside families. For a light tree-like structure $\mathfrak{A}$ and $a \in A$ we denote by $A_a$ the set of all nodes in the subtree rooted at $a$ and by $\mathfrak{A}_a$ the corresponding substructure.

Let $\mathfrak{A}$ be a light tree-like structure. A sequence of nodes $a_1, \ldots, a_N \in A$ is a downward path in $\mathfrak{A}$ if for each $i$ $a_{i+1}$ is a child of $a_i$. A downward-$T_u$-path is a downward path such that for each $i$ we have $\mathfrak{A} \models T_u a_i a_{i+1}$.

The $T_u$-rank of a downward-$T_u$-path $\vec{a}$, $\tau_u^\mathfrak{A}(\vec{a})$, is the cardinality of the set $\{i : \mathfrak{A} \models \neg T_{u_i} a_i a_{i+1}\}$. The $T_u$-rank of an element $a \in A$ is defined as $\tau_u^\mathfrak{A}(a) = \sup\{\tau_u^\mathfrak{A}(\vec{a}) : \vec{a} = a, a_2, \ldots, a_N; \vec{a} \text{ is a downward-$T_u$-path}\}$. For an integer $M$, we say that $\mathfrak{A}$ has downward-$T_u$-paths bounded by $M$ when for all $a \in A$ we have $\tau_u^\mathfrak{A}(a) \leq M$, and that $\mathfrak{A}$ has transitive paths bounded by $M$ if it has downward-$T_u$-paths bounded by $M$ for all $u$. Note that a downward-$T_u$-path bounded by $M$ may have more than $M$ nodes, as the symmetric $T_u$-connections do not increase the rank.
Given an arbitrary model \( \mathfrak{A} \) of a normal form UNFO\(^2\)+\( S \) formula \( \varphi \) we can simply construct its light tree-like model of degree bounded by \( |\varphi| \). We define a light-\( \varphi \)-tree-like unraveling \( \mathfrak{A}' \) of \( \mathfrak{A} \) and an associated function \( h : A' \to A \) in the following way. \( \mathfrak{A}' \) is divided into levels \( L_0, L_1, \ldots \). Choose an arbitrary element \( a \in A \) and add to level \( L_0 \) of \( A' \) an element \( a' \) such that \( \text{atp}^\mathfrak{A}(a') = \text{atp}^\mathfrak{A}(a) \); set \( h(a') = a \). The element \( a' \) will be the only element of \( L_0 \) and will become the root of \( \mathfrak{A}' \). Having defined \( L_i \), repeat the following for every \( a' \in L_i \). For every \( j \), if \( h(a') \) is not a witness for \( \varphi_j \) and itself then choose in \( \mathfrak{A} \) a witness \( b \) for \( h(a') \) and \( \varphi_j \). Add a fresh copy \( b' \) of \( b \) to \( L_{i+1} \), make \( \mathfrak{A}'\{a', b'\} \) isomorphic to \( \mathfrak{A}\{h(a'), b\} \) and set \( h(b') = b \). Complete the definition of \( \mathfrak{A}' \) transitively closing all relations from \( \sigma_{\text{init}} \).

**Lemma 2** ((f1) \( \to \) (f2), light). Let \( \mathfrak{A} \) be a finite model of a normal form UNFO\(^2\)+\( S \) formula \( \varphi \). Let \( \mathfrak{A}' \) be a light-\( \varphi \)-tree-like unraveling of \( \mathfrak{A} \). Then \( \mathfrak{A}' \models \varphi \) and \( \mathfrak{A}' \) is a light tree-like structure of degree bounded by \( |\varphi| \), and transitive paths bounded by \( |A| \).

Our next task is making the transition (f2) \( \to \) (f3). For this purpose we introduce a notion of light declarations. It is closely related to a notion of declarations which will be used in the general case, but simpler than the latter. Fix a signature and let \( \alpha \) be the set of 1-types over this signature.

For \( T \subseteq \{T_1, \ldots, T_{2k}\} \) we write \( \mathfrak{A} \models T ab \) iff \( \mathfrak{A} \models T_u ab \) for all \( T_u \in T \). A light declaration is a function of type \( \mathcal{P}((T_1, \ldots, T_{2k})) \to \mathcal{P}(\alpha) \). Given a light tree-like structure \( \mathfrak{A} \) and its node \( a \) we say that \( a \) respects a light declaration \( \delta \) if for every \( T \), for every \( a \in \delta(T) \) there is no node \( b \in A \) of 1-type \( a \) such that \( \mathfrak{A} \models T ab \). We denote by \( \text{ldec}^\mathfrak{A}(a) \) the maximal light declaration respected by \( a \). Formally, for every \( T \subseteq \{T_1, \ldots, T_{2k}\} \), \( \text{ldec}^\mathfrak{A}(a)(T) = \{ \alpha : \text{for every node } b \text{ of type } a \text{ we have } \neg \mathfrak{A} \models T ab \} \). Intuitively, \( \text{ldec}^\mathfrak{A}(a) \) says, for any combination of transitive relations, which 1-types have no realizations to which \( a \) is connected by this combination in \( \mathfrak{A} \). Note that if \( a \) respects a light declaration \( \delta \) then for any \( T \) we have \( \delta(T) \subseteq \text{ldec}^\mathfrak{A}(a)(T) \). We remark that it would be equivalent to define the light declarations without the negations, listing the 1-types that a given node is connected with, however we choose a version with negations to make them uniform with the corresponding (more complicated) notion in the general case, where negations are more convenient.

Now we define the local consistency conditions (LCCs) for a system of light declarations \( (\delta_a)_{a \in A} \) assigned to all nodes of a tree-like structure \( \mathfrak{A} \). Let \( F \) be the downward family of some node \( a \). We say that the system satisfies LCCs at \( a \) if for every \( a_1, a_2 \in F \) and for every \( T \) such that \( \mathfrak{A} \models T a_1 a_2 \) the following two conditions hold: (1d1) for every \( a \in \alpha \), if \( a \in \delta_{a_1}(T) \) then \( a \in \delta_{a_2}(T) \), (1d2) \( \text{atp}^\mathfrak{A}(a_2) \notin \delta_{a_1}(T) \). Given a light tree-like structure \( \mathfrak{A} \) we say that a system of light declarations \( (\delta_a)_{a \in A} \) is locally consistent if it satisfies LCCs at each \( a \in A \) and is globally consistent if \( \delta_a(T) \subseteq \text{ldec}^\mathfrak{A}(a)(T) \) for each \( a \in A \) and each \( T \). Note that the global consistency means that all nodes \( a \) respect their light declarations \( \delta_a \). It is not difficult to see that local and global consistency play along in the following sense.

**Lemma 3** (Local-global, light). Let \( \mathfrak{A} \) be a light tree-like structure. Then, (i) if a system of light declarations \( (\delta_a)_{a \in A} \) is locally consistent then it is globally consistent; and (ii) the canonical system of light declarations, \( (\text{ldec}^\mathfrak{A}(a))_{a \in A} \), is locally consistent.

Given a light tree-like structure \( \mathfrak{A} \), by the generalized type of a node \( a \) of \( \mathfrak{A} \) we will mean a pair \( (\text{ldec}^\mathfrak{A}(a), \text{atp}^\mathfrak{A}(a)) \), and denote it as \( \text{gtp}^\mathfrak{A}(a) \). We introduce a concept of top-down tree pruning. Let \( \mathfrak{A} \) be a light tree-like structure. A top-down tree pruning process on \( \mathfrak{A} \) has countably many steps 0, 1, 2, \ldots, each of them producing a new light tree-like structure by removing some nodes from the previous one and naturally stitching together the surviving
nodes. We emphasise that the universes of all structures build in this process are subsets of the universe of the original structure $\mathfrak{A}$. More specifically, we take $\mathfrak{A}_0 := \mathfrak{A}$, and having constructed $\mathfrak{A}_i$, $i \geq 0$ construct $\mathfrak{A}_{i+1}$ as follows. For every node $a$ of $\mathfrak{A}_i$ of depth $i + 1$ (we assume that the root has depth 0) either leave the subtree rooted at $a$ untouched or replace it by a subtree rooted at some descendant $b$ of $a$ having in the original structure $\mathfrak{A}$ the same generalised type as $a$, and then transitively close all transitive relations. The result of the process is a naturally defined limit structure $\mathfrak{A}'$, in which the pair of elements $a, b$, of depth $d_a$ and $d_b$ respectively, has its 2-type taken from $\mathfrak{A}_{\text{max}(d_a,d_b)}$. Note that this 2-type is not modified in the subsequent structures, so the definition is sound.

**Lemma 4** (Tree-pruning, light). Let $\mathfrak{A}$ be a light tree-like structure. Let $(\vartheta_a)_{a \in A}$ be the canonical system of light declarations on $\mathfrak{A}$, $\vartheta_a := \text{ldec}^\mathfrak{A}(a)$. Let $\mathfrak{A}'$ be the result of a top-town tree pruning process on $\mathfrak{A}$. Then (i) the system of light declarations $(\vartheta_a)_{a \in A'}$ (the canonical declarations from $\mathfrak{A}$ of the nodes surviving the pruning process) in $\mathfrak{A}'$ is locally consistent, (ii) for any pair of elements $a,a' \in A'$ there is a homomorphism $\mathfrak{A}'[\{a,a'\}] \to A$ preserving the 1-types; it also follows that (iii) for a normal form $\varphi$, if $\mathfrak{A}$ is a model of $\varphi$ such that any node $a$ has all its witnesses in its downward family then $\mathfrak{A}' \models \varphi$.

It is not difficult to devise a strategy of top-down tree pruning leading to a model with short transitive paths in a simple scenario where only one transitive relation is present. With several transitive relations, however, a quite intricate strategy seems to be required. The main obstacle is that when decreasing the $T_u$-rank of an element $a$, for some $u$, we may accidentally increase the $T_v$-rank of $a$ for some $v \neq u$. Nevertheless, an appropriate strategy exists (see Appendix D.4), which allows us to state:

**Lemma 5** ((f2) $\rightsquigarrow$ (f3), light). Let $\varphi$ be a normal form UNFO$^2 + S$ formula. Let $\mathfrak{A} \models \varphi$ be a light tree-like structure over signature $\sigma_\varphi$, with transitive paths bounded by some natural number $M$, such that each element has all the required witnesses in its downward family. Then $\varphi$ has a light tree-like model with transitive paths bounded doubly exponentially in $|\varphi|$.

### 3.2 Finite model construction in the two-variable case

In this section we show the following small model property. To this end, in particular, we will make the transition (f3) $\rightsquigarrow$ (f5).

**Theorem 6.** Every finitely satisfiable two-variable UNFO+$S$ formula $\varphi$ has a finite model of size bounded triply exponentially in $|\varphi|$.

Let us fix a finitely satisfiable normal form UNFO+$S$ formula $\varphi$ over a signature $\sigma_\varphi = \sigma_{\text{base}} \cup \sigma_{\text{dist}}$ for $\sigma_{\text{dist}} = \{T_1, \ldots, T_{2k}\}$. Denote by $\alpha$ the set of 1-types over this signature. Fix a light tree-like model $\mathfrak{A} \models \varphi$, with linearly bounded degree and doubly exponentially bounded transitive paths (in this section we denote this bound by $M_\varphi$), as guaranteed by Lemma 5. We show how to build a ‘small’ finite model $\mathfrak{A}' \models \varphi$. For a set $E \subseteq \sigma_{\text{dist}}$, closed under inverses, and $a \in A$ we denote by $[a]_E$ the set consisting of $a$ and all elements $b \in A$ such that $\mathfrak{A} \models T_u a b$ for all $T_u \in E$. Note that $[a]_E$ is either a singleton or each of the $T_u \in E$ is total on $[a]_E$, that is, for each $b_1, b_2 \in [a]_E$ we have $\mathfrak{A} \models T_u b_1 b_2$ for all $T_u \in E$. We note that $[a]_0 = A$.

In our construction we inductively produce finite fragments of $\mathfrak{A}'$ corresponding to some (potentially infinite) classes $[a]_E$ of $\mathfrak{A}$. Essentially, the induction goes downward on the size of $E$. Intuitively, if a relation is total then it plays no important role, so we may forget about it during the construction. Every such fragment will be obtained by an appropriate arrangement
Let us formally state our inductive lemma. In this statement we do not explicitly include any bound on the size of promised finite models, but such a bound will be implicit in the proof and will be presented later. Recall that \( \mathfrak{A} \) is the model fixed at the beginning of this subsection.

**Lemma 7** (Main construction, light). Let \( a_0 \in A \) and let \( E_0 \subseteq \sigma_\text{aut} \) be closed under inverses, let \( E_\text{tot} := \sigma_\text{aut} \setminus E_0 \). Let \( \mathfrak{A}_0 = \mathfrak{A}_{a_0}|[a_0]_{E_\text{tot}} \). Then there exist a finite structure \( \mathfrak{A}_0' \), a function \( p : A_0' \rightarrow A_0 \) and an element \( a_0' \in A_0' \), called the origin of \( \mathfrak{A}_0' \), such that

1. \( A_0' \) is a singleton or every symbol from \( E_\text{tot} \) is interpreted as the total relation on \( \mathfrak{A}_0' \).
2. \( p(a_0') = a_0 \).
3. For each \( a' \in A_0' \) and each \( i \), if \( p(a') \) has a child being its witness for \( \varphi_i \) in \( \mathfrak{A}_0 \) then \( a' \) has a witness for \( \varphi_i \) in \( \mathfrak{A}_0' \). Moreover, \( l_{\text{dec}}(a') = l_{\text{dec}}(p(a')) \).
4. For every pair \( a', b' \in A_0' \) there exists a homomorphism \( h : \mathfrak{A}_0'[\{a', b\}] \rightarrow \mathfrak{A} \) preserving 1-types such that \( h(a') = p(a') \), and for any 1-type \( \alpha \) and \( T \subseteq \{1, \ldots, 2k\} \), if \( \mathfrak{A}_0' \models T \alpha b' \) and \( \alpha \notin l_{\text{dec}}(p(b'))(T) \) then \( \alpha \notin l_{\text{dec}}(p(a'))(T) \).

Observe first that Lemma 7 indeed allows us to build a particular finite model of \( \varphi \). Apply it to \( E_0 = \sigma_\text{aut} \) (which means that \( E_\text{tot} = \emptyset \) and \( [a_0]_{E_\text{tot}} = A_0 \)) and \( a_0 \) being the root of \( \mathfrak{A} \) (which means that \( \mathfrak{A}_0 = \mathfrak{A} \)) and use Lemma 7 to see that the obtained structure \( \mathfrak{A}_0' \) is a model of \( \varphi \). Indeed, Condition (a1) of Lemma 7 follows directly from Condition (b3), as in this case \( p(a') \) has all witnesses in \( \mathfrak{A}_0' \). Condition (a2) is directly implied by Condition (b3).

The proof of Lemma 7 goes by induction on \( l \), where \( l = |E_0|/2 \). In the base of induction, \( l = 0 \), we have \( E_\text{tot} = \sigma_\text{aut} \). Without loss of generality we may assume that the classes \( [a]_{E_\text{tot}} \) are singletons for all \( a \in A \). (If this is not the case, we just add artificial transitive relations \( T_{2k+1} \) and \( T_{2k+2} \) both interpreted as the identity in \( \mathfrak{A} \).) We simply take \( \mathfrak{A}_0' := \mathfrak{A}_0 = \mathfrak{A}' \{a_0\} \) and set \( p(a_0) = a_0 \). It is readily verified that the conditions (1) and (4) are then satisfied.

For the inductive step assume that Lemma 7 holds for arbitrary \( E_0 \) closed under inverses of size \( 2(l-1) < 2k \). We show that then it holds for \( E_0 \) of size \( 2l \). Take such \( E_0 \), and assume, w.l.o.g., that \( E_0 = \{T_1, \ldots, T_2\} \). In the next two subsections we present a construction of \( \mathfrak{A}_0' \).

We argue that it is correct in Appendix D.6 Finally we estimate the size of the produced models and establish the complexity of the finite satisfiability problem.

### 3.2.1 Pattern components

We plan to construct \( \mathfrak{A}_0' \) out of basic building blocks called *components*. Each component will be an isomorphic copy of some pattern component.

Let \( \gamma[A_0] \) be the set of the generalized types realized in \( \mathfrak{A}_0 \). For every \( \gamma \in \gamma[A_0] \) we construct two pattern structures, a *pattern component* \( \mathfrak{C}^\gamma \) and an *extended pattern component* \( \mathfrak{G}^\gamma \). \( \mathfrak{C}^\gamma \) is a finite structure whose universe is divided into \( 2l \) layers \( L_1, \ldots, L_{2l} \). \( \mathfrak{G}^\gamma \) extends \( \mathfrak{C}^\gamma \) by an additional, *interface layer*, denoted \( L_{2l+1} \). See the left part of Fig. 4. We now define \( \mathfrak{G}^\gamma \), obtaining then \( \mathfrak{C}^\gamma \) just by the restriction of \( \mathfrak{G}^\gamma \) to non-interface layers.

Each non-interface layer \( L_i \) is further divided into *sublayers* \( L_i^1, L_i^2, \ldots, L_i^{M_i+1} \). Additionally, in each sublayer \( L_i^1 \) its initial part \( L_i^{1,\text{init}} \) is distinguished. In particular, \( L_i^{1,\text{init}} \) consists of a single element called the *root*. The interface layer \( L_{2l+1} \) has no internal division but, for convenience, is sometimes referred to as \( L_{2l+1}^{1,\text{init}} \). The elements of \( L_{2l} \) are called *leaves* and the elements of \( L_{2l+1} \) are called *interface elements*. See Fig. 4.
\[ \mathcal{G}^\gamma \] will have a shape resembling a tree, with structures obtained by the inductive assumption as nodes, though it will not be tree-like in the sense of Section 3.1 (in particular, the internal structure of nodes may be complicated). All elements of \( \mathcal{G}^\gamma \), except for the interface elements, will have appropriate witnesses (those required by (H3) provided. The crucial property we want to enforce is that the root of \( \mathcal{G}^\gamma \) will not be joined to its interface elements by any transitive path.

We remark that during the process of building a pattern component we do not yet apply the transitive closure to the distinguished relations. Postponing this step is not important from the point of view of the correctness of the construction, but will allow us for a more precise presentation of the proof of this correctness. Given a component \( \mathcal{C} \) (extended component \( \mathcal{G} \)) we will sometimes denote by \( \mathcal{C}_+ \) (\( \mathcal{G}_+ \)) the structure obtained from \( \mathcal{C} \) (\( \mathcal{G} \)) by applying all the appropriate transitive closures.

The role of every non-interface layer \( L_u \) is, speaking informally, to kill \( T_u \), that is to ensure that there will be no \( T_u \)-connections from \( L_u \) to \( L_{u+1} \). See the right part of Fig. 1. The role of sublayers of \( L_u \), on the other hand, is to decrease the \( T_u \)-rank of the patterns of elements. The purpose of the interface layer, \( L_{2l+1} \), will be to connect the component with other components.

If \( \gamma \) is the generalized type of \( a_0 \) then take \( a := a_0 \); otherwise take as \( a \) any element of \( A_0 \) of generalized type \( \gamma \). We begin the construction of \( \mathcal{G}^\gamma \) by defining \( L_1^{\text{init}} = \{a'\} \) for a fresh \( a' \), setting \( \text{atp} \mathcal{G}^\gamma(a') = \text{atp} \mathcal{G}(a) \) and \( \text{p}(a') = a \).

**Construction of a layer:** Let \( 1 \leq u \leq 2l \). Assume we have defined layers \( L_1, \ldots, L_{u-1} \), the initial part of sublayer \( L_u^1 \), \( L_{u, \text{init}}^1 \), and both the structure of \( \mathcal{G}^\gamma \) and the values of \( \text{p} \) on \( L_1 \cup \ldots \cup L_{u-1} \cup L_{u, \text{init}}^1 \). We are going to kill \( T_u \). We now expand \( L_{u, \text{init}}^1 \) to a full layer \( L_u \).

**Step 1:** Subcomponents. Assume that we have defined sublayers \( L_{u, \text{init}}^1, \ldots, L_{u, \text{init}}^2 \), and both the structure of \( \mathcal{G}^\gamma \) and the values of \( \text{p} \) on \( L_1 \cup \ldots \cup L_{u-1} \cup L_1 \cup \ldots \cup L_{u, \text{init}}^2 \). For each \( b \in L_{u, \text{init}}^2 \) perform independently the following procedure. Apply the inductive assumption to \( \text{p}(b) \) and the set \( \mathcal{E}_0 \setminus \{T_u, T_u^{-1}\} \) obtaining a structure \( \mathcal{B}_0 \), its origin \( b_0 \) and a function \( \text{p}_b : B_0 \to A_0 \cap \{b\} \) \( \mathcal{E}_0 \setminus \{T_u, T_u^{-1}\} \subseteq A_0 \) with \( \text{p}_b(b_0) = \text{p}(b) \). Identify \( b_0 \) with \( b \) and add the remaining elements of \( \mathcal{B}_0 \) to \( L_u \), retaining the structure. Substructures \( \mathcal{B}_0 \) of this kind will be called subcomponents (note that all appropriate relations are transitively closed in subcomponents). Extend \( \text{p} \) so that \( \text{p}B_0 = \text{p}_b \). This finishes the definition of \( L_u \).

**Step 2:** Providing witnesses. For each \( b \in L_u \) and \( 1 \leq s \leq m \) independently perform the following procedure. Let \( \mathcal{B}_0 \) be the subcomponent created inductively in Step 1, such that \( b \in B_0 \). If \( \text{p}(b) \) has a witness for \( \varphi_s(x, y) \) in \( A_0 \) then we want to reproduce such a witness for \( b \). Choose one such witness \( c \) (being a child of \( \text{p}(b) \) for \( \text{p}(b) \)). Let us denote \( \beta = \text{atp} \mathcal{G}(\text{p}(b), c) \). If \( \{T_u xy, T_u^{-1} xy\} \subseteq \beta \) then by Condition (H3) of the inductive assumption \( b \) already has an appropriate witness in the subcomponent \( \mathcal{B}_0 \). So we do nothing in this case. If \( T_u xy \in \beta \)
and \( T_u^{-1} xy \notin \beta \) then we add a copy \( c' \) of \( c \) to \( L_u^{i+1,\text{init}} \); if \( T_u xy \notin \beta \) then we add a copy \( c' \) of \( c \) to \( L_u^{i,\text{init}} \). We join \( b \) with \( c' \) by \( \beta \) and set \( p(c') = c \).

An attentive reader may be afraid that when adding witnesses for elements of the last sublayer \( L_u^{\hat{M}_u+1} \) of \( L_u \) we may want to add one of them to the non-existing layer \( L_u^{\hat{M}_u+2} \). There is however no such danger, which follows from the following claim.

 большим отрезком

Claim 8. (i) Let \( b \in L_u^{i,\text{init}} \) and let \( \mathcal{B}_b \) be the subcomponent created for \( b \) in Step 1. Then for all \( b' \in \mathcal{B}_b \) we have \( r_0^0(b) \geq r_0^0(b') \). (ii) Let \( b \in L_u^i \) and let \( c' \in L_u^{\hat{M}_u+1} \) be a witness created for \( b \) in Step 2. Then \( r_0^0(b) > r_0^0(c') \).

Hence, when moving from \( L_u^i \) to \( L_u^{\hat{M}_u+1} \) the \( T_u \)-ranks of pattern elements for the elements of these sublayers strictly decrease. Since these ranks are bounded by \( \hat{M}_u \), then, even if the \( T_u \)-ranks of the patterns of some elements of \( L_u^i \) are equal to \( \hat{M}_u \), then, if \( L_u^{\hat{M}_u+1} \) is non-empty, the \( T_u \)-ranks of the patterns of its elements must be \( 0 \), which means that they cannot have witnesses connected to them one-directionally by \( T_u \).

The construction of \( \mathcal{G}^\gamma \) is finished when layer \( L_{2i} \) is fully processed. We have added some elements to the interface layer, \( L_{2i+1} \). Recall that it has only its ‘initial part’.

### 3.2.2 Joining the components

In this section we take some number of copies of pattern components and arrange them into the desired structure \( \mathcal{A}_0^0 \), identifying interface elements of some components with the roots of some other. Some care is needed in this process in order to avoid any modifications of the internal structure of closures \( \mathcal{C}_c \) of components \( \mathcal{C} \), which could potentially result from the transitivity of relations. In particular we need to ensure that if for some \( u \) a pair of elements of a component \( \mathcal{C} \) is not connected by \( T_u \) inside \( \mathcal{C} \), then it will not become connected by a chain of \( T_u \)-edges external to \( \mathcal{C} \).

We create a pattern component \( \mathcal{C}_c^\gamma \) and its extension \( \mathcal{G}^\gamma \) for every \( \gamma \in \gamma[\mathcal{A}_0] \). Let \( \gamma_{\alpha_0} \) be the generalized type of \( \alpha_0 \). Let \( \text{max} \) be the maximal number of interface elements across all the \( \mathcal{G}^\gamma \). For each \( \mathcal{G}^\gamma \) arbitrarily number its interface elements from 1 up to, maximally, \( \text{max} \).

For each \( \gamma \) we take copies \( \mathcal{G}^\gamma_{\gamma_{\alpha}} \) of \( \mathcal{G}^\gamma \) for \( g \in \{0, 1\} \), \( 1 \leq i \leq \text{max} \) and \( \gamma' \in \gamma[\mathcal{A}_0] \). The parameter \( g \) is sometimes called a color (red or blue); it is convenient to think that the non-interface elements of \( G^\gamma_{\gamma,\gamma'} \) are of color \( g \), but its interface elements have color \( 1-g \), cf. the left part of Fig. 1 as the latter will be later identified with the roots of some components of color \( 1-g \). We import the numbering of the interface elements to these copies. We also take an additional copy \( \mathcal{G}^\gamma_{\gamma_{\alpha},0} \) of \( \mathcal{G}^\gamma_{\gamma_{\alpha}} \). Its root will become the origin of the whole \( \mathcal{A}_0^0 \). By \( \mathcal{G}^\gamma_{\gamma_{\alpha},\gamma'_{\alpha}} \) we denote the restriction of \( \mathcal{G}^\gamma_{\gamma_{\alpha}} \) to its non-interface elements.

For each \( \gamma, g \) consider extended components of the form \( \mathbf{G}^\gamma_{\gamma,\gamma'} \), where the placeholders \( \cdot \) can be substituted with any combination of proper indices. Perform the following procedure for each \( 1 \leq i \leq \text{max} \). Let \( b \) be the \( i \)-th interface element of any such extended component, let \( \gamma' \) be the generalized type of \( p(b) \). Identify the \( i \)-th interface elements of all \( \mathcal{G}^\gamma_{\gamma,\gamma'} \) with the root \( c_0 \) of \( \mathcal{G}^\gamma_{1,\gamma,\gamma-1} \). Note that the values of \( p(c_0) \) and \( p(b) \) may differ. However, by construction, they have identical generalized types \( \gamma' \). For the element \( c^* \) obtained in this identification step we define \( p(c^*) = p(c_0) \).

Define the graph of components used in the above construction, \( G^\text{comp} \), by joining two components by an edge iff we identified an interface element of the extended version of one of them with the root of the other. Let \( \mathcal{A}_0^1 \) be the union of the components accessible from \( \mathcal{G}^\gamma_{\gamma_{\alpha},0} \) in \( G^\text{comp} \) and let \( \mathcal{A}_0^0 \) be the induced structure. Note that in \( \mathcal{A}_0^0 \) we still do not take the transitive closures of relations. We define \( \mathcal{A}_0^1 \) by transitively closing all relations from \( \sigma_{\text{init}} \) in \( \mathcal{A}_0^0 \). Finally, we choose as the origin \( \alpha_0' \) of \( \mathcal{A}_0^0 \) the root of the pattern component \( \mathcal{C}^\gamma_{\gamma_{\alpha},0} \).
We remark that it is sufficient to take as the universe of $A'$ the union of the universes of some components $C_i$, and not of their extended versions $G_i$, from which we started our construction, since the interface elements from these extended components were identified with some roots of other components.

For the correctness proof of our construction see Appendix D.6. In this proof it is helpful to think about $A_0$ and $A'$ as the structures placed on a cylindrical surface and divided into $4l$ levels, see Fig. 2. What is crucial, any transitive path in $A_0$ can cross at most one of the two borders between colors.

3.2.3 Size of models and complexity

By a rather routine calculation we can show that models produced in the proof of Lemma 7 are of size bounded triply exponentially in the length of input formulas. This finishes the proof of Thm. 6, which immediately gives the decidability of the finite satisfiability problem for UNFO$^2$+$S$ and suggests a simple 3-NExpTime-procedure: guess a finite structure of size bounded triply exponentially in the size of input $\varphi$ and verify that it is indeed a model of $\varphi$. We can however do better and show a doubly exponential upper bound matching the known complexity of the general satisfiability problem. For this we design an alternating exponential space algorithm searching for models of the form (f3). The lower bound can be obtained for the two-variable UNFO$^2$+$S$ in the presence of one transitive relation by a straightforward adaptation of the lower bound proof for GF$^2$ with transitive guards [19].

(Theorem 9. The finite satisfiability problem for UNFO$^2$+$S$ is 2-ExpTime-complete.

4 The general case and its further extensions

In Appendix E we generalize the ideas from Section 3 to show:

(Theorem 10. The finite satisfiability problem for UNFO+$S$ is 2-ExpTime-complete.

We also obtain a triply exponential upper bound on the size of minimal finite models of finitely satisfiable formulas. The structure of the proofs is similar to the two-variable case, though some details are more complicated. In particular, we need to go through form (f4) of models: regular trees with bounded ranks of transitive paths. We also explain that in addition to general transitive relations we can use also equivalences and partial orders.

In Appendix F we further extend Thm. 10 by considering an extension, UNFO+$SOH$, of UNFO+$S$ by constants and inclusion of binary relations of the form $B_1 \subseteq B_2$, interpreted in a natural way: $\mathfrak{A} \models B_1 \subseteq B_2$ iff $\mathfrak{A} \models \forall xy (B_1 xy \rightarrow B_2 xy)$.

(Theorem 11. The finite satisfiability problem for UNFO+$SOH$ is 2-ExpTime-complete.
As mentioned in the Introduction, UNFO+$SOH$ captures several interesting description logics. This implies that we can solve FOMQA problem for them. In particular, we have the following corollary, which, up to our knowledge is the first decidability result for FOMQA in the case of a description logic with both transitive roles and role hierarchies.

**Corollary 12.** Finite ontology mediated query answering, FOMQA, for the description logic $SHOT^{\ominus}$ is decidable and $2$-$\text{ExpTime}$-complete.

$SHOT^{\ominus}$ and some related logics are considered, e.g., in [11]. For more about FOMQA for description logics with transitivity see [12]. For more about OMQA for description logics see, e.g., references in [12].

Somewhat orthogonally to the extensions motivated by description logics in Appendix G we consider the base-guarded negation fragment with transitivity, BGNFO+$\mathcal{S}$, for which the general satisfiability problem was shown decidable in [1]. We do not solve its finite satisfiability problem here, but, analogously to the extension with equivalence relations, UNFO+$EQ$ [7], we are able to lift our results to its one-dimensional restriction, BGNFO$_1$+$\mathcal{S}$, admitting only formulas in which every maximal block of quantifiers leaves at most one variable free.

**Theorem 13.** The finite satisfiability problem for BGNFO$_1$+$\mathcal{S}$ is $2$-$\text{ExpTime}$-complete.

Surprisingly, in contrast to UNFO+$\mathcal{S}$, BGNFO$_1$+$\mathcal{S}$ becomes undecidable when extended by inclusions of binary relations.

## 5 Conclusions

We proved that the finite satisfiability problem for the unary negation fragment with transitive relations, UNFO+$\mathcal{S}$, is decidable and $2$-$\text{ExpTime}$-complete, complementing this way the analogous result for the general satisfiability problem for this logic implied by two other papers. Further, we identified some decidable extensions of our base logic capturing the concepts of nominals and role hierarchies from description logics. We noted that our work has some interesting implications on the finite query answering problem both under the classical (open-world) database scenario as well as in the description logics setting.

One open question is the decidability of the finite satisfiability problem for the full logic BGNFO+$\mathcal{S}$ from [1]. We made a step in this direction here, by solving this problem for the one-dimensional restriction of that logic. Another question is if our techniques can be adapted to a setting in which we do not assert that some distinguished relations are transitive but where we can talk about the transitive closure of the binary relations, or, more generally, to the extension of UNFO with regular path expressions from [16].

We finally remark that we do not know if our small model construction, producing finite models of size bounded triply exponentially in the size of the input formulas, is optimal with respect to the size of models. The best we can do for the lower bound is to enforce models of doubly exponential size (actually, this can be done in UNFO even without transitive relations).

### References

1. A. Amarilli, M. Benedikt, P. Bourhis, and M. Vanden Boom. Query answering with transitive and linear-ordered data. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence, IJCAI 2016*, pages 893–899, 2016.
Finite Satisfiability of Unary Negation Fragment with Transitivity

2 H. Andréka, J. van Benthem, and I. Németi. Modal languages and bounded fragments of predicate logic. *Journal of Philosophical Logic*, 27:217–274, 1998.

3 J.-F. Baget, M. LeClère, and M.-L. Mugnier. Walking the decidability line for rules with existential variables. In *Proceedings of the 12th International Conference on Principles of Knowledge Representation and Reasoning, KR 2010*, pages 466–476, 2010.

4 J.-F. Baget, M. Leclère, M.-L. Mugnier, and E. Salvat. Extending decidable cases for rules with existential variables. In *Proceedings of the 21st International Joint Conference on Artificial Intelligence, IJCAI 2009*, pages 677–682, 2009.

5 V. Bárány, B. ten Cate, and L. Segoufin. Guarded negation. *J. ACM*, 62(3):22, 2015.

6 A. Calì, G. Gottlob, and M. Kifer. Taming the infinite chase: Query answering under expressive relational constraints. *J. Artif. Intell. Res.*, 48:115–174, 2013.

7 D. Danielski and E. Kieronski. Unary negation fragment with equivalence relations has the finite model property. In *33rd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2018*, pages 285–294, 2018.

8 D. Danielski and E. Kieronski. Unary negation fragment with equivalence relations has the finite model property. *CoRR*, abs/1802.01318, 2018. [arXiv:1802.01318](http://arxiv.org/abs/1802.01318).

9 M. Dzieciołowski. Satisfiability issues for unary negation logic. Bachelor’s thesis, University of Wrocław, 2017.

10 H. Ganzinger, Ch. Meyer, and M. Veanes. The two-variable guarded fragment with transitive relations. In *14th Annual IEEE Symposium on Logic in Computer Science, LICS 1999*, pages 24–34, 1999.

11 B. Glimm and Y. Kazakov. Role conjunctions in expressive description logics. In *Logic for Programming, Artificial Intelligence, and Reasoning, 15th International Conference, LPAR 2008*, pages 391–405, 2008.

12 T. Gogacz, Y. A. Ibáñez-García, and F. Murlak. Finite query answering in expressive description logics with transitive roles. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Sixteenth International Conference, KR 2018*, pages 369–378, 2018.

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

14 E. Grädel, P. Kolaitis, and M. Y. Vardi. On the decision problem for two-variable first-order logic. *Bulletin of Symbolic Logic*, 3(1):53–69, 1997.

15 E. Grädel, M. Otto, and E. Rosen. Undecidability results on two-variable logics. *Archive für Mathematische Logik und Grundlagenforschung*, 38(4-5):313–354, 1999.

16 J. Ch. Jung, C. Lutz, M. Martel, and T. Schneider. Querying the unary negation fragment with regular path expressions. In *International Conference on Database Theory, ICDT 2018*, pages 15:1–15:18, 2018.

17 Y. Kazakov. *Saturation-based decision procedures for extensions of the guarded fragment*. PhD thesis, Universität des Saarlandes, Saarbrücken, Germany, 2006.

18 E. Kieronski. Results on the guarded fragment with equivalence or transitive relations. In *Computer Science Logic*, volume 3634 of LNCS, pages 309–324. Springer, 2005.

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

20 E. Kieronski and L. Tendera. Finite satisfiability of the two-variable guarded fragment with transitive guards and related variants. *ACM Trans. Comput. Logic*, 19(2):8:1–8:34, 2018.

21 I. Pratt-Hartmann. The finite satisfiability problem for two-variable, first-order logic with one transitive relation is decidable. *Mathematical Logic Quarterly*, 2018.

22 I. Pratt-Hartmann, W. Szaszt, and L. Tendera. The fluted fragment revisited. *Journal of Symbolic Logic*, Forthcoming, 2019.

23 I. Pratt-Hartmann and L. Tendera. The fluted fragment with transitivity. In *44th International Symposium on Mathematical Foundations of Computer Science, MFCS 2019*, pages 15:1–15:15, 2019.
24 W. V. Quine. On the limits of decision. In Proceedings of the 14th International Congress of Philosophy, volume III, pages 57–62, 1969.

25 D. Scott. A decision method for validity of sentences in two variables. Journal Symbolic Logic, 27:477, 1962.

26 W. Szwast and L. Tendera. The guarded fragment with transitive guards. Annals of Pure and Applied Logic, 128:227–276, 2004.

27 B. ten Cate and L. Segoufin. Unary negation. Logical Methods in Comp. Sc., 9(3), 2013.
A UNFO with functional restrictions

The following result is implicit in [27]. Here we prove it directly, by a simple reduction from domino tilings.

Theorem 14. The satisfiability and the finite satisfiability problems for UNFO with four variables (even without transitive relations) and two functional binary relations and some unary relations are undecidable.

Proof. We axiomatize a class of grid structures, with \( H \) being the horizontal successor relation and \( V \) being the vertical successor relation, as follows. We assert that \( H \) and \( V \) are functional

\[
\forall x(\exists y H xy \land \exists y V xy)
\]

and then define grids using the following UNFO formula

\[
\forall x(\exists y z t (H xy \land V x z \land V y t \land H z t))
\]

Let \( \lambda \) be the conjunction of the two above formulas. Clearly, the standard grids on \( \mathbb{N} \times \mathbb{N} \) and on the \( t \times t \) tori, for \( t \in \mathbb{N} \), are models of \( \lambda \). Conversely, any model of \( \lambda \) homomorphically embeds the standard grid. Having \( \lambda \) with such properties, reducing the undecidable domino tiling problem: given a domino system verify if it tiles \( \mathbb{N} \times \mathbb{N} \) (some \( \mathbb{Z}_t \times \mathbb{Z}_t \)) to satisfiability (finite satisfiability) is routine.

B Normal form

Lemma 15 (Scott-normal form). For every UNFO+\( S \) sentence \( \varphi \) one can compute in polynomial time a normal form UNFO+\( S \) sentence \( \varphi' \) over signature extended by some fresh unary symbols, such that any model of \( \varphi' \) is a model of \( \varphi \) and any model of \( \varphi \) can be expanded to a model of \( \varphi' \) by an appropriate interpretation of the additional unary symbols.

Proof. (Sketch) Take any UNFO+\( S \) sentence \( \varphi \). Recall that it uses no universal quantifiers. First we convert it to its UN-normal form, in which each maximal block of quantifiers leaves at most one variable free. This can be done as described in [27]. Then we consider an innermost subformula of \( \varphi \), starting with a block of quantifiers, \( \exists y \psi(x, y) \), replace it by a fresh unary predicate \( P(x) \), and add two auxiliary conjuncts \( \forall x \exists y (\neg P(x) \lor \psi(x, y)) \) and \( \forall x y (\psi(x, y) \lor \neg P(x)) \), whose conjunction is equivalent to \( \forall x (P(x) \leftrightarrow \exists y \psi(x, y)) \). Moving up the original formula \( \varphi \) we repeat this procedure for subformulas that are now innermost, and so forth. The formula obtained in this process has, up to trivial logical transformations, the desired shape and properties.

C Proof of Lemma 1

Proof. Due to (a1) all elements of \( A' \) have the required witness structures for all \( \forall \exists \)-conjuncts. It remains to see that the \( \forall \)-conjunct is not violated. But since \( A \models \neg \varphi_0 \) and \( \varphi_0 \) is a quantifier-free formula in which only unary atoms may be negated, it is straightforward, using (a2).
D. Danielski and E. Kieroński

D. Missing proofs and a strategy description from Section 3

D.1 Proof of Lemma 2

Proof. It is readily verified that $\mathfrak{A}'$ meets the properties required by Lemma 1. In particular function $h$ associated with the given unramlling is the required homomorphism. That $\mathfrak{A}'$ is tree-like and has an appropriately bounded degree is also straightforward. For the last condition assume to the contrary that there exist $u$ and a downward-\(T_u\)-path $\langle a_i \rangle_{i=0}^N$ in $\mathfrak{A}'$ with rank bigger than $|A|$. Then there are indices $i_0, \ldots, i_{|A|}$ such that $\mathfrak{A} \models T_u a_{i_0} a_{i_1+1} \land \mathfrak{A} \models \neg T_u a_{i_0} a_{i_1+1}$. Since $h$ preserves the connections between elements and their witnesses we have $\mathfrak{A}_0 \models T_u h(a_{i_0}) h(a_{i_1}) \land \mathfrak{A} \models \neg T_u h(a_{i_0}) h(a_{i_1})$. By the pigeonhole principle there exist $x < x'$ such that $h(a_{i_0}) = h(a_{i_1})$. This gives, by transitivity of $T_u$, that $\mathfrak{A}_0 \models T_u h(a_{i_0}) h(a_{i_1})$. Contradiction. Q.E.D.

D.2 Proof of Lemma 3

Proof. (i) Assume to the contrary that the given system is locally consistent but not globally consistent. This means that for some node $a_i$, for some $\alpha \in \mathfrak{A}$, and some $\mathcal{T}$ we have that $\alpha \in \mathfrak{A} \mathfrak{d} \mathfrak{a} (\mathcal{T})$ but $\alpha \notin \mathfrak{I} \mathfrak{d} \mathfrak{c} \mathfrak{a} (\alpha) (\mathcal{T})$, that is there is a node $b$, of atomic type $\alpha$ such that $\mathfrak{A} \models \mathcal{T} ab$. Thus, there exists a sequence of distinct nodes $a = a_0, a_1, \ldots, a_N = b$ such that $a_i$ is either a child or the parent of $a_{i+1}$ and $\mathfrak{A} \models \mathcal{T} a_i a_{i+1}$. Observe that it must be $a \neq b$ since otherwise Condition (ld2) would not be satisfied at $a$. By induction, using Condition (ld1), we can show that $\alpha \in \mathfrak{A}_i \mathfrak{d} \mathfrak{a} (\mathcal{T})$ for all $i$, in particular for $i = N - 1$. We now get a contradiction with (ld2) at $a_{N-1}$ or $a_N$ (depending on which of them is the parent of the other). Part (ii) is straightforward. Q.E.D.

D.3 Proof of Lemma 4

Proof. (i) Follows from the fact that for every $a \in A'$ its downward family in $\mathfrak{A}'$ is an isomorphic copy of the downward family of $a$ in $\mathfrak{A}$. Moreover, due to the requirement that a subtree with root $a$ is replaced by a subtree with the root of the same generalized type as the type of $a$, this copy also preserves declarations. (ii) Consider now any pair of elements $a, a' \in A'$. Assume that they have $1$-types, resp., $\alpha$ and $\alpha'$. If $a$ is a child of $a'$ or $a'$ is a child of $a$ then the edge that joins them is an isomorphic copy of an edge from $\mathfrak{A}$. Otherwise, due to the definition of light tree-like structure, $a$ and $a'$ may be joined only by some transitive relations. Let $\mathcal{T}$ be the set of all transitive relations $T_u$ such that $\mathfrak{A}' \models T_u a a'$. By part (i) of this lemma the system of declarations is locally consistent on $\mathfrak{A}'$. By Lemma 3 it is also globally consistent. In particular $\mathfrak{d}(\alpha)(\mathcal{T}) \subseteq \mathfrak{I} \mathfrak{d} \mathfrak{c} \mathfrak{a} (\alpha)(\mathcal{T})$. Since $\alpha' \notin \mathfrak{I} \mathfrak{d} \mathfrak{c} \mathfrak{a} (\alpha)(\mathcal{T})$ it follows that $\alpha' \notin \mathfrak{d}(\alpha)(\mathcal{T})$. Thus, there is a realization $b$ of $\alpha'$ in $\mathfrak{A}$ such that $\mathfrak{A} \models \mathcal{T} ab$, and hence the function mapping $a$ to itself and $a'$ to $b$ is the required homomorphism. (iii) To see that all nodes of $\mathfrak{A}'$ have the required witnesses again just note that for every $a \in A'$ its downward family in $\mathfrak{A}'$ is an isomorphic copy of the downward family of $a$ in $\mathfrak{A}$. That $\mathfrak{A}' \models \phi$ follows now from part (ii) of this lemma and from Lemma 1. Q.E.D.

D.4 The pruning strategy

Let $\mathfrak{A}$ be a light-$\varphi$-tree-like unramlling of a finite model of $\varphi$. Let $(\mathfrak{d}_a)_{a \in A}$, $\mathfrak{d}_a := \mathfrak{I} \mathfrak{d} \mathfrak{c} \mathfrak{a} (\alpha)$ be the canonical system of light declarations on $\mathfrak{A}$. During a top-down pruning process we define a function $s$ assigning to the surviving nodes a permutation of the set $\{1, \ldots, 2k\}$. In each $\mathfrak{A}_i$ this function is partial and defined for all nodes of depth at most $i$ (and is not
modified in the subsequent structures for these nodes). Its values can be then transferred to $\mathfrak{A}'$, where it becomes total. The purpose of $\mathfrak{s}$ is to define some order of shortening paths at a given node. Intuitively, for $\mathfrak{s}(a) = \tau$, if $v < v'$ then we prefer to shorten $T_{\tau(v)}$ over $T_{\tau(v')}$. 

In $\mathfrak{A}_i$, let $\mathfrak{s}$ assigns an arbitrary permutation to the root. Assume that we have constructed $\mathfrak{A}_i$, for $i \geq 0$, and we have assigned the values of $\mathfrak{s}$ to all its nodes of depth at most $i$. Consider a node $a$ of $\mathfrak{A}_i$, of depth $i + 1$. Denote its parent by $a'$. Our task is to choose a descendant $b$ of $a$ whose subtree will replace the subtree of $a$ (or decide that this subtree is left untouched).

To make our choice we will look at permutation $\tau = \mathfrak{s}(a')$ assigned to the parent of $a$ and at three sets of indices, $K$, $S$, $D$, whose definition depends on the connection between $a'$ and $a$, as follows.

We say that $T_u$ is (i) killed at $a$ (or: at the edge $(a', a)$) if $\mathfrak{A} \models \neg T_{\tau(v)}' a' a$, (ii) sustained at $a$ if $\mathfrak{A} \models T_{\tau(v)}' a' a$, and (iii) diminished at $a$ if $\mathfrak{A} \models T_{\tau(v)}' a' a'$. Let $K = \{v : T_{\tau(v)}$ is killed at $a\}$, $S := \{v : T_{\tau(v)}$ is sustained at $a\}$, $D := \{v : T_{\tau(v)}$ is diminished at $a\}$. Note that the above sets contain not the numbers of transitive relations but rather their positions in the permutation $\tau$.

If $D = \emptyset$ then we just choose $b$ to be $a$, that is we decide to leave the subtree of $a$ as it is. If $D \neq \emptyset$ then let $v_D = \min D$ and choose $b$ to be a node of $A_d$ such that (i) $\text{gtp}_D(a) = \text{gtp}_D(b)$ (the standard requirement in the pruning process), (ii) for all $v < v_D$, $v \in S$: $\tau(a')(v) \leq \tau(a')(v)$ (where the ranks may be equivalently computed in $\mathfrak{A}_i$), and (iii) $\tau_\mathfrak{s}(a'(v_D))$ is the lowest possible. Note that such an element exists (however, it may happen that $b = a$) and $\tau_\mathfrak{s}(a'(v_D)) \leq \tau_\mathfrak{s}(a)(v)$.

It remains to define $\mathfrak{s}(b)$. If $K \neq \emptyset$ then let $v_K = \min K$ and set $\mathfrak{s}(b) := \tau \circ (v_K, v_K + 1, \ldots, 2k - 1, 2k)$, where $\circ$ denotes permutation composition, and the second argument is a cyclic permutation. In other words, we cyclically move the elements on positions $v_K, \ldots, v_{2k}$ in $\tau$ by one position to the left. This way the relation with the biggest priority among the relations that are killed in the current step now gets the lowest priority. If $K = \emptyset$ then set $\mathfrak{s}(b) := \tau$.

**Lemma 16.** Let $\mathfrak{A} \models \varphi$ be a light tree-like structure over signature $\sigma$, with transitive paths bounded by some natural number $M$. Then the result $\mathfrak{A}'$ of any top-down pruning process respecting our pruning strategy is a light tree-like structure with transitive paths bounded doubly exponentially in $|\varphi|$.

**Proof.** Let $M_\varphi$ be the number of the generalized types realized in $\mathfrak{A}$ increased by 2. Clearly, $M_\varphi$ is bounded doubly exponentially in $|\varphi|$. Let us first make an auxiliary estimation.

**Claim 17.** Let $v_0$ and a downward-$T_u$-path $\vec{a} = (a_i)_{i=1}^{N}$ in $\mathfrak{A}'$ be such that for all $i, i'$ and $v \leq v_0$ we have $\mathfrak{s}(a_i)(v) = \mathfrak{s}(a_{i'})(v)$ (in this case, slightly abusing notation, we write $\mathfrak{s}(\vec{a})(v) = \mathfrak{s}(a_i)(v)$) and let $u := \mathfrak{s}(\vec{a})(v_0)$. Let $\text{Sum} = \sum_{v \leq v_0} \tau^\mathfrak{s}(\mathfrak{s}(\vec{a})(v))$. Then $\tau^\mathfrak{s}(\vec{a}) \leq M_\varphi \cdot \text{Sum} + \text{Sum} + M_\varphi$.

**Proof.** Consider first the case $v_0 = 1$ and take a downward-$T_u$-path $\vec{a} = (a_i)_{i=1}^{N}$ in $\mathfrak{A}'$ such that $\mathfrak{s}(\vec{a})(1) = u$. $\text{Sum} = 0$ in this case, so we need to show that $\tau^\mathfrak{s}(\vec{a}) \leq M_\varphi$.

Observe first that the $T_u$-rank of elements, computed in $\mathfrak{A}$, is non-increasing along $\vec{a}$. More precisely, for $1 \leq i < N$, if $\mathfrak{A}' \models T_u a_i a_{i+1} \land \neg T_u a_{i+1} a_i$, then $\tau^\mathfrak{s}(a_i) > \tau^\mathfrak{s}(a_{i+1})$ and if $\mathfrak{A}' \models T_u a_i a_{i+1} \land T_u a_{i+1} a_i$, then $\tau^\mathfrak{s}(a_i) \geq \tau^\mathfrak{s}(a_{i+1})$. Both properties follow from our strategy: the former from condition (ii) (note that in this case $v_D = 1$) and the latter from condition (ii). Assume now to the contrary that $\tau^\mathfrak{s}(\vec{a}) > M_\varphi$. This means that there are at least $M_\varphi + 1$ elements $a_{i+1}$ such that $\mathfrak{A}' \models T_u a_i a_{i+1} \land \neg T_u a_{i+1} a_i$. Thus, by the pigeonhole principle, there are at least two such elements, $a_{x+1}$ and $a_{x'+1}$, say $x < x'$, having the same
generalized types in $\mathfrak{A}$. By the observation above, $r^\mathfrak{A}_u(a_{x+1}) > r^\mathfrak{A}_u(a_{x+1})$. But then, condition (iii) of our strategy requires us to use $a_{x+1}$ instead of $a_{x+1}$ when looking for a child of $a_x$. Contradiction.

We now show that the Claim is true for arbitrary $2 \leq v_0 \leq 2k$. Take a downward-$T_u$-path $\bar{a} = (a_i)_{i=1}^N$ in $\mathfrak{A}'$ such that $s(a_i)(v)$ is constant on $\bar{a}$ for all $v \leq v_0$ and $s(\bar{a})(v_0) = u$. Note that none of $1, \ldots, v_0$ belongs to any of the sets $K$ computed during the construction of $\bar{a}$ (since if $v \in K$ then $s(a_i)(v)$ changes). Thus, relations $T_{\bar{a}(1)}, \ldots, T_{\bar{a}(v_0)}(v) = T_u$ are either diminished or sustained along $\bar{a}$. Assume to the contrary that $r^\mathfrak{A}_u(a) > M_\varphi \cdot \text{Sum} + \text{Sum} + M_\varphi$.

Consider the edges of $\bar{a}$ such that $T_u$ is diminished on them. The number of such edges on which additionally some of $T_{\bar{a}(v_0)}(v)$ for $v < v_0$ is diminished is bounded by $\text{Sum}$ (by the definition of ranks). Thus at more than $M_\varphi \cdot \text{Sum} + M_\varphi$ edges we chose $v_D = v_0$ along the considered path. Let $Q$ be the set of such edges.

We now divide $\bar{a}$ into fragments containing $M_\varphi$ edges from $Q$ (a suffix of $\bar{a}$ with less then $M_\varphi$ edges may be left). There are at least $\text{Sum} + 1$ such fragments. It follows, by the pigeonhole principle, that in at least one of them, call it $\bar{a}^*$, all of the $T_{\bar{a}(v)}(v)$, for $v < v_0$ are sustained. By arguments similar to those given in the case $v_0 = 1$ we see that the ranks $r^\mathfrak{A}_{\bar{a}(1)}(a)$ are non-increasing along $\bar{a}^*$ for $v \leq v_0$, and $r^\mathfrak{A}_{\bar{a}(v_0)}(a)$ decreases at least $M_\varphi$ times. The latter happens, again by the pigeonhole principle, at least twice for edges leading to elements with the same generalized types in $\mathfrak{A}$, so, as in the case of $v_0 = 1$, we get a contradiction with our strategy.

The above claim allows us in particular to compute recursively a (uniform) doubly exponential bound on $r^\mathfrak{A}_u(\bar{a})$ for all $v_0$, $u$ and $\bar{a}$ as in assumption. Denote this bound by $\overline{M}_\varphi$.

Consider now any downward-$T_u$-path $\bar{a} = (a_i)_{i=1}^N$ in $\mathfrak{A}'$. For each node $a_i$ from $\bar{a}$ let $v_u(a_i)$ be such that $s(a_i)(v_u) = u$. Due to the strategy that we use to define $s$ the value of $v_u$ is non-increasing along $\bar{a}$. Indeed, when moving from $a_i$ to $a_i + 1$ the value of $v_u$ is either unchanged or decreases by 1; the only chance of increasing it would be to change it to $2k$ but this happens only when $T_u$ is killed. Let us divide $\bar{a}$ into fragments $\bar{a}_1, \bar{a}_2, \ldots$ on which $v_u$ is constant. The number of such fragments is obviously bounded by $2k$. On each of such fragments $\bar{a}_i$ for all $v \leq v_u$ we have that $s(\bar{a}_i)(v)$ is constant. So we can apply Claim 17 to bound $r^\mathfrak{A}_u(\bar{a}_i)$ by $\overline{M}_\varphi$. This gives the desired doubly exponential bound $\overline{M}_\varphi = 2k \overline{M}_\varphi$ on $r^\mathfrak{A}_u(\bar{a})$ and finishes the proof of Lemma 16.

Our strategy, together with Lemma 4 gives Lemma 5.

D.5 Proof of Claim 8

Proof. (i) Take any $b' \in B_0$. Note that $p \upharpoonright B_0$ goes into $A_{p(b)} \cap [p(b)]_{E_{T_u} \cup \{T_u, T_u^{-1}\}}$, so $p(b')$ belongs to the subtree of $p(b)$ and is connected to it by both $T_u$ and $T_u^{-1}$. It follows that extending any downward-$T_u$-path starting at $p(b')$ by the path from $p(b)$ to $p(b')$ does not change its $T_u$-rank. Hence $r^\mathfrak{A}_u(p(b')) \geq r^\mathfrak{A}_u(p(b'))$. (ii) By our strategy of choosing witnesses and assigning layers to them we know that $p(c')$ is a child of $p(b)$ and $\mathfrak{A} \models T_u(p(b)p(c') \land \neg T_u p(c')p(b))$. Thus, extending any downward-$T_u$-path starting at $p(c')$ by the edge from $p(b)$ to $p(c')$ increases its rank by 1. The claim thus follows.

D.6 Correctness of the construction in the proof of Lemma 7

Recall Fig. 2. We naturally divide $\mathfrak{A}_0^g$ and $\mathfrak{A}_i^g$ into $4l$ levels. For $g = 0, 1$ and $1 \leq i \leq 2l$, level $2g + i$ is the union of layers $L_i$ of all components of color $g$. 
Figure 3 $T_3$ and $T_4$-connections (recall: $T_4 = T_3^{-1}$) between levels of $\mathfrak{A}_0$. The absence of an arrow from a level (group of levels) $A$ to level (group of levels) $B$ means that there are no $T_3$-connections from elements of $A$ to elements of $B$. Groups of levels in the right figure contain $2l - 1$ levels each.

(1) Assume that $\mathfrak{A}_{a_0} \models \{a_0\}_{E_{\text{tot}}}$ is not a singleton (in this case also $\mathfrak{A}_0$ is not a singleton). Thus each of the $T_u \in E_{\text{tot}}$ is total on it, in particular it is reflexive. By the inductive assumption it is total on subcomponents (for singleton subcomponents it follows from (1), using the fact that $h$ preserves the 1-types, and thus, in particular, reflexivity of the $T_u$).

When components are formed out of subcomponents we always use 2-types from $[a_0]_{E_{\text{tot}}}$. It is thus straightforward that, after taking a connected fragment of the graph of components and applying the transitive closures to get $\mathfrak{A}_0$, all relations from $E_{\text{tot}}$ become total.

(2) As $a'_0$ we take the root of $C^{\gamma_{a_0},0}_{i,l,k}$. Recall that we explicitly map the root of the pattern component $C^{\gamma_{a_0}}$ by $p$ to $a_0$.

(3) If we prove that an element and its pattern have the same 1-types, then the existence of witnesses is easy to show. Indeed, we explicitly take care of this when building components in Step 2 (Providing witnesses). In each component, every element from layer $L_i^1$ has its witnesses in $L_j^1 \cup L_j^{1,\text{init}} \cup L_i^{1,\text{init}}$. Every interface element is identified with the root of some other component so it also has its witnesses. So, the only potential danger is that some 1-types are enlarged. While we initially explicitly copy the 1-types from the original model, it is probably not completely obvious that they remain the same after taking the transitive closures: the potential danger is that we may possibly form a $T_u$-cycle from an element $a'$, such that $\mathfrak{A} \models \lnot T_u p(a') p(a')$.

To see that this cannot happen, as well as to prepare ourselves for a proof of (1), we now spend a while on understanding transitive paths in $\mathfrak{A}_0$ and thus transitive connections in $\mathfrak{A}_0$. First, observe that each 2-type in $\mathfrak{A}_0$ is either a copy of a 2-type between an element and its witness (Step 2), was set when putting a subcomponent (Step 1), or is trivial, that is, it makes true only some unary atoms. We say that a sequence $a_1, \ldots, a_N \in A_0$ satisfying $a_i \neq a_{i+1}$ for all $1 \leq i \leq N - 1$ is a path in $\mathfrak{A}_0$ if for all $1 \leq i \leq N - 1$ the elements $a_i, a_{i+1}$ either belong to the same subcomponent or one is put as a witness for the other, and a $T_u$-path if for all $1 \leq i \leq N - 1$ we have $\mathfrak{A}_0 \models T_u a_i a_{i+1}$. Observe that in particular every $T_u$-path in $\mathfrak{A}_0$ is a path and every path in $\mathfrak{A}_0$ is automatically a $T_u$-path for all $u \geq 2l$.

Consider a pair of elements $a', b'$ (possibly $a' = b'$). We are interested in the 2-type of $(a', b')$ in $\mathfrak{A}_0$, in particular in the $T_u$-paths joining $a', b'$ in $\mathfrak{A}_0$. Assuming that this 2-type contains some binary symbol other than the symbols from $E_{\text{tot}}$, we argue, that it is identical to a 2-type of some pair in some simplifications of $\mathfrak{A}_0$. (The case where $a', b'$ are joined only by relations from $E_{\text{tot}}$ is simple and will be treated separately.)

Reduction 1. Note that for any $T_u$-path, $1 \leq u \leq 2l$, there is a set of at most $2l$ consecutive
levels of our cylindrical structure in which all elements of this path are contained. Consider, e.g., the case of a $T_3$-path, see Fig. 3. This path must be contained either in the union of red levels $L_1-L_3$, blue levels $L_5-L_7$, and at most one (red or blue) level $L_4$, or, symmetrically, in the union of blue levels $L_1-L_3$, red levels $L_5-L_7$, and at most one level $L_4$. Analogously for the other $T_n$. This becomes evident when looking at the graph from the right part of Fig. 3 whose nodes are strongly connected components (consisting of at most $2l-1$ levels) of the graph from the left part of this figure.

Assume that $a'$ and $b'$ are connected by a $T_w$-path, for some $1 \leq u \leq 2l$ or by a non-transitive connection crossing one of the borders between colors, that is using an edge between a leaf of color $g$ and a root of color $1-g$, for some $g = 0, 1$. Then for any $1 \leq v \leq 2l$, including $v = u$, any $T_v$-path from $a'$ to $b'$ cannot cross the other border since the minimal set of consecutive levels containing the levels of $a'$, $b'$ and the levels adjacent to that other border would have cardinality greater than $2l$. Thus we can cut all the connections between leaves of color $1-g$ and roots of color $g$ (that is, make any atom containing a pair of such elements false). See Fig. 3 where this step is illustrated for $g = 1$. Let $D'_0$ be the structure so-obtained and $D'_0$ its transitive closure, call it a transsection of $A'_0$. By the discussion above, the inclusion map $\iota : A'_0 \{\{a, b\} \to D'_0$ is a homomorphism

Reduction 2. Take $g$ from the previous reduction. By our strategy of joining the components, there exists a generalized type $\gamma$ such that any $T_u$-path joining $a'$ and $b'$ is contained in components of the forms $E^0_{\gamma} \cdot \gamma$ and $E^{1-\gamma}_{\gamma}$. Denote $E^0_{\gamma}$ the restriction of $E^0_0$ to these components and $E^0_{\gamma}$ its transitive closure. Choose any component $E^0$ of the form $E^0_{\gamma} \cdot \gamma$. Recall that all of them are isomorphic copies of the pattern component $E^0$. Now, define another auxiliary structure $A'_0$ obtained by restricting $E^0_{\gamma}$ to the union of $C^0$ and the domains of the components of the form $E^{1-\gamma}_{0}$. Let $A'_0$ be its transitive closure. There is a natural projection $\pi : E^0_0 \to A'_0$, which maps the elements of the components of the form $E^0_{\gamma} \cdot \gamma$ into the corresponding elements of component $E^0$ (being the identity on the other elements). Observe that $\pi : E^0_0 \to A'_0$ is a homomorphism, and as we can apply it to transitive paths, we obtain that also $\pi : E^0_0 \to A'_0$ is a homomorphism. See Fig.

$A'_0$ looks like a single component but is twice as high. It can be viewed as a tree $\tau$ defined by taking the subcomponents used to build the components of $A'_0$ (Step 1) as the nodes and connecting two of them if one of them contains a witness for some element of the other (Step 2). Note that this way the leaves of component $E^0$ are parents of some roots of the remaining components of $A'_0$.

Now we are ready to come back to the proof of (13). What remains is to show that for every $a' \in A'_0$, we have $\text{atp}^A(a') = \text{atp}^A(p(a'))$. Recall that the only possible reason for not being so is that the 1-type of $a'$ was enlarged by the application of the transitive closure to $A'_0$. That is, there exists, for some $1 \leq u \leq 2l$, a $T_u$-path that connects $a'$ with itself in $A'_0$. Taking $b' = a'$, we use Reduction 1 and 2 (if $a'$ is an element of a component of color $g$, then we choose $E^0$ to be the component of $a'$), and after an application of $\tau$, we have a $T_u$-path connecting $a$ with itself in $A'_0$. Due to the tree shape of $A'_0$ either there is such a path in the subcomponent of $a'$ or in the 2-element substructure joining $a'$ with one of its witnesses, denote it $w'$. It follows that $T_{u \cdot x\cdot xx} \in \text{atp}^A(p(a'))$. Indeed, in the former case we get it by the fact that the subcomponents are closed transitively and by the 1-type assumption in (13) for the subcomponent of $a'$. In the latter case we just recall that the 2-type for the pair $a', w'$ was copied to $A'_0$ from the original structure, which was transitively closed.

(b4) Suppose that $a', b' \in A'_0$ are not connected by any relation except for those from $E_{\text{cat}}$. Then it suffices to take $\mathfrak{h} := p(\{a', b'\})$, use (b3) for the 1-type preservation and recall that $p(A'_0) \subseteq [a_0]E_{\text{cat}}$. From this the other required condition also follows. From now we assume
that \( a', b' \) are connected by some binary relation not belonging to \( E_{tot} \).

Assume that \( \mathfrak{A}_0 \models T a'b' \), that is for any \( T_u \in T \) there exists a \( T_u \)-path in \( \mathfrak{A}_0 \) that joins \( a' \) and \( b' \). Observe that such paths may differ for different \( T_u \) and, firstly, we want to find a common path for all the \( T_u \in T \). As before, apply Reductions 1 and 2 to get that \( \pi(a'), \pi(b') \) are connected in \( \mathfrak{A}_0 \) by \( T_u \)-paths for all \( T_u \in T \). If we prove the thesis for \( \pi(a'), \pi(b') \) (in particular, looking for an appropriate partial homomorphism from \( \mathfrak{A}_0 \) into \( \mathfrak{A} \)), then it holds also for \( a', b' \), as \( \pi \) is a homomorphism from \( \mathfrak{A}_0 \) to \( \mathfrak{A}_0 \), \( p \circ \pi = p \) and \( \pi \) preserve the 1-types.

Now, using standard tree reasoning and the fact that the subcomponents are transitively closed, one may observe that the shortest path (in \( \mathfrak{A}_0 \)) connecting \( \pi(a') \) with \( \pi(b') \) is a \( T_u \)-path for any \( T_u \in T \). Denote this path \( \pi(a') = a_1, \ldots , a_N = \pi(b') \).

Take a 1-type \( \alpha \not\in \text{ldec}^\pi(p(b')) \). We prove by induction on \( j = N, N-1, \ldots , 1 \) that \( \alpha \not\in \text{ldec}^\pi(p(a_j)) \). In particular we prove the second part of condition (h4). The thesis for \( j = N \) clearly holds, since we explicitly assumed it. Now consider \( a_j \) and \( a_{j+1} \). Then either they appear together in some subcomponent or a copy of a witness structure (\( a_j \) is a witness for \( a_{j+1} \) or vice versa). In the first case, use the condition (b4) for such subcomponent, which holds by the inductive assumption of Lemma \( \overline{7} \). In the second, just use the fact that (ld1) holds for \( p(a_j) \) and \( p(a_{j+1}) \).

Finally, we build the required homomorphism \( \mathfrak{b} \). If \( \pi(a') = \pi(b') \), then it suffices to put \( \mathfrak{b}(\pi(a')) = p(\pi(a')) \) and recall (b3). If \( \pi(a') \not= \pi(b') \) and they are connected by some not-transitive atom, then \( N = 2 \) (that is, the path is \( \pi(a') = a_1, a_2 = \pi(b') \)). Then if \( \{ \pi(a'), \pi(b') \} \) is a copy of a pair consisting of an element and its witness, then put \( \mathfrak{b} := p(\pi(a'), \pi(b')) \); and if they appear in the same subcomponent, use a homomorphism guaranteed by condition (h4) of the inductive assumption. So now assume that \( \pi(a') \not= \pi(b') \) and \( T = \{ T_u : \mathfrak{A}_0 \models T_u \pi(a') \pi(b') \} \) is non-empty. First, observe that \( \text{ldec}^\pi(p(a_{N-1})) \not\in \alpha \), where \( \alpha = \text{atp}^\pi(a_N) \).
Indeed, if \( a_{N-1}, a_N \) are an element and its witness (or vice versa), then it suffices to use the fact that \( p \) preserves the 1-types (that is, a part of (b3)). If \( a_{N-1}, a_N \) belong to the same subcomponent, then by the condition (b4) of the inductive assumption of Lemma 7 there exists in a homomorphism \( h \), which gives us that \( \mathrm{ldec}^2(\{p(a_{N-1})\}) \not\models \mathrm{atp}^2(\{h(a_N)\}) (\equiv \alpha) \). Thus, applying the induction just like one paragraph above, we get that \( \mathrm{atp}^2(\{h(a_N)\}) \not\models \alpha \). By the maximality of such declarations, there exists \( b \in A \) such that \( \mathrm{atp}^2(\{h(b)\}) = \alpha \) and \( A \models T(p(a'))b \). Put \( h(\pi(a')) := p(\pi(a')) \), \( h(\pi(b')) := b \).

### D.7 Size of models constructed in the proof of Lemma 7

> Claim 18. The construction in the proof of Lemma 7 produces models of size at most triply exponential in the size of the given formula.

**Proof.** The following routine estimation shows that \( |A|_0 \) is triply exponential in \( n = |\varphi| \), regardless of the choice of the initial tree-like model \( A \). We calculate a bound \( S_{2l} \) on the size of the structure obtained in the proof of Lemma 7 for \( |\varphi_0| = 2l \). We are interested in \( S_{2l+2} \), which is the desired bound on the size of \( A \) (we use \( S_{2k+2} \) here, rather than \( S_{2k} \), because we may potentially introduce the auxiliary identity relations in the base step of induction). Recall that any pattern component may be viewed as a (rooted) tree of subcomponents consisting of at most \( 2((M_{\varphi} + 1) \) sublayers, which is automatically a bound on the height of this tree. The root (the only vertex of depth 1) corresponds to a single subcomponent, that contains at most \( S_{2l-2} \) elements. Each of these elements may then require at most \( n \) witnesses, each of which is added to the initial part either of the next sublayer of the first layer or of the first sublayer of the next layer (in either case: to a subcomponent of depth 2). Thus we get that we put at most \( S_{2l-1}n \) elements, each of them then gives rise to a subcomponent (being itself its origin), so we have at most \( S_{2l-2}n \) elements corresponding to the vertices of the tree of depth 2. Iterating, we have at most \( S_{2l-2}n^{l-1} \) elements corresponding to the vertices of the tree of depth \( i \), which leads to an estimate \( (S_{2l-2}n)^{2((M_{\varphi} + 1) + 1)} \) both on the number of elements in a pattern component, and the number of interface elements in its extended version. In the joining phase we thus use at most \( 2 \cdot |A|^2 \cdot (S_{2l-2}n)^{2((M_{\varphi} + 1) + 1)} \) components. Finally, estimating \( l \) in the exponents by \( 2n \), we get a bound \( S_{2l} = 2 |A|^2 (S_{2l-2}n)^{8n(M_{\varphi} + 1) + 2} \). Solving this recurrence relation, and recalling that \( M_{\varphi} \) and \( |\varphi| \) are doubly exponential in \( |\varphi| \), we obtain a triply exponential bound on \( S_{2k+2} \). ▶

### D.8 Proof of Theorem 9

**Proof.** The lower bound can be obtained for the two-variable UNFO\(^2\)\(S \) in the presence of one transitive relation. The proof is a straightforward adaptation of the lower bound proof for GF\(^2\) with transitive relations in guards [19].

For the upper bound, we design an AExpSPACE algorithm. Given \( \varphi \) in UNFO\(^2\)\(S \) this algorithm converts it to normal form \( \varphi' \) and then looks for a model of \( \varphi' \) of form (b3), that is, a light tree-like model of with doubly exponentially bounded transitive paths (as in Lemma 5). As we proved, \( \varphi' \) has such a model iff \( \varphi \) has a finite model. We first calculate the bound \( M \) on the rank of transitive paths, and let \( M := (M + 1)^{2k} \cdot |\gamma| \), where \( \gamma \) is the set of all the generalized types over the signature of \( \varphi' \). Both values are bounded doubly exponentially in \( |\varphi| \).

In an alternating fashion we construct a single downward path of a model. We start from the root, by guessing its generalized type, and setting the values of variables \( r_1, \ldots, r_{2k} \) to 0.
The intention is that \( r_u \) is the maximal rank of a downward \( T_u \)-path ending at the current node. Then, having constructed an element \( a \), we universally choose a \( \exists \beta \)-conjunct of \( \varphi' \), and, if \( a \) is not a witness for itself, add a witness \( a' \) for \( a \) and this conjunct. We guess a 2-type joining \( a \) and \( a' \), guess the generalized type of \( a' \) and update the values of the \( r_u \), in accordance with the guessed 2-type (each value may stay unchanged, increase by 1, or reset to 0). We check that \( a' \) is indeed a witness for \( a \) and the considered conjunct and verify the LCC conditions between the generalized types of \( a \) and \( a' \), rejecting if any of these fails. We reject also if any of the \( r_u \) exceeds \( M \). We keep an additional counter, measuring the depth of the current node in the path constructed so far, and accept if this depth exceeds \( M \).

It is clear that the described algorithm can be implemented in \textup{AExpSpace}: we only need to store information about a pair of nodes, values of the \( r_u \) plus a counter. These can be written using exponentially many bits.

It is also not difficult to see that the above-described algorithm accepts its input \( \varphi \) iff \( \varphi \) has a finite model. If \( \varphi \) has a finite model then its normal form \( \varphi' \) also has such a model. Let \( \mathfrak{A} \models \varphi' \) be a light tree-like model promised in Lemma 5. The algorithm can then accept by making all the guesses in accordance with \( \mathfrak{A} \). In the opposite direction, if the algorithm has an accepting run, then from this run we can naturally infer a tree-like structure \( \mathfrak{A}^* \) consisting of at most \( M+1 \) levels. Note that on each path of length \( M+1 \) from the root of to a leaf in \( \mathfrak{A}^* \) there is a pair of nodes for which the guessed generalized types, and the calculated values of all the \( r_u \) are identical. Cut each branch at the first position on which the above parameters reappear and make a link from this point to the their first occurrence on the considered branch. Naturally unravel so-obtained structure into an infinite tree-like structure \( \mathfrak{A} \). It should be clear that the guessed generalized types still respect LCCs in \( \mathfrak{A} \) and that the values of the \( r_u \) copied to \( \mathfrak{A} \) from \( \mathfrak{A}^* \) remain correct. Thus \( \mathfrak{A} \) is indeed a tree-like model of \( \varphi' \) with appropriately bounded transitive paths.

\section{The general case}

In this section we consider UNFO+\( \mathcal{S} \) with arbitrarily many variables. The material here is self-contained, we will, however, make some references to Section 3 in which we dealt with the simplified two-variable case. In such references we will emphasise the differences and similarities between both variants.

\subsection{Tree pruning}

\subsubsection{Tree-like unravellings}

We first naturally generalize the notion of light tree-like structures used in the two-variable case. Recall that any set consisting of a node \( b \) of a tree and all its children is called a family, or the downward family of \( b \). We say that \( \mathfrak{A} \) is a tree-like structure if its nodes can be arranged into a rooted tree in such a way that if \( \mathfrak{A} \models B\overline{a} \) for some non-transitive relation symbol \( B \), then \( \overline{a} \) is contained in some family, and if \( \mathfrak{A} \models T_u a a' \) for some transitive \( T_u \), then either \( a = a' \) or there is a sequence of distinct nodes \( a = a_0, a_1, \ldots, a_k = a' \) such that \( a_i \) and \( a_{i+1} \) belong to the same family and \( \mathfrak{A} \models T_u a_i a_{i+1} \). A slight difference, compared to light tree-like structures is that now we admit direct (omitting the parent) transitive connections between some children of a node. For a tree-like structure \( \mathfrak{A} \) and \( a \in A \) we denote by \( A_a \) the set of all nodes in the subtree rooted at \( a \) and by \( \mathfrak{A}_a \) the corresponding substructure.

Let us recall some further definitions, related to tree-like structures, used in the two-variable case. Let \( \mathfrak{A} \) be a tree-like structure. A sequence of nodes \( a_1, \ldots, a_N \in A \) is a
downward path in $\mathfrak{A}$ if for each $i$ $a_{i+1}$ is a child of $a_i$. A downward-$T_u$-path is a downward path such that for each $i$ we have $\mathfrak{A} \models T_u a_i a_{i+1}$. The $T_u$-rank of a downward-$T_u$-path $\vec{a}$, $t_u^\mathfrak{A}(\vec{a})$, is the cardinality of the set \{ $i : \mathfrak{A} \models \neg T_u a_i a_{i+1}$\}. The $T_u$-rank of an element $a \in A$ is defined as $t_u^\mathfrak{A}(a) = \sup\{t_u^\mathfrak{A}(\vec{a}) : \vec{a} = a, a_2, \ldots, a_N; \vec{a}$ is a downward-$T_u$-path\}. For an integer $M$, we say that $\mathfrak{A}$ has downward-$T_u$-paths bounded by $M$ when for all $a \in A$ we have $t_u^\mathfrak{A}(a) \leq M$, and that $\mathfrak{A}$ has transitive paths bounded by $M$ if it has downward-$T_u$-paths bounded by $M$ for all $u$. Note that a downward-$T_u$-path bounded by $M$ may have more than $M$ nodes, as the symmetric $T_u$-connections do not increase the rank.

Given an arbitrary model $\mathfrak{A}$ of a normal form UNFO+$S$ formula $\varphi$ we can simply construct its tree-like model of degree bounded by $|\varphi|$. Essentially, the construction works as in the two-variable case. We define a $\varphi$-tree-like unraveling $\mathfrak{A}'$ of $\mathfrak{A}$, together with an associated function $h : A' \to A$ in the following way. $\mathfrak{A}'$ is divided into levels $L_0, L_1, \ldots$. Choose an arbitrary element $a \in A$ and add to level $L_0$ of $A'$ an element $a'$ such that atp$^\mathfrak{A'}(a') =$ atp$^\mathfrak{A}(a)$; set $h(a') = a$. The element $a'$ will be the only element of $L_0$ and will become the root of $\mathfrak{A}'$. Having defined $L_i$, repeat the following for every $a' \in L_i$. Choose in $\mathfrak{A}$ a $\varphi$-witness structure for $\varphi(a')$. Assume it consists of $h(a'), a_1, \ldots, a_s$. Add a fresh copy $a'_i$ of every $a_j$ to $L_{i+1}$, make $\mathfrak{A}'\{}(a', a'_1, \ldots, a'_s)\text{ isomorphic to } \mathfrak{A}'\{}(h(a'), a_1, \ldots, a_s)\text{ and set } h(a'_i) = a_j$. Complete the definition of $\mathfrak{A}'$ transitivity closing all relations from $\sigma_{\text{out}}$.

As in the two-variable case we easily get the following fact.

\begin{lemma}[(1f) $\Rightarrow$ (f2)]\label{lem:19}
Let $\mathfrak{A}$ be a model of a normal form UNFO+$S$ formula $\varphi$. Let $\mathfrak{A}'$ be a $\varphi$-tree-like unraveling of $\mathfrak{A}$. Then $\mathfrak{A}' \models \varphi$ and $\mathfrak{A}'$ is a tree-like structure of degree bounded by $|\varphi|$ and transitive paths bounded by $|A|$.
\end{lemma}

\begin{proof}
It is readily verified that $\mathfrak{A}'$ meets the properties required by Lemma \ref{lem:19}. In particular $\mathfrak{h}$ is the required homomorphism. That $\mathfrak{A}'$ is tree-like and has an appropriately bounded degree is also straightforward. For the last condition assume to the contrary that there exist $u$ and a downward-$T_u$-path $(a_i)_{i=0}^N$ in $\mathfrak{A}'$ with rank bigger than $|A|$. Then there are indices $i_0, \ldots, i_{|A|}$ such that $\mathfrak{A} \models T_u a_i a_{i+1} \land \neg T_u a_{i+1} a_i$. Since $\mathfrak{h}$ preserves the connections between elements and their witnesses we have $\mathfrak{A}_0 \models T_u h(a_{i_0}) h(a_{i_0+1}) \land \neg T_u h(a_{i_0+1}) h(a_{i_0})$. By the pigeonhole principle there exist $x < x'$ such that $h(a_{i_0}) = h(a_{i_0'})$. This gives, by transitivity of $T_u$, that $\mathfrak{A}_0 \not\models T_u h(a_{i_0+1}) h(a_{i_0'})$. Contradiction.
\end{proof}

We often work with tree-like models $\mathfrak{A}$ of normal form $\varphi$ in which the downward family of every element $a$ forms a $\varphi$-witness structure. In such case we call this downward family the $\varphi$-witness structure for $a$ even if some other $\varphi$-witness structures for $a$ exist in $\mathfrak{A}$.

\subsection{E.1.2 Declarations}

Our next task is making the transition (f2) $\Rightarrow$ (f3). To this end we introduce an apparatus of declarations that allows us to perform some surgery on tree-like models of normal form formulas. Its main purpose is dealing with their universal conjuncts $\forall \bar{x} \neg \varphi_0(\bar{x})$. For a normal form $\varphi$, a $\varphi$-declaration is a description of some patterns of connections taking into account the literals of $\varphi_0$ (and, for technical reasons, some additional transitive atoms, equalities and inequalities). In particular, it may describe some dangerous patterns, leading to a violation of $\varphi_0$.

We remark that, while declarations are a counterpart of light-declarations from the previous section, they are much more complicated than the latter. One difference, worth pointing out, is that light declarations were defined as independent from the given formula: they were only dependent on the signature. In the current scenario, which patterns are
considered by declarations depends on the literals of $\varphi_0$. Making declarations independent from $\varphi$ would be possible, but would not allow us to obtain tight complexity bounds.

Let us turn to a formal definition, for which recall that $\varphi_0 = \varphi_0(x_1, \ldots, x_t)$ is in NNF, and that $2k$ is the number of transitive relations.

**Definition 20.** Let $\varphi$ be a normal form UNFO$+$S formula. Let $\mathcal{R}$ be the set consisting of all non-transitive literals (atoms or negated atoms) that appear in $\varphi_0$ (recall that only atoms which have at most one variable may be negated), $\mathcal{T} = \{1, \ldots, 2k\} \times \{1, \ldots, t\}^2$ and $\mathcal{Q} = \{1, \ldots, t\}$. A $\varphi$-declaration is a set consisting of some triples $(R, T, Q)$ such that $R \subseteq \mathcal{R}$, $T \subseteq \mathcal{T}$ and $Q \subseteq \mathcal{Q}$.

A triple $\mathcal{d} = (R, T, Q)$ may be alternatively viewed as a formula describing a pattern of connections on a tuple consisting of $t + 1$ (not necessarily distinct) elements: $\psi_\mathcal{d}(x_1, \ldots, x_t, y) = \bigwedge_{r \in R} r(\overline{x}) \land \bigwedge_{(a, j, j') \in T} T_{a, j} x_j x' \land \bigwedge_{i \in Q} x_i = y \land \bigwedge_{i \in Q} x_i \neq y$. In the sequel we often identify the triple and the formula that it represents.

Let $\mathfrak{A}$ be a tree-like structure and $a \in A$. We say that $a$ respects a $\varphi$-declaration $\mathcal{d}$ if for each $\psi \in \mathcal{d}$ we have $\mathfrak{A}_a \models \neg \exists x \psi(x, a)$. Given an element $a$ in $\mathfrak{A}$, we denote by $\text{dec}\mathcal{A}(a)$ the (unique) maximal declaration respected by $a$. Note that if $a_0$ is the root of $\mathfrak{A}$ then, knowing $\text{dec}\mathcal{A}(a_0)$, we can determine if $\mathfrak{A}_a \models \forall x \psi(\overline{x})$.

Let us now give some intuitions and describe how we are going to use declarations. We work with tree-like structures with $\varphi$-declarations assigned to all its nodes. Assigning $\mathcal{d}$ to a node $a$ of a structure $\mathfrak{A}$ may be treated as a *promise that none of the patterns described by $\mathcal{d}$ appear in $\mathfrak{A}_a$*. Note that we do not require that $\mathcal{d}$ equals $\text{dec}\mathcal{A}(a)$. Given a system of declarations assigned to all nodes of $\mathfrak{A}$ we formulate some natural local conditions such that their violation at a node $a$ breaks the promise of $a$ (i.e., some forbidden pattern occurs), and, the other way round, they are sufficient to guarantee that for every node $a$ the declaration $\mathcal{d}$ assigned to $a$ is a subset of $\text{dec}\mathcal{A}(a)$, which means that $a$ respects $\mathcal{d}$, that is, fulfills its promise. This allows us to proceed as follows: Take a tree-like model $\mathfrak{A} \models \varphi$, perform on it some surgery, obtaining a new tree-like structure $\mathfrak{A}'$. Assign to the nodes of $\mathfrak{A}'$ a system of $\varphi$-declarations in such a way that (i) the root of $\mathfrak{A}'$ gets the declaration $\text{dec}\mathcal{A}(a_0)$ where $a_0$ is the root of $\mathfrak{A}$, and (ii) for a node $a'$ its downward family $F' = \{a', a'_1, \ldots, a'_s\}$ gets the declarations $\text{dec}\mathcal{A}(a'_1), \ldots, \text{dec}\mathcal{A}(a'_s)$, where $F = \{a, a_1, \ldots, a_s\}$ is the downward family of some node $a$ from $\mathfrak{A}$, and the structures on $F$ and $F'$ are isomorphic. This guarantees that the system of declarations satisfies the local conditions and thus that its promises are fulfilled. Due to the declaration of the root we have that $\mathfrak{A}'$ satisfies the universal conjunct of $\varphi$.

We are now ready for the details. Let $F = \{a, a_1, \ldots, a_s\}$ be the downward family of a node $a$. We say that a function $f : \{1, \ldots, t\} \rightarrow \{a, a_1, a_{a_1}, \ldots, a_s, a_{a_s}\}$ is a fitting (to $F$). We think that a fitting describes a distribution of elements of a $t$-element tuple of nodes of $\mathfrak{A}_a$ among the downward family of $a$ and the subtrees rooted at the children of $a$. With a fitting $f$ we associate a function $\overline{f} : \{1, \ldots, t\} \rightarrow \{a, a_1, \ldots, a_s\}$ defined as follows: $\overline{f}(i) = a$ iff $f(i) = a$ and $\overline{f}(i) = a_j$ iff $j(i) \in \{a_j, A_{a_j}\}$. Let $b_1, \ldots, b_t = b \subseteq A_a$. The fitting $f$ induced by $b$ is defined naturally: $\overline{f}(i) = a$ iff $b_i = a$, $\overline{f}(i) = a_j$ iff $b_i = a_j$ and $\overline{f}(i) = A_{a_j}$ iff $b_i \in A_{a_j} \setminus \{a_j\}$.

Let $f$ be a fitting, $(R, T, Q)$ a tuple belonging to some declaration and $F = \{a, a_1, \ldots, a_s\}$ the downward family of some $a$. If $r$ is a literal from $R$ (resp. a tuple $(u, j, j') \in T$) then we denote by $\text{Varr}$ the set of the indices of variables of $r$ (resp. the set $\{j, j'\}$). If $r$ is a literal from $R$ or a literal $T_{a, j} x_j x'$ then we say that $r$ is fully fitted (to $F$) if $\overline{f}(\text{Varr}) \subseteq F$.

Now we define the local consistency conditions (LCCs) for a system of declarations. Consider declarations $\mathcal{d}, \mathcal{d}_1, \ldots, \mathcal{d}_s$ assigned to the elements of some family $F = \{a, a_1, \ldots, a_s\}$. We say that they satisfy LCCs at $a$ if for each fitting $\overline{f}$ and $\psi_{(R, T, Q)} \in \mathcal{d}$ at least one of the following conditions holds.
(11) Some $R$-conjunct $r \in R$ is not fully fitted, and $a \notin f(\text{Varr})$ or $a_j, a_j' \in f(\text{Varr})$ for some $j \neq j'$.

(12) Some $R$-conjunct $r \in R$ is fully fitted but $\mathfrak{A} \models \neg r(\bar{f}(1), \ldots, \bar{f}(l))$.

(13) Some $T$-conjunct $T_{i}x_{i}x_{j}$ is fully fitted but $\mathfrak{A} \models \neg T_{i}f(j)f(j')$.

(14) For some $Q$-conjunct $x_{i} = a$ we have $f(i) \neq a$.

(15) For some $(Q \setminus Q)$-conjunct $x_{i} \neq a$ we have $f(i) = a$.

(16) Some $R$-conjunct $r \in R$ is not fully fitted and is 'distributed over several subtrees', that is $\bar{f}(\text{Varr}) \cap \{a_1, \ldots, a_s\} \geq 2$.

(17) Two elements of two different subtrees cannot be transitively joined due to the structure on $F$, that is for some $T$-conjunct $T_{i}x_{i}x_{j}$, we have $f(j) \neq f(j')$ but $\mathfrak{A} \models \neg T_{i}f(j)f(j')$.

(18) All elements are fitted to a single subtree, that is for some $i$ and all $j$ we have $f(j) = a_i$, and the promise is propagated to this subtree: $(R, T, f^{-1}(a_i)) \in \mathfrak{A}$.

(19) There exists $i$ such that $a_i \in \text{Rng} \bar{f}$ and $\mathfrak{A}$ contains $(R', T', Q')$ defined as follows: fix some $h \in \{1, \ldots, t\} \setminus f^{-1}(a_i)$ and (i) $R' := \{r \in R : \text{Varr} \subseteq f^{-1}(a_i)\}$, (ii) $T'$ is the minimal set such that for $(u, j, j') \in T$ if $f(j) = f(j') = a_i$ then $(u, j, j') \in T'$ and if $f(j) = a_i$ (resp. $f(j) \neq a_i$) and $f(j') \neq a_i$ (resp. $f(j') = a_i$) then $(u, j, h) \in T'$ (resp. $(u, h, j') \in T'$), and (iii) $Q' = f^{-1}(a_i) \cup \{h\} \cup (Q \setminus f^{-1}(a_i))$.

Note that the above conditions are of two sorts. Conditions (11)–(17) describe situations in which we immediately, just looking at the structure on $F$, observe that any tuple of elements corresponding to the given fitting does not break the promise of $a$. Conditions (18)–(19), on the other hand, describe situations in which, intuitively speaking, we need to relegate such observation to one of the children of $a$.

Given a structure $\mathfrak{A}$ we say that a system of declarations $(\mathfrak{d}_a)_{a \in A}$ is locally consistent if it satisfies LCCs at each $a \in A$ and is globally consistent if $\mathfrak{d}_a \subseteq \text{dec}^\varphi(a)$ for each $a \in A$. Note that the global consistency means that the promises of all nodes are fulfilled. Conditions (11)–(17) are tailored so that local and global consistency play along in the following lemma, which is a natural counterpart of Lemma 3.

**Lemma 21 (Local-global).** Let $\mathfrak{A}$ be a tree-like structure. Then (i) if a system of declarations $(\mathfrak{d}_a)_{a \in A}$ is locally consistent then it is globally consistent (ii) the canonical system of declarations $(\text{dec}^\varphi(a))_{a \in A}$ is locally consistent.

**Proof.** (i) Assume to the contrary that there exist $a \in A$, $\psi \in \mathfrak{d}_a$ and $\bar{b} \subseteq A_a$ such that $\mathfrak{A}_a \models \psi(\bar{b}, a)$. Take the fitting $\bar{f}$ to the downward family $F = \{a, a_1, \ldots, a_s\}$ of $a$ induced by $\bar{b}$. By the choice of $\bar{b}$, none of (11)–(17) holds. Thus $\bar{b} \subseteq F$ and there exist some $a_i \in F \setminus \{a\}$ and $\psi' \in \mathfrak{d}_{a_i}$ such that $\mathfrak{A}_{a_i} \models \psi'(\bar{b'}, a_i)$ where $\bar{b'} = b'_1, \ldots, b'_t$ is defined as follows: $b'_j = b_j$ if $f(j) = a_i$, otherwise $b'_j = a_i$. Denote by depth($\bar{b}$) the maximal level of $\mathfrak{A}$ inhabited by an element of $\bar{b}$. Obviously depth($\bar{b}$) $\leq$ depth($\bar{b}$). Thus after finitely many steps we get $\bar{a}^*$, $\psi^* \in \mathfrak{d}_{a^*}$ and $\bar{b}^*$ contained in the downward family $F^*$ of $a^*$ such that $\mathfrak{A}_{a^*} \models \psi(\bar{b}^*, a^*)$. But this cannot happen, since neither (18) nor (19) can hold for the fitting to $F^*$ induced by $\bar{b}^*$.

(ii) Follows from a careful inspection of the definition of LCCs. Basically, if for some $a$, $\psi$ and $\psi \in \text{dec}^\varphi(a)$ none of (11)–(17) holds then we can find $\bar{b} \subseteq A_a$ such that $\mathfrak{A} \models \psi(\bar{b}, a)$. Indeed, use non-satisfaction of (18)–(19) to find fragments of $\bar{b}$ belonging to the respective subtrees $(\bar{b} \cap \mathfrak{A}_a)$ (i.e., to find appropriate $j \in \mathfrak{A}_a$ for $j \in f^{-1}(a_i)$); non-satisfaction of (11)–(17) implies that they are connected so that $\psi(\bar{b}, a)$ holds. ▶

### E.1.3 Shortening transitive paths

To make transition $(2) \rightsquigarrow (3)$ we proceed, essentially, as in the two-variable case, that is we perform the same tree pruning process, with the same pruning strategy, with $\varphi$-declarations
playing now the role of light declarations. Correctness of this approach is justified mainly by Lemma 21 being a counterpart of Lemma 3. The only difference is the way we argue that a model obtained in the pruning process satisfies the universal conjunct of the given formula. In the two-variable case it was done by constructing an appropriate homomorphism into the original model. In the presence of more than two variables it is not always possible. Instead, it is sufficient to use the fact that the system of declarations on the produced model is globally consistent and look at the declaration of its root.

A reader understanding the tree pruning process in the two-variable case may thus safely skip this section, noting only the above-mentioned difference in dealing with the \( \forall \)-conjunct in the correctness proof.

For completeness, we give here a detailed description, presenting it, however, in a slightly more compact way than in the two-variable case.

**Lemma 22 ((\( \forall \omega \)) \( \rightarrow \) (f3)).** If a normal form UNFOS formula \( \varphi \) has a tree-like model of degree bounded linearly and transitive paths bounded by some natural number then it has a tree-like model of degree bounded linearly and transitive paths bounded doubly exponentially in \(|\varphi|\).

**Proof.** Let \( \mathfrak{A} \models \varphi \) be a tree-like model with bounded transitive paths. Let \( M_\varphi := |\alpha| \cdot |D_\varphi| + 2 \), where \( \alpha \) is the set of atomic 1-types over the signature of \( \varphi \) and \( D_\varphi \) is the set of \( \varphi \)-declarations. Clearly, \( M_\varphi \) is bounded doubly exponentially in \(|\varphi|\).

Consider a mapping: \( A \ni a \mapsto g(a) \). Observe that \( |\text{Rng}| \leq M_\varphi - 2 \). We construct a tree-like model \( \mathfrak{A}' \) having levels \( L'_0, L'_1, \ldots \). During our construction we maintain a pattern function \( p : A' \to A \) and a function \( s : A' \to \text{Sym}(\{1, \ldots, 2k\}) \) whose purpose is to define some order of shortening paths at a given node. Intuitively, for \( s(a) = \tau \), if \( v < v' \) then we prefer to shorten \( T_{\tau(v)} \) over \( T_{\tau(v')} \).

Let \( L'_0 \) consist of \( a'_0 \) — a copy of the root \( a_0 \) of \( \mathfrak{A} \) (i.e. \( \text{atp}^{\mathfrak{A}}(a'_0) = \text{atp}^{\mathfrak{A}}(a_0) \)). Put \( p(a'_0) = a_0 \) and set \( s(a'_0) \) arbitrarily. Suppose that we have defined \( L'_i \). For each \( a' \in L'_i \) let \( \{p(a'), a_1, \ldots, a_s\} \) be the downward family of \( p(a') \) in \( \mathfrak{A} \) and let \( s(a') = \tau \). Take fresh copies \( a'_j \) of \( a_j \) and make \( \mathfrak{A}'(\{a'_1, a'_2, \ldots, a'_s\}) \) isomorphic to \( \mathfrak{A}(\{p(a'), a_1, \ldots, a_s\}) \).

Presently we set the \( p(a'_j) = K := \{v : \mathfrak{A} \models -T_{\tau(v)}p(a')a_j\} \) (the \( T_{\tau(v)} \) killed at \( a'_j \)), \( S(a'_j) = S := \{v : \mathfrak{A} \models T_{\tau(v)}p(a')a_j \land T_{\tau(v)}a_jp(a')\} \) (the \( T_{\tau(v)} \) sustained at \( a'_j \)) and \( D(a'_j) = D := \{v : \mathfrak{A} \models T_{\tau(v)}p(a')a_j \land -T_{\tau(v)}a_jp(a')\} \) (the \( T_{\tau(v)} \) diminished at \( a'_j \)).

In this paragraph we refer to them without the argument, as it is always the same. If \( D \neq \emptyset \) then let \( v_D(a'_j) = v_D := \min D \) and take as \( p(a'_j) \) a \( b_j \in A_{a_j} \) such that (i) \( g(b_j) = g(a_j) \) (ii) for all \( v < v_D, v \in S : r^{a_j}_{\tau(v)}(b_j) \leq r^{a'_j}_{\tau(v)}(p(a'_j)) \) (iii) \( r^{a_j}_{\tau(v)}(b_j) \) is the lowest possible among the elements satisfying (i) and (ii). Note that such an element exists (\( a_j \) satisfies (i) and (ii); for (iii), the ranks in \( \mathfrak{A} \) are bounded, in particular finite) and \( r^{a_j}_{\tau(v)}(b_j) < r^{a'_j}_{\tau(v)}(p(a'_j)) \). If \( D = \emptyset \) then let \( p(a'_j) = a_j \). If \( K \neq \emptyset \) then let \( v_K = \min K \) and set \( s(a'_j) := \tau' \) where \( \tau' \) is defined as follows: for \( v < v_K \) let \( \tau'(v) = \tau(v) \), for \( v_K \leq v < 2k \) let \( \tau'(v) = \tau(v + 1) \) and let \( \tau'(2k) = \tau(v_K) \). Otherwise (that is \( K = \emptyset \)) put \( s(a'_j) = \tau \). To finish the construction, transitively close all the appropriate relations in \( \mathfrak{A}' \).

We claim that \( \mathfrak{A}' \) constructed as above is a model of \( \varphi \) and has the desired properties.

\( \forall \exists \)-conjuncts are satisfied since for all \( a' \in A' \) the structure on the downward family of \( a' \) in \( \mathfrak{A}' \) is isomorphic to the structure on the downward family of \( p(a') \) in \( \mathfrak{A} \) and the latter is the \( \varphi \)-witness structure for \( p(a') \).

For the universal conjunct of \( \varphi \) consider the system of declarations \( \{\text{dec}^\mathfrak{A}_2(p(a'))\}_{a' \in A'} \).

Note that in this system the declarations on the downward family of any node \( a' \) in \( A' \) are copies of the declarations on the downward family of \( p(a') \) in \( \mathfrak{A} \) in the canonical system of
We conclude this section by showing that for finitely satisfiable formulas we can always make an auxiliary estimation.

Claim 23. Let \( v_0 \) and a downward-\( T_u \)-path \( \vec{a} = (a_i)_{i=1}^N \) in \( \mathfrak{A}' \) be such that for all \( v \leq v_0 \) we have that the map \( i \mapsto s(a_i)(v) \) is constant (in this case, slightly abusing notation, we write \( s(\vec{a})(v) = s(a_1)(v) \)). Let \( u := s(\vec{a})(v_0) \). Then \( r_{T_u}^\mathfrak{A}'(\vec{a}) \leq (M_\varphi + 1)(\sum_{v < v_0} r_{T_u}^\mathfrak{A}'(\vec{a}(v))) + M_\varphi \).

Proof. Assume to the contrary that there is a downward-\( T_u \)-path \( \vec{a} \) in \( \mathfrak{A}' \) meeting the required conditions such that \( r_{T_u}^\mathfrak{A}'(\vec{a}) > (M_\varphi + 1)(\sum_{v < v_0} r_{T_u}^\mathfrak{A}'(\vec{a}(v))) + M_\varphi \). Then there are more than \( M_\varphi(1 + \sum_{v < v_0} r_{T_u}^\mathfrak{A}'(\vec{a}(v))) \) indices \( s \) such that \( v_0 = v_D(a_i) \). So there exist indices \( i_1, \ldots, i_M_\varphi \) such that for all \( j \) it holds that \( v_D(a_{i_j}) = v_0 \) and for all \( i_1 \leq i \leq i_M_\varphi \) and \( v < v_0 \) we have \( v \notin D(a_{i_j}) \) (and thus \( v \in S(a_{i_j}) \)). It follows that for all \( v \leq v_0 \) the function \( i \mapsto r_{T_u}^\mathfrak{A}'(p(a_i)) \) is non-increasing (on \( \{i : i_1 \leq i \leq i_M_\varphi\} \)) and the function \( j \mapsto r_{T_u}^\mathfrak{A}'(p(a_{i_j+1})) \) is strictly decreasing. By the pigeonhole principle there exist \( x < x' < M_\varphi \) satisfying \( g(p(a_{i_x+1})) = g(p(a_{i_{x+1}})) \). This contradicts the choice of \( p(a_{i_x+1}) \).

The above claim allows us in particular to compute a (uniform) doubly exponential bound on \( r_{T_u}^\mathfrak{A}'(\vec{a}) \) for all \( v_0, u \) and \( \vec{a} \) as in assumption. Denote this bound by \( \overline{M_\varphi} \).

Consider now any downward-\( T_u \)-path \( \vec{a} = (a_i)_{i=1}^N \) in \( \mathfrak{A}' \). For each node \( a_i \) from \( \vec{a} \) let \( v_{a_i}(a_i) \) be such that \( s(a_i)(v_{a_i}) = u \). Due to the strategy that we use to define \( s(v) \) the value of \( v_{a_i} \) is non-increasing along \( \vec{a} \). Indeed, when moving from \( a_i \) to \( a_{i+1} \) the value of \( v_{a_i} \) is either unchanged or decreases by 1; the only chance of increasing it would be to change it to \( 2k \) but this happens only when \( T_u \) is killed. Let us divide \( \vec{a} \) into fragments \( \vec{a}_1, \vec{a}_2, \ldots \) on which \( v_{a_i} \) is constant. The number of such fragments is obviously bounded by \( 2k \). On each of such fragments \( \vec{a}_i \) for all \( v \leq v_{a_i} \) we have that \( s(\vec{a}_i)(v) \) is constant. So we can apply Claim 23 to bound \( r_{T_u}^\mathfrak{A}'(\vec{a}_i) \) by \( \overline{M_\varphi} \). This gives the desired doubly exponential bound \( \overline{M_\varphi} = 2k\overline{M_\varphi} \) on \( r_{T_u}^\mathfrak{A}'(\vec{a}) \). This finishes the proof of Lemma 22.

E.1.4 Regular tree-like models

We conclude this section by showing that for finitely satisfiable formulas we can always construct regular tree-like models with bounded transitive paths. We recall that this step was unnecessary in the two-variable case and was ommitted there.

Let us introduce a tool, which allows us to verify the property of having bounded transitive paths looking only at some local conditions. Let \( \mathfrak{A} \) be a tree-like structure with root \( a_0 \). Then a function \( S : A \to \{0, \ldots, M\} \) is the \( (T_u, M) \)-stopwatch labeling if: \( S(a_0) = 0 \); for every \( a \in A \) and its child \( b \): (i) if \( \mathfrak{A} \models T_uab \wedge T_uab \) then \( S(b) = S(a) \), (ii) if \( \mathfrak{A} \models T_uab \wedge \neg T_uab \) then \( S(b) = S(a) + 1 \) (in particular \( S(a) < M \) ) (iii) if \( \mathfrak{A} \models \neg T_uab \) then \( S(b) = 0 \).

It is easy to see that the value of the \( (T_u, M) \)-stopwatch labeling in \( a \) is equal to the maximal rank of a downward-\( T_u \)-path ending in \( a \), therefore the \( (T_u, M) \)-stopwatch labeling exists iff the structure has downward-\( T_u \)-paths bounded by \( M \).
Lemma 24 ((13)⇒(14)). If $\varphi$ has a tree like model with linearly bounded degree and doubly exponentially bounded transitive paths, then it has a regular such model, that is a model with doubly exponentially many non-isomorphic subtrees.

Proof. Let $\mathfrak{A}$ be a tree-like model of $\varphi$ with linearly bounded degree and transitive paths bounded doubly exponentially by $\hat{M}_\varphi$. For each $\mathfrak{T}_u$ take the $(\mathfrak{T}_u, \hat{M}_\varphi)$-stopwatch labeling $\mathcal{S}_u$ of $\mathfrak{A}$. Consider the mapping $A \ni a \mapsto (\hbox{atp}(a), \hbox{dec}(a), (\mathcal{S}_u(a))^t_{u=1})$. Note that $|\hbox{Rng}|$ is bounded doubly exponentially in $|\varphi|$. We rebuild $\mathfrak{A}$ into a regular model $\mathfrak{A}'$.

For each $p \in \hbox{Rng}$ choose a representative $c(p) \in g^{-1}(p)$. Add to $L'_0$ an element $a'_0$ such that atp$\mathfrak{A}'(a'_0) = \hbox{atp}(a_0)$, where $a_0$ is the root of $\mathfrak{A}$, and let $p(a'_0) = c(g(a_0))$. Having defined $L'_i$, for $i \geq 0$, repeat the following for all $a' \in L'_i$. Denoting $p(a'), a_1, \ldots, a_s$ the downward family of $p(a')$ in $\mathfrak{A}$, add a fresh copy $a'_i$ of each $a_i$ to $L'_{i+1}$ and make $\mathfrak{A}' \upharpoonright \{a', a'_1, \ldots, a'_s\}$ isomorphic to $\mathfrak{A} \upharpoonright \{p(a'), a_1, \ldots, a_s\}$. Set $p(a'_i) := c(g(a_i))$. Finally, transitively close all transitive relations in $\mathfrak{A}'$.

The proof that $\mathfrak{A}' \models \varphi$ is similar to the corresponding proof in Lemma 22, we observe that all elements have appropriate $\varphi$-witness structures copied from $\mathfrak{A}$ and then use the apparatus of declarations to argue that the $\forall\exists$ conjunct of $\varphi$ is respected. By construction $\mathfrak{A}'$ is a regular tree-like model with the number of different subtrees bounded by $|\hbox{Rng}|$. To see that $\mathfrak{A}'$ has transitive paths bounded by $\hat{M}_\varphi$ create stopwatch labelings for $\mathfrak{A}'$ just by transferring them from $\mathfrak{A}$ using $p$. It is not difficult to see that they meet the conditions from the definition of stopwatch labelings.

E.1.5 A few words for automata fans

The content of Section E.1 could be alternatively presented with help of (Büchi) tree automata, however we decided not to present it this way. The reasons will be explained in a moment, after explaining how automata could be used. This comment should not be treated as a formal description (and does not bother with details), but rather as a glossary of terms one after explaining how automata could be used. Equivalently, we could design $\mathcal{A}_3$ with the set of states being the set of all possible declarations (the accepting states) plus the Black Hole (BH) (the rejecting state), and transitions described by the LCCs, plus two additional ones: anyone can go to the BH, and no one escapes the BH. The starting state can be defined using the description in the 'For the universal conjunct...' paragraph in the proof of Lemma 22. Now, Lemma 24 is a counterpart of the statement that $\mathcal{A}_2$ accepts exactly the models of the $\forall$ conjunct. Lemma 22 can be stated as: if there exists $M$ such that $L(\mathcal{A}_1 \cap \mathcal{A}_2 \cap \mathcal{A}_3(M)) \neq \emptyset$ then there exists such $M$ doubly exponential in $|\varphi|$. Finally, Lemma 24 and Theorem 10 can be treated as some standard properties of automata. So, why we decided not to use automata? Firstly, since it was not the way we originally thought about the problem. Secondly, and mainly, it would not allow us to omit the main obstacles, namely the definitions of the declarations, and the proofs of Lemmas 21 and 22. Furthermore, we wanted to keep our presentation uniform, and not to alternate between logic and automata for this (not the most complicated) task. And, finally, we would not need the full power of Büchi automata.
E.2 Building small finite models

In this section we show the following small model property. To this end we make the missing transition (f4) ~ (f5). Generally, our constructions here are quite similar to the constructions from the two-variable case. The main differences are: bigger components, using the isomorphism types of subtrees of a regular tree-like model instead of generalized types, and a more complicated way of building witness structures. Also, the proof of correctness of the construction becomes now more complicated, in particular, it uses non-trivially the regularity of tree-like models from the previous subsection.

\[\text{Lemma 24.}\]

\[\text{Every finitely satisfiable UNFO+S formula } \varphi \text{ has a finite model of size bounded triply exponentially in } |\varphi|.\]

Let us fix a finitely satisfiable normal form UNFO+S formula \( \varphi \) over a signature \( \sigma_{\text{univ}} \cup \sigma_{\text{dist}} \) for \( \sigma_{\text{dist}} = \{T_1, \ldots, T_{2k}\} \). Recall that we consider structures that for each \( 1 \leq u \leq k \) interpret the transitive symbol \( T_{2u} \) as the inverse of \( T_{2u-1} \) and we sometimes write \( T_{2u}^{-1} \) for \( T_{2u-1} \) and \( T_{2u-1}^{-1} \) for \( T_{2u} \). For a set \( \mathcal{E} \subseteq \sigma_{\text{dist}} \), closed under inverses, and \( a \in A \) we denote by \([a]_{\mathcal{E}}\) the set consisting of \( a \) and all elements \( b \in A \) such that \( A \models T_{\alpha}ab \) for all \( T_\alpha \in \mathcal{E} \). Note that \([a]_{\mathcal{E}}\) is either a singleton or each of the \( T_u \in \mathcal{E} \) is total on \([a]_{\mathcal{E}}\), that is, for each \( b_1, b_2 \in [a]_{\mathcal{E}} \) we have \( A \models T_{\alpha}b_1b_2 \) for all \( T_\alpha \in \mathcal{E} \). We assume that \([a]_{\mathcal{E}} = A\). Fix a regular tree-like model \( A \models \varphi \), with linearly bounded degree, doubly exponentially bounded transitive paths (in this section we denote this bound by \( \hat{M}_F \)) and doubly exponentially many non-isomorphic subtrees, as in Lemma 24.

We show how to build a ‘small’ finite model \( A' \models \varphi \). In our construction we inductively produce fragments of \( A' \) in which all relations from some subset of \( \sigma_{\text{dist}} \), that is closed under inverses, are total (or, for some technical reasons, such a fragment may be a singleton and then these relations do not need to be total). The induction is, essentially, over the number of the non-total \( T_u \). Intuitively, if a relation is total then it plays no important role, so we may forget about it during the construction. On the \( l \)-th level of induction we produce a substructure for every isomorphism type of a subtree of \( A \) and any (closed under taking inverses) combination of \( 2l \) non-total transitive relations, so that its each element has provided its partial \( \varphi \)-witness structure. This partial \( \varphi \)-witness structure is an isomorphic copy of the restriction of the \( \varphi \)-witness structure (in \( A \)) for some \( a \in A \) to the set \([a]_{\mathcal{E}_l}\), where \( \mathcal{E}_l \) is the set of current total transitive relations. Every substructure in the \( l \)-th level of induction is constructed by an appropriate arrangement of some number of basic building blocks, called components. Each of the components is obtained by some number of applications of the inductive assumption to situations in that one of the non-total transitive relations and its inverse are added to the set of total ones.

An important property of the substructures created during our inductive process is that they admit some partial homomorphisms to the pattern tree-like model \( A \) which restricted to (partial) witness structures act as isomorphisms into the corresponding parts of the \( \varphi \)-witness structures in \( A \). We impose that every homomorphism respects the required condition using directly the structure of \( A \). To this end we introduce further fresh (non-transitive) binary symbols \( W^i \) whose purpose is to relate elements to their witnesses. We number the elements of the \( \varphi \)-witness structures in \( A \) arbitrarily (recall that each element is a member of its own \( \varphi \)-witness structure) and interpret \( W^i \) in \( A \) so that for each \( a, b \in A \), \( A \models W^iab \) iff \( b \) is the \( i \)-th element of the \( \varphi \)-witness structure for \( a \) (from now, for short, we refer to the element \( b \) satisfying \( W^iab \) as the \( i \)-th witness for \( a \)). Do this in such a way that if two subtrees of \( A \) were isomorphic before interpreting the \( W^i \) then they still are after such expansion of the structure. Now, if we mark \( b \) as the \( i \)-th witness for \( a \) during the construction (that is set
Finite Satisfiability of Unary Negation Fragment with Transitivity

For each $p$, for each $b$, for each $a$, is implied by Condition (b4) (since $b$, as the structures $W_i$ has cardinality 1. If this is not the case, we simply add artificial transitive relations. Moreover, if $a'_0 \in a$, then we can choose $\bar{h}$ so that $\bar{h}(a'_0) = a_0$.

(b5) For each $a \in A_0$, we have $\mathcal{M}_a \cong \mathcal{M} \upharpoonright A_0$ where $\mathcal{M}$ is the $\varphi$-witness structure for $p(a)$. (Note that, by the definition of the $W^i$, each such isomorphism sends a to $p(a)$.)

Before we prove Lemma 26 let us observe that it indeed allows us to build a particular finite model of $\varphi$. Apply Lemma 26 to $\mathcal{E}_0 = \mathcal{E}$ (which means that $\mathcal{E}_0 = \emptyset$ and $a_0$ being the root of $\mathcal{A}$ (which means that $\mathcal{A}_0 = \mathcal{A}$). We use Lemma 3 to see that the obtained structure $\mathcal{A}'_0$ is a model of $\varphi$. Indeed, Condition (a1) of Lemma 1 follows directly from Condition (1.5), as the structures $\mathcal{M}_a$ from (1.5) are full $\varphi$-witness structures in this case, Condition (a2) is implied by Condition (1.4) (since $\bar{a} \subseteq \mathcal{M}_a$ and $\bar{h}|W_a$ is an isomorphism, $\bar{h}|\bar{a}$ preserves 1-types).

The proof of Lemma 26 goes by induction on $l = |\mathcal{E}_0|/2$. In the base of induction, $l = 0$, we have $\mathcal{E}_\text{tot} = \mathcal{E}_0$. Without loss of generality we may assume that for each $a \in A$ the set $[a]_{\sigma_{\text{dist}}}$ has cardinality 1. If this is not the case, we simply add artificial transitive relations $T_{2k+1}$ and $T_{2k+2}$ and interpret them as the identity in $\mathcal{A}$ (this way they also satisfy the requirement of being each other’s inverses). We simply take $\mathcal{A}'_0 := \mathcal{A}_0 = \mathcal{A} \upharpoonright \{a_0\}$ and set $p(a_0) = a_0$. It is readily verified that the conditions (1.1)–(1.5) are then satisfied.

For the inductive step assume that Lemma 26 holds for arbitrary closed under inverses set of size $2(l-1)$. We show that then it holds for (closed under inverses) $\mathcal{E}_0$ of size $2l$. Without loss of generality we assume that $\mathcal{E}_0 = \{T_1, \ldots, T_{2l}\}$. In the next two subsections we present a construction of $\mathcal{A}'_0$ and then, in the following subsection, we argue that it is correct. Finally we estimate the size of the produced models and establish the complexity of the finite satisfiability problem.

E.2.1 Pattern components

We plan to construct $\mathcal{A}'_0$ out of basic building blocks called components. Each component will be an isomorphic copy of some pattern component. Let $\gamma[A_0]$ be the set of isomorphism types of subtrees of $\mathcal{A}$ rooted at $A_0$. Note that this way we overload the notation $\gamma[A_0]$ which in the two-variable case meant the set of generalized types realized in $\mathcal{A}_0$. For every $\gamma \in \gamma[A_0]$ we construct a pattern component $\mathcal{C}_\gamma$ as well as the extended pattern component...
\( \mathcal{G}^\gamma \). An important difference, compared to the two-variable case, is that components in this section will have more layers.

The extended pattern component, \( \mathcal{G}^\gamma \), is a finite structure whose universe is divided into \( 2(2l + 1) \) inner layers, \( L_1, \ldots, L_{2(2l+1)} \), and a single interface layer, denoted \( L_{2(2l+1)+1} \). The structure \( \mathcal{C}^\gamma \) is obtained by the restriction of \( \mathcal{G}^\gamma \) to its inner layers. Each inner layer \( L_i \) is further divided into sublayers \( L_{i,1}, L_{i,2}, \ldots, L_{i,M_i+1} \). Additionally, in each sublayer \( L_{i,j} \) its initial part \( L_{i,j}^{\text{init}} \) is distinguished. In particular, \( L_{1,1}^{\text{init}} \) consists of a single element called the root. The interface layer \( L_{2(2l+1)+1} \) has no internal division but, for convenience, is sometimes referred to as \( L_{1,1}^{\text{init}} \). The elements of \( L_{2(2l+1)+1} \) are called leaves and the elements of \( L_{2(2l+1)+1} \) are called interface elements. The structure of components is similar to the structure of components for the two-variable case, as depicted in Fig. 1, just recall that the number of layers is now bigger.

\( \mathcal{G}^\gamma \) (and \( \mathcal{C}^\gamma \)) will have a shape resembling a tree, with structures obtained by the inductive assumption as nodes, though it will not be tree-like in the sense of Section 3.4 (in particular, the internal structure of nodes may be very complicated). All elements of \( \mathcal{G}^\gamma \), except for the interface elements, will have appropriate partial \( \varphi \)-witness structures provided.

We remark that during the process of building a pattern component we do not yet apply the transitive closure to the distinguished relations. Postponing this step is not important from the point of view of the correctness of the construction, but will allow us for a simpler presentation of the proof of its correctness. Given a pattern component \( \mathcal{C} \) we will sometimes denote by \( \mathcal{C}_u \) the structure obtained from \( \mathcal{C} \) by applying all the appropriate transitive closures.

The crucial property we want to enforce is that the root of \( \mathcal{C}^\gamma \) will be far from its leaves in the following sense. Denote by \( G_t(\mathcal{G}) \), for a \( \sigma \)-structure \( \mathcal{S} \), the Gaifman graph of the structure obtained by removing from \( \mathcal{G} \) the relations belonging to \( \mathcal{E}_{\text{tot}} \). Then there is no connected induced subgraph of \( G_t(\mathcal{C}^\gamma) \) of size \( t \) containing an element of one of the first \( 2l \) layers and, simultaneously, an element of one of the last \( 2l \) layers of \( \mathcal{C}^\gamma \).

The role of every inner layer \( L_i \) is, speaking informally, to kill one of the \( T_u \)-connections from \( L_i \) to \( L_{i+1} \). See the right part of Fig. 1. The role of sublayers, on the other hand, is to decrease the \( T_u \)-rank of elements. The purpose of the interface layer, \( L_{2(2l+1)+1} \), is to join the connect the component with other components.

Now we proceed to the construction of \( \mathcal{G}^\gamma \). If \( \gamma = \gamma_{a_0} \) then take \( a = a_0 \); otherwise take any element \( a \in A_0 \) that is the root of a subtree of type \( \gamma \). Define \( L_{1,1}^{\text{init}} = \{a'\} \) for a fresh \( a' \), setting \( atp^{\mathcal{G}^\gamma}(a') = atp^{\mathcal{G}}(a) \) and \( p(a') = a \).

Construction of an inner layer: Let \( 1 \leq i \leq 2(2l+1) \). Assume we have defined layers \( L_1, \ldots, L_{i-1} \), the initial part of sublayer \( L_{i,1}^{\text{init}} \), and both the structure of \( \mathcal{G}^\gamma \) and the values of \( p \) on \( L_1 \cup \ldots \cup L_{i-1} \cup L_{i,1}^{\text{init}} \). Let \( v = 1 + (i - 1 \mod 2l) \). We are going to kill \( T_v \).

We now expand \( L_{1,1}^{\text{init}} \) to full layer \( L_i \).

**Step 1: Subcomponents.** Assume that we have defined sublayers \( L_1^{\text{init}}, \ldots, L_{i,1}^{\text{init}}, \) and both the structure of \( \mathcal{G}^\gamma \) and the values of \( p \) on \( L_1 \cup \ldots \cup L_{i-1} \cup L_{1}^{\text{init}} \cup \ldots \cup L_{i,1}^{\text{init}} \). For each \( b \in L_{i,1}^{\text{init}} \) perform independently the following procedure. Apply the inductive assumption to \( p(b) \) and the set \( \mathcal{E}_0 \setminus \{T_v, T_v^{-1}\} \) obtaining a structure \( \mathcal{B}_0 \), its origin \( b_0 \) and a function \( \mathsf{p}_b : B_0 \to A_p(b) \cap [p(b)]_{\mathcal{E}_0 \cup \{T_v, T_v^{-1}\}} \subseteq A_0 \) with \( \mathsf{p}_b(b_0) = p(b) \). Identify \( b_0 \) with \( b \) and add the remaining elements of \( \mathcal{B}_0 \) to \( L_i \), retaining the structure. Substructures \( \mathcal{B}_0 \) of this kind will be called subcomponents (note that all appropriate relations are transitively closed in subcomponents). Extend \( p \) so that \( p|B_0 = p_b \). This finishes the definition of \( L_i \).

**Step 2: Providing witnesses.** For each \( b \in L_i \) independently perform the following procedure. Let \( \mathcal{B}_0 \) be the subcomponent created inductively in Step 1, such that \( b \in B_0 \). Let \( \mathcal{W} \) be the
Figure 6 Providing witnesses. Thick arrows denote $T_v$-connections.

Let $\mathfrak{A} = \mathfrak{M}[[p(b)]]_{E_{tot}}$ and $\mathfrak{F} = \mathfrak{M}[[p(b)]]_{E_{tot} \cup \{T_v, T_v^{-1}\}}$. Note that $\mathfrak{F}$ is a substructure of $\mathfrak{A}$. By (i) $b$ has the partial $\varphi$-witness structure $\mathfrak{F}'$, isomorphic to $\mathfrak{F}$, provided in $\mathfrak{B}_0$. Extend $\mathfrak{F}'$ to an isomorphic copy $\mathfrak{F}'$ of $\mathfrak{E}$. The structure $\mathfrak{E}'$ will be the structure $\mathfrak{M}_0$ in $\mathfrak{G}'$ and then in $\mathfrak{A}_0$. The elements of $E' \setminus F'$ are fresh, and are assigned their sublayers as follows. For $c \in E' \setminus F'$ if $\mathfrak{E}' \models T_v bc$ (observe that in this case $\mathfrak{E}' \models \neg T_v cb$) then add $c$ to $L_i^{1,init}$, otherwise add $c$ to $L_i^{1,init}$. See Fig. 6. Take as the values of $p(E' \setminus F')$ the corresponding elements of $E \setminus F'$.

An attentive reader may be afraid that when adding witnesses for elements of the last sublayer $L_i^{M_{\varphi}+1}$ of $L_i$ we may want to add one of them to the non-existing layer $L_i^{M_{\varphi}+2}$. There is however no such danger, which follows from the following claim.

**Claim 27.** (i) Let $b \in L_i^{1,init}$ and let $\mathfrak{B}_0$ be the subcomponent created for $b$ in Step 1. Then for all $b' \in \mathfrak{B}_0$ we have $\tau_{E}^{3}(p(b)) \geq \tau_{E}^{3}(p(b'))$. (ii) Let $b \in L_i^{1}$ and let $\mathfrak{E}', \mathfrak{F}'$ be the partial $\varphi$-witness structures for $b$ considered in Step 2. Then for any $c \in E' \setminus F'$ such that $\mathfrak{E}' \models T_v bc$ (so $c \in L_i^{1+1}$) the inequality $\tau_{E}^{3}(p(b)) > \tau_{E}^{3}(p(c))$ holds.

**Proof.** (i) By the inductive assumption applied to $\mathfrak{B}_0$, $p(b') = p(b') \in \mathfrak{A}_p(b')[[p(b')]]_{E_{tot} \cup \{T_v, T_v^{-1}\}}$. This means that any downward-$T_v$-path (in $\mathfrak{A}$) starting in $p(b')$ can be extended to one starting in $p(b)$. Therefore $\tau_{E}^{3}(p(b)) \geq \tau_{E}^{3}(p(b'))$.

(ii) By the choice of $\mathfrak{E}'$ and $\mathfrak{F}'$ we have $\mathfrak{E}' \models T_v bc \land \neg T_v cb$, thus by the choice of $p$ we have $\mathfrak{A} \models T_v p(b)p(c) \land \neg T_v p(c)p(b)$ and finally $\tau_{E}(p(b)) > \tau_{E}(p(c))$. \hfill \endproof

Hence, when moving from $L_i^{1}$ to $L_i^{1+1}$ the $T_v$-ranks of pattern elements for the elements of these sublayers strictly decrease. Since these ranks are bounded by $M_{\varphi}$, then, even if the $T_v$-ranks of the patterns of some elements of $L_i^{1}$ are equal to $M_{\varphi}$, then, if $L_i^{M_{\varphi}+1}$ is non-empty, the $T_v$-ranks of the patterns of its elements must be 0, which means that they cannot have witnesses connected to them one-directionally by $T_v$.

The construction of $\mathfrak{G}'$ is finished when the interface layer, $L_{2(2i+1)+1}$ is defined (recall that it has only its ‘initial part’).

### E.2.2 Joining the components

In this section we take some number of copies of extended pattern components and arrange them into the desired structure $\mathfrak{A}_0$, identifying interface elements of some components with
the roots of some other. Some care is needed in this process in order to avoid any modifications of the internal structure of closures $\mathcal{E}_p$ of components $\mathcal{E}$, which could potentially result from transitivity of relations. In particular we need to ensure that if for some $u$ a pair of elements of a component $\mathcal{E}$ is not connected by $T_u$ inside $\mathcal{E}$, then it will not become connected by a chain of $T_u$-edges external to $\mathcal{E}$.

We create a pattern component $\mathcal{C}^\gamma$ together with its extension $\mathcal{G}^\gamma$ for every $\gamma \in \gamma[A_0]$. Let $\text{max}$ be the maximal number of interface elements across all the $\mathcal{G}^\gamma$. For each $\mathcal{G}^\gamma$ we number its interface elements arbitrarily using the numbers from 1 up to, potentially, $\text{max}$.

For each $\gamma \in \gamma[A_0]$ we take copies $\mathcal{G}^{\gamma,g}$ of $\mathcal{G}^\gamma$ for $g \in \{0, 1\}$ ($g$ is often called a color; it is advised to think that if an extended component is of color $g$, then the elements of the inner layer of the component are of color $g$, while the elements of the interface layer are of color $1 - g$ since the latter will become identified with some roots of components of color $1 - g$), $1 \leq i \leq \text{max}$ and $\gamma' \in \gamma[A_0]$, together with the previously chosen numbering of the interface elements. We also take an additional copy $\mathcal{G}_{1,1}^{\gamma,0}$ of $\mathcal{G}^{\gamma,0}$. Its root will become the origin of the whole $\mathcal{A}_0$.

For each $\gamma$, $g$ consider components of the form $\mathcal{G}^{\gamma,g}$. Perform the following procedure for each $i$—the number of an interface element. Let $b$ be the $i$-th interface element of any such component, let $\gamma'$ be the type of $\mathcal{A}_{p(b)}$. Identify the $i$-th interface elements of all $\mathcal{G}^{\gamma,g}$ with the root $c_0$ of $\mathcal{G}^{\gamma,1-g}_{1,1}$. See Fig. 7.

Note that the values of $p(c_0)$ and $p(b)$ (the latter equals to the value of $p$ on the $i$-th interface element in all the $\mathcal{G}^{\gamma,g}$) may differ. However, by construction, $\mathcal{A}_{p(b)} \cong \mathcal{A}_{p(c_0)}$ (in particular, the $1$-types of $b$ and $c_0$ match). For the element $c^*$ obtained in this identification step we define $p(c^*) = p(c_0)$.

For the extended component $\mathcal{G}^{\gamma}_{i,\gamma'}$, denote its restriction to its interface elements (being naturally a copy of the pattern component $\mathcal{C}^\gamma$). Define the graph of components used in the above construction, $G^{\text{comp}}$, by joining two components by an edge iff we identified an interface element of extension of one of them with the root of the other. Take $\mathcal{A}_0^0$ as the structure restricted to the components accessible from $\mathcal{C}^{\gamma,0}_{1,1}$ in $G^{\text{comp}}$. Note that in $\mathcal{A}_0^0$ we still do not take the transitive closures of relations. We define $\mathcal{A}_0^0$ by transitively closing all appropriate relations in $\mathcal{A}_0^0$. Later we will keep using the convention of marking some auxiliary structures in which the transitive closures are not yet applied with the superscript 0. Finally, we choose as the origin $a_0'$ of $\mathcal{A}_0^0$ the root of the pattern component $\mathcal{C}^{\gamma,0}_{1,1}$.

### E.2.3 Correctness of the construction

1. By the construction, after taking the transitive closures, on each (extended) pattern component either all $\mathcal{E}_{\text{tot}}$ relations are total or it consists of one element. Next observe that if two extended components get joined, then at least one of them has cardinality greater than 1 and all $\mathcal{E}_{\text{tot}}$ relations, after taking the transitive closure, are total on their sum. So, by the definition of the graph of components $G^{\text{comp}}$, all $\mathcal{E}_{\text{tot}}$ relations are total on $\mathcal{A}_0^0$ of it consists of a single element.

2. As $a_0'$ we take the root of $\mathcal{C}^{\gamma,0}_{1,1}$. Recall that we explicitly map the root of the pattern component $\mathcal{C}^{\gamma,0}_{1,1}$ by $p$ to $a_0$.

3. The interpretations of the $W^a_i$ are defined in the step of providing witnesses where, implicitly, we take care of this condition for every element $a'$ of the inner layers by extending the fragment of the partial $\varphi$-witness structure for $a'$ created on the previous level of induction by a copy of a further fragment of the same pattern $\varphi$-witness structure. Note also that during the step of joining the components all the interface elements become identified with
some roots, which are elements of inner layers, and that the identifications do not spoil the required property.

\[ \text{For simplicity, let us ignore the ‘moreover’ part of this condition for some time. We will explain how to take care of it near the end of this proof. Now we find a homomorphism } h \text{ such that } \mathfrak{A}_{p(a)} \cong \mathfrak{A}_{h(a)} \text{ for all } a \in \bar{a} \text{ (we say that such a homomorphism has the subtree isomorphism property). Later we will show that its restrictions to the substructures } \mathfrak{M}_a \text{ are indeed isomorphisms. The proof starts with several homomorphic reductions which show that instead of } \mathfrak{A}_0 \text{ we can consider a structure looking like a pattern component but twice as high.} \]

\textbf{Reduction 0.} First observe that if } \mathfrak{A}_0 \text{ consists of just one element, then the structure } \mathfrak{A}'_0 \text{ consists of one element and the only map } h : A'_0 \to A_0 \text{ is the required homomorphism. For the rest of the proof of } \textit{we assume the } A_0 \text{ has cardinality at least } 2, \text{ in particular all } E_{\text{tot}} \text{ relations are total on it. Consider a tuple } \bar{a} \subseteq A'_0 \text{ such that } |\bar{a}| \leq t. \text{ Observe that for each } a \in \bar{a} \text{ the structure } \mathfrak{M}_a \text{ is connected in } G_T(\mathfrak{A}_0' \mid W_\bar{a}) \text{ (recall the definition of Gaifman graph } G_T(\mathfrak{S}) \text{ and the interpretation of the symbols } W'). \text{ Let } \mathfrak{M}_{a_1}, \ldots, \mathfrak{M}_{a_t} \text{ be the connected components of } \mathfrak{M}_a \text{ in } G_T(\mathfrak{A}_0' \mid W_\bar{a}). \text{ If we have homomorphisms } h_i : \mathfrak{M}_{a_i} \to \mathfrak{A}_0 \text{ satisfying the subtree isomorphism property then we can take } h = \bigcup h_i : \mathfrak{M}_a \to \mathfrak{A}_0 \text{ which is a homomorphism, since all } E_{\text{tot}} \text{ relations are total on } \mathfrak{A}_0, \text{ that still has the subtree isomorphism property. Owing to this reduction we can restrict attention to tuples } \bar{a} \text{ with } \mathfrak{M}_a \text{ connected (in the above sense).} \]

\textbf{Reduction 1.} By the construction, for all } 1 \leq i \leq 2(2t+1) \text{ and } v = 1 + (i - 1 \mod 2t), \text{ there is no } T_v \text{-path in any component from an element of } L_i \text{ to an element of } L_{i+1}. \text{ Thus, if we divide (inner) layers of components into groups of size } 2t, \text{ a transitive path may join at most elements of two neighboring groups. Obviously, non-transitive relations join only tuples consisting of elements of at most two consecutive layers, and, in particular, each of the } \mathfrak{M}_a \text{ lies in at most two consecutive layers. It follows, that given a connected } \mathfrak{M}_a, |\bar{a}| \leq t, \text{ by our choice of the number of layers in a component, there exists } g \in \{0, 1\} \text{ such that removing all the connections between leaves of color } 1 - g \text{ and roots of color } g \text{ (in other words: any connections between elements of } L_{2(2t+1)} \text{ and elements of } L_{2(2t+1) + 1} \text{ in components of color } 1 - g \text{ does not remove any connections among the elements of } \mathfrak{M}_a. \]

More formally, let } \mathfrak{D}_0' \text{ be the structure obtained from } \mathfrak{A}_0' \text{ by removing all the connections as described above, and let } \mathfrak{D}_0 \text{ be the transitive closure of } \mathfrak{D}_0'. \text{ Then the inclusion map } \iota : \mathfrak{M}_a \to \mathfrak{D}_0' \text{ is a homomorphism. Clearly } \mathfrak{A}_{p((a))} \cong \mathfrak{A}_{p(a)} \text{ since } a = \iota(a). \text{ Thus we can restrict attention to a tuple } \bar{a} \text{ for which } \mathfrak{M}_a \text{ is connected and search for a homomorphism } \mathfrak{M}_a \to \mathfrak{A}_0 \text{ treating } \mathfrak{M}_a \text{ as a substructure of } \mathfrak{D}_0'. \]

\textbf{Reduction 2.} Observe that by our scheme of arranging the copies of pattern components there is at most one type } \gamma \text{ of components of color } g \text{ (where } g \text{ is the color from the previous reduction) that contains some element of a connected } \mathfrak{M}_a \text{ (consider the shape of a connected fragment of the graph of components } G_{\text{comp}} \text{ with connections between leaves of color } g \text{ and roots of color } 1 - g \text{ removed). Furthermore, all elements of } \bar{a} \text{ of color } 1 - g \text{ are contained in components of the form } \mathfrak{C}^{\gamma=1-g}. \text{ Choose one component of type } \gamma \text{ of color } g \text{ and call it } \mathfrak{C}^\gamma. \text{ Consider the structure } \mathfrak{C}_0^\gamma \text{ (resp. } \mathfrak{C}_0^\gamma) \text{ obtained as the restriction of the structure } \mathfrak{D}_0^\gamma \text{ from the previous reduction to the union of the domains of the components of the form } \mathfrak{C}^{\gamma=1-g} \text{ (resp. the domain of } \mathfrak{C}^\gamma) \text{ and the domains of all the components of the form } \mathfrak{C}^{\gamma=1-g}. \text{ Let } \mathfrak{C}_0^\gamma \text{ (resp. } \mathfrak{C}_0^\gamma) \text{ be their transitive closures. Consider a projection } \pi \text{ that projects all elements of } \mathfrak{C}_0^\gamma \text{ of color } g \text{ onto } \mathfrak{C}^\gamma \text{ and is the identity on the others. We claim that } \pi \mid W_\bar{a} : \mathfrak{M}_a \to \mathfrak{C}_0^\gamma \text{ is a homomorphism. To see this, observe that the paths connecting elements of } \mathfrak{M}_a \text{ in } \mathfrak{C}_0^\gamma \text{ are contained in } E_0^\gamma \text{ and } \pi : \mathfrak{C}_0^\gamma \to \mathfrak{C}_0^\gamma \text{ is a homomorphism. See Fig. 7. Clearly for each } a \in \bar{a} \text{ we} \]
have $\mathfrak{A}_{p(a)} = \mathfrak{A}_{p(\pi(a))}$ since $p(a) = p(\pi(a))$. Thus, finally, we can restrict attention to a tuple $\bar{a}$ for which $\mathfrak{M}_{\bar{a}}$ is connected and search for a homomorphism $\mathfrak{M}_{\bar{a}} \rightarrow \mathfrak{A}_0$ treating $\mathfrak{M}_{\bar{a}}$ as a substructure of $\mathfrak{S}_0$.

**Essential homomorphism construction.** Note that $\mathfrak{S}_0$ looks like a single component but is twice as high. Consider the tree of subcomponents of $\mathfrak{S}_0$, $\tau$, defined as follows: make a subcomponent $B$ the parent of $B'$ if $B'$ contains a witness for an element of $B$. Observe that so obtained $\tau$ is indeed a tree. For a subcomponent $B \in \tau$ denote by $B^\land$ the union of domains of subcomponents belonging to the subtree of $\tau$ rooted at $B$.

Since we might have cut some connections between an element and some of its witnesses during Reduction 1, we define for each $a \in F'_0$ the surviving part $V_a$ of $W_a$ by $V_a = \mathfrak{S}_0|V_a$ where $V_a = \{b : \exists i \mathfrak{S}_0 \models W'_a b\}$. For a tuple $\bar{b}$ denote $V_{\bar{b}} = \bigcup_{b \in \bar{b}} V_b$ and $V_{\bar{b}} = \mathfrak{S}_0|V_{\bar{b}}$. Note that $V_a \subseteq W_a$, and generally, this inclusion may be strict, but for all $a \in \bar{a}$ we have $\mathfrak{M}_a = \mathfrak{M}_{\bar{a}}$, and thus, in particular, the claim below finishes the proof of the currently considered part of (b4), that is the proof of the existence of a homomorphism satisfying the subtree isomorphism property.

Returning to the shape of $\mathfrak{S}_0$, it consists of some subcomponents arranged into tree $\tau$ glued together by the structure on the surviving parts of witness structures. Note that all such building blocks (that is both the subcomponents and the surviving parts of the partial witness structures) are transitively closed. Moreover, by the tree structure of $\tau$, if some elements of such a building block are connected by some atom in $F'_0$, then they already have been connected by the same atom in $\mathfrak{S}_0$, therefore the identity map from $\mathfrak{S}_0^\lor$ to $\mathfrak{S}_0^\lor$ acts as an isomorphism when restricted to such a building block.

> **Claim 28.** For every subcomponent $B_0 \in \tau$ with origin $b_0$, and $\bar{a} \subseteq B_0^\lor$, $|\bar{a}| \leq t$, there exists a homomorphism $h : \mathfrak{M}_{\bar{a}} \rightarrow \mathfrak{A}_{p(b_0)}[p(b_0)]_{\text{Ext}}$ such that for all $a \in \bar{a}$ we have $\mathfrak{A}_{h(a)} \cong \mathfrak{A}_{p(a)}$, and if $b_0 \in \bar{a}$ then $h(b_0) = p(b_0)$. 

![Figure 7](image-url) Joining the components and Reductions 1 and 2. Elements connected by dashed lines are identified.
Finite Satisfiability of Unary Negation Fragment with Transitivity

Proof. Bottom-up induction over tree.

Induction base: If $\mathfrak{B}_0$ is a leaf of $\tau$ then $\mathfrak{B}_0 \subseteq \mathfrak{B}_0$ and the claim follows by the inductive assumption of Lemma 26 (note that here we implicitly use the fact that the identity map is an isomorphism between $\mathfrak{B}_0[0]B_0$ and $\mathfrak{B}_0[0]B_0$).

Induction step: Let $\mathfrak{B}_1, \ldots, \mathfrak{B}_K$ be the list of all the children of $\mathfrak{B}_0$ in $\tau$ such that $B_i^\infty$ contains some element of $a$. If $K = 1$ and $a \subseteq B_i^\infty$ the thesis follows from the inductive assumption of this claim.

Otherwise, for $1 \leq i \leq K$ (note that it is possible that $K = 0$), denote by $b_i$ the origin of $\mathfrak{B}_i$ and let $c_i \in \mathfrak{B}_i$ be such that $b_i$ is a witness chosen by $c_i$ in the step of providing witnesses or during the step of joining the components. By the inductive assumption of this claim there exist homomorphisms $h_i : \mathfrak{B}_i[0]B_i^\infty \rightarrow \mathfrak{B}_i[0]B_i$ such that $h_i(b_i) = p(b_i)$. From the inductive assumption of Lemma 26 we have a homomorphism $h_0 : \mathfrak{B}_0[0]B_0[0]B_0 \rightarrow \mathfrak{B}_0[0]B_0$ such that $h_0(b_0) \in \mathfrak{B}_0[0]B_0$. We extend it in the only possible way to $h_0^*$ defined on the whole $\mathfrak{B}_0[0]B_0[0]B_0$. For each $a \in \mathfrak{B}_0$ we have $h_0^* \left| \mathfrak{B}_0[a] \right. = h_0[0]B_0$. We will prove that $h_0^* \left| \mathfrak{B}_0[a] \right. = h_0[0]B_0$ for the appropriate $a$.

Now we show that $\mathfrak{B}_1, \ldots, \mathfrak{B}_K$ are the witnessing objects. For each $a \in \mathfrak{B}_0$ we have $\mathfrak{B}_0[0]B_0[0]B_0[0]B_0 \rightarrow \mathfrak{B}_0[0]B_0$ such that $h_0^* \left| \mathfrak{B}_0[a] \right. = h_0[0]B_0$. We will prove that $h_0^* \left| \mathfrak{B}_0[a] \right. = h_0[0]B_0$ for the appropriate $a$.

We will consider the inductive assumption of Lemma 26 (note that here we implicitly use the fact that the identity map is an isomorphism between $\mathfrak{B}_0[0]B_0$ and $\mathfrak{B}_0[0]B_0$). Since $h_0^* \left| \mathfrak{B}_0[a] \right. = h_0[0]B_0$, by the inductive assumption of this claim we can conclude that for each $a \in \mathfrak{B}_0$ we have $h_0^* \left| \mathfrak{B}_0[a] \right. = h_0[0]B_0$.

We naturally join $h_0^*, h_1^*, \ldots, h_K^*$ into $h : \mathfrak{B}_0 \rightarrow \mathfrak{B}_0$ with $h_0^*(b_i) = b_i'$. Note that such $h$ is well defined, even though the value of $h$ on each of the $b_i$ is defined twice, since $b_i$ belongs to both $\text{Dom} h_0$ and $\text{Dom} h_i$ (has been defined on the other elements exactly once). For each $a \in \text{Dom} h_i$ (for $i \geq 0$) we have $\mathfrak{B}_0[a] = h_0[0]B_0 \left| \mathfrak{B}_0[a] \right. = h_0[0]B_0 \left| \mathfrak{B}_0[a] \right. = \mathfrak{B}_0[a]$, by the inductive assumptions of this claim and Lemma 26. Since $a \subseteq \text{Dom} h_0 \cup \bigcup_{i > 0} \text{Dom} h_i$, we can conclude that for each $a \in \mathfrak{B}_0$ we have $h_0^* \left| \mathfrak{B}_0[a] \right. = h_0[0]B_0$.

The fact that $h$ is a homomorphism follows from the tree structure of $\tau$. In particular, there cannot be any connections (before taking the transitive closures) between (non-origin) elements of two different $B_i^\infty$ (for $1 \leq i \leq K$). The full proof that $h$ is a homomorphism is tedious, therefore we show two representative cases that use all the major ideas required. First, consider $a, a' \in V_{ab \cdots B_k \cdots c_k}$ such that $a \in B_i^\infty$, $a' \in B_j^\infty$ for some $i, j$ such that $c_i \neq c_j$. Assume that $\mathfrak{B}_0[a] = T_0a$ for some $u$. We will prove that $\mathfrak{B}_0[a] = T_0a$. By a standard argument, owing to the tree structure of the $\tau$ (some more care is needed since there may be some connections in the structures $\mathfrak{B}_0$), there exist $d_j \in V_{c_j} \cap B_0$ and $d_i \in V_{c_i} \cap B_0$ such that $\mathfrak{B}_0[a] = T_0a$, $\mathfrak{B}_0[V_{c_i}] = T_0b_i$, $\mathfrak{B}_0[V_{c_j}] = T_0b_j$, $\mathfrak{B}_0[V_{a}] = T_0b_i d_i$ and $\mathfrak{B}_0[V_{a}] = T_0b_j d_j$ (we assumed that $a, b_i, d_i, b_j, d_j, a'$ are pairwise different; otherwise some parts of such path become trivial). Since $h_0^* h_0$ and $h_i^*$ are homomorphisms, $\mathfrak{B}_0[a] = T_0a \cup T_0b_i d_i \cup T_0b_j d_j$ and therefore by the choice of $h_0^* h_0$, $\mathfrak{B}_0[a] = T_0a \cup T_0b_i d_i$ and by the choice of the extension of $h_0$ to $h_0^* h_0$, $\mathfrak{B}_0[a] = T_0b_i d_i$. Let $b_i^*$ be the $i$-th witness of $\mathfrak{B}_0[a]$. We now show that $\mathfrak{B}_0[a] = T_0b_i d_i$.
and \( d^p_i \) be the \( i_{th} \)-th witness of \( p(c_i) \). By construction, \( \mathcal{A}_0 \models T_a b^p_i d^p_i \). But \( \mathcal{A}_{p(c_i)} \cong \mathcal{A}_{b(c_i)} \) and by the uniqueness of the numbers of the witnesses, any isomorphism between these subtrees sends \( b^p_i \) to \( h(b_i) \) and \( d^p_i \) to \( h(d_i) \), therefore \( \mathcal{A}_0 \models T_a h(b_i) h(d_i) \). Similarly \( \mathcal{A}_0 \models T_a h(d_j) h(b_j) \).

Joining the pieces together, by transitivity of \( T_a \), \( \mathcal{A}_0 \models T_a h(b_i) h(a') \). Joining the case when \( \mathcal{A}_0 \models R(a') \) for some non-transitive symbol \( R \) and \( a' \subseteq V_{\bar{a}b_1 \ldots b_K c_1 \ldots c_K} \). By construction, \( R(a') \) was set either during in the process of building some subcomponent or during the step of providing witnesses. Thus \( a' \) is either contained in \( B_0 \) or one of the \( B^K_{\bar{a}} \) or one of the \( V_a \) for some \( a \in \bar{a}b_1 \ldots b_K c_1 \ldots c_K \). Now we can prove, using arguments similar to ones used for appropriate parts of the path in the previous case, that \( \mathcal{A}_0 \models R(h(a')) \).

Since by construction \( h_0 \subseteq h \), if \( b_0 \in \bar{a} \) then \( h_0(b_0) = h_0(b_0)(= p(b_0) \) by the inductive assumption of Lemma [26]. To finish the inductive step, we restrict \( h \) to \( V_{\bar{a}} \).

Now we prove the additional property required for \( h \) by (1), that is, that for each \( a \in \bar{a} \), \( h|W_a \) is an isomorphism. By the numbering of witnesses, as explained before the statement of this lemma, \( h \) moves \( W_a \) into the part of the witness structure of \( h(a) \) contained in \( A_0 \) and is one-to-one by the uniqueness of the numbers of witnesses in a witness structure. The other way around, we can use a similar argument as in the first case presented in the proof that the map built in Claim [28] is a homomorphism. That is, if for some \( \bar{a}' \subseteq W_a \) and some (arbitrary) relation \( R \), \( \mathcal{A}_0 \models R(h(\bar{a}')) \), then, since \( \mathcal{A}_{h(a)} \cong \mathcal{A}_{p(a)} \) and any isomorphism preserves the numbering of witnesses and the structure on \( W_a \) was copied from a part of the witness structure for \( p(a) \) (together with such numbering), \( \mathcal{A}_0 \models R(\bar{a'}) \) and therefore the inverse of \( h|\mathcal{W}_a \) is also a homomorphism, so \( h|\mathcal{W}_a \) is an isomorphism.

Now we return to the ‘moreover’ part of (1). Let us assume that \( a'_0 \in \bar{a} \). We will slightly modify the above proof. Reductions 0 and 1 do not move \( a'_0 \) and we keep them unchanged. Notice that in Reduction 1 we have that \( g = 0 \). Now, in Reduction 2 we have that \( \gamma = \gamma_{a_0} \) and we choose \( c^? = c_{1,0}^\gamma \). This way application of \( \pi \) does not move \( a'_0 \). To finish the proof, it is sufficient to see that by Claim [28] \( h(a'_0) = p(a'_0) = a_0 \).

Apply (1) to a tuple consisting of just \( a \) to obtain an isomorphism \( h : \mathcal{W}_a \to \mathcal{A}_0[h(W_a)] \).
and then apply an isomorphism between $\mathfrak{A}_{0(a)}$ and $\mathfrak{A}_{p(a)}$.

### E.2.4 Size of models and complexity

To complete the proof of Thm. 25 we need to show an appropriate upper bound on the size of finite models produced by our construction. The following routine estimation shows that $|A'_0|$ is triply exponential in $n = |\varphi|$, regardless of the choice of the initial tree-like model $\mathfrak{A}$. We calculate a bound $S_{2l}$ on the size of the structure obtained in the proof of Lemma 26 for $|E_0| = 2l$. We are interested in $S_{2k+2}$, which is the desired bound on the size of $\mathfrak{A}_0$ (we use $S_{2k+2}$ here, rather than $S_{2k}$, because we may potentially introduce the auxiliary identity relation in the base step of induction). By the construction any pattern component is a tree of subcomponents consisting of at most $2((2t+1)(M_{\varphi}+1))$ sublayers (so, also this is a bound on the depth of the tree). In the sublayer of depth $1$ we have at most $S_{2t-2}$ elements, in the sublayers in the second one—at most $S_{2t-2}n$ subcomponents; this jointly gives $S_{2t-2}^n$ elements. Iterating, we have at most $S_{2t-2}^n l^{l-1}$ elements in the sublayers of depth $i$, which jointly gives an estimate $(S_{2t-2}n)^{2((2t+1)(M_{\varphi}+1))}$ on both the number of inner elements and the number of interface elements in a pattern component. Multiplying it by the number of components used in the joining phase, and then estimating $t$ and $l$ in the exponent by $n$ and $n+1$ respectively, we get a bound $S_{2l} = 2|\gamma[A]|^2(S_{2t-2}n)^{2((n+1)(2n+1)(M_{\varphi}+1)+1)}$. Solving this recurrence relation, and recalling that $M_{\varphi}$ and $|\gamma[A]|$ are doubly exponential in $|\varphi|$ we obtain a triply exponential bound on $S_{2t+2}$.

This finishes the proof of Thm. 25. We do not know if our construction is optimal with respect to the size of models. The best we can do for the lower bound is to enforce models of doubly exponential size (actually, it can be done in UNFO even without transitive relations).

Thm. 25 immediately gives the decidability of the finite satisfiability problem for UNFO+ S and suggests a simple 3-\textsc{NExpTime}-procedure: convert a given formula $\varphi$ into normal form $\varphi'$, guess a finite structure of size bounded triply exponentially and verify that it is a model of $\varphi'$. We can however do better and show a doubly exponential upper bound matching the known complexity of the general satisfiability problem. The following theorem has already been stated in the main body of this paper.

**Theorem 29 (restating of Thm. 10).** The finite satisfiability problem for UNFO+ S is 2-\textsc{ExpTime}-complete.

**Proof.** The lower bound is inherited from pure UNFO 27 or from UNFOS (Thm. 9).

For the upper bound, we describe an algorithm in A\textsc{ExpSpace}. Fix $\varphi$ in normal form. We have proved that $\varphi$ has a finite model if it has a tree-like model with doubly exponentially bounded transitive paths (as in Lemma 22). We will look for the latter. We advise the reader to recall the proof of Lemma 24 as we presently use a similar apparatus. In our procedure we produce, in an alternating fashion, a finite tree $\mathfrak{A}^*$, corresponding to some number of the upper levels of a model. Simultaneously, we define a function $g^*$ returning an element of $A^*$ its 1-type together with some $\varphi$-declaration and one stopwatch for each of the $T_u$ (cf. the proof of Lemma 24).

More precisely, let $M_{\varphi}$ be the bound on transitive paths obtained in Lemma 22 and $M$ be a bound on $|\text{Rng}^*|$ (we use $(T_u, M_{\varphi})$-stopwatches in $g^*$). The alternating algorithm works as follows. Calculate $M_{\varphi}$ and $M$. Note that both are doubly exponential in $|\varphi|$. Construct the root of $\mathfrak{A}$ and guess its 1-type $\alpha$, a $\varphi$-declaration $\mathfrak{d}$ containing all the formulas of the form $\varphi^0_1(x) \land A_{i \in Q} x_i = y \land \bigwedge_{i \in Q \setminus Q} x_i \neq y$ for any $Q \subseteq Q$ and $1 \leq j \leq z$ (recall that $\varphi^0$ is equivalent to $\varphi^0_1 \lor \ldots \lor \varphi^0_n$ with the $\varphi^0_n$ being conjunctions of some $\mathcal{R}$ and $\mathcal{T}$ formulas).
Set $g^*(a) = (\alpha, \varnothing, (0)_{a=1}^k)$. Now construct the downward family of $a$, $F = \{a, a_1, \ldots, a_s\}$, for some $s < |\varphi|$, guess its (transitively closed) structure, and guess the values $g^*(a_1), \ldots, g^*(a_s)$.

Check whether $F$ is a $\varphi$-witness structure for $a$, the 1-types assigned by $g^*$ agreed with the structure, the declarations assigned by $g^*$ satisfy the LCCs and the stopwatches assigned by $g^*$ satisfy the local condition described in the definition of $(T_u, \hat{M}_F)$-stopwatch labeling. If not, reject. Next universally choose one of the $a_i$. Then proceed as for $a$—guess the downward family of $a_i$ and values of $g^*$, and check their consistency as above, universally choose one of the children of $a_i$ and so on. We additionally keep a counter containing the number of the current level in $\mathcal{A}^*$. If it reaches $M + 1$, we accept.

It is clear that the described algorithm can be implemented in $\text{AExpSpace}$: we only need to store the structure and the values of $g^*$ on a single family, plus a counter. All of these can be written using exponentially many bits.

**Correctness proof.** To see that if $\varphi$ has a model $\mathcal{A}$ with bounded transitive paths then the algorithm accepts, it is sufficient to make the guesses in accordance with $\mathcal{A}^*$—the structure induced on the first $M + 1$ levels of $\mathcal{A}$ with $g^*$ defined as follows $A^* \ni a \mapsto (\text{atp}^\mathcal{A}(a), \text{dec}^\mathcal{A}(a), (\mathcal{S}_u)_{u=1}^{2k})$, where $\mathcal{S}_u$ is the $(T_u, \hat{M}_F)$-stopwatch labeling of $\mathcal{A}$. The fact that such a strategy leads to an accepting run of the algorithm is almost straightforward. In particular, the local consistency of declarations follows from Lemma~[21](ii). The opposite implication uses ideas similar to the ones from the proof of Lemma~[24]. Assume that the algorithm has an accepting run. From this run we can naturally infer a tree-like structure $\mathcal{A}^*$ consisting of $M + 1$ levels, and a function $g^*$. Note that on each path from the root to a leaf in $\mathcal{A}^*$ some value of $g^*$ appears at least twice. Cut each branch at the first position on which the value of $g^*$ reappears and make a link from this point to the first occurrence of this value on the considered branch. Naturally unravel so obtained structure into an infinite tree-like structure $\mathfrak{A}$. Define on $\mathfrak{A}$ function $g$ just copying the values of $g^*$. We show that $\mathfrak{A} \models \varphi$ and has transitive paths bounded by $\hat{M}_F$. Note that the downward families in $\mathfrak{A}$ and the values of $g$ on them are copies of some downward families in $\mathcal{A}^*$ and their values of $g^*$, so each $a \in A$ has a $\varphi$-witness structure (\mathfrak{A} satisfies all the $\forall\exists$-conjuncts of $\varphi$) and also $g$ gives a locally consistent set of declarations and $(T_u, \hat{M}_F)$-stopwatch labelings. The latter guarantee that $\mathfrak{A}$ has bounded transitive paths; the former, together with the choice of the declaration $\varnothing$ for the root of $\mathfrak{A}^*$, allows us to conclude that $\mathfrak{A}$ satisfies the $\forall$-conjunct of $\varphi$.

As remarked in the Introduction, we can state our results in a slightly stronger way, for a setting in which we may not only require some binary symbols to be interpreted as arbitrary transitive relations, but we can, more specifically, require some of them to be equivalences and some other—partial order. Indeed, assuming that $T_u$ is transitive we can enforce it in UNFO to be a (strict) partial order, writing $\neg \exists xy(T_u xy \land T_u yx)$. Non-strict partial orders can be then simulated by disjunctions $T_u xy \lor x = y$. An equivalence relation can be simulated by some $T_u$ by replacing every usage of $T_u xy$ by $T_u xy \land T_u^{-1} xy \lor x = y$ (and then ignoring the non-symmetric interpretations of $T_u$; we remark that it is not possible to enforce in $\text{UNFO} + S T_u$ to be interpreted as an equivalence [1]).

**Corollary 30.** The finite satisfiability problem for $\text{UNFO}$ with transitive relations, equivalences and partial orders is $2^{\text{ExpTime}}$-complete.

We note that our approach does not allow us to deal with linear orders. Actually, the presence of a strict linear order $<$ makes the satisfiability problem for UNFO undecidable, as it allows for a reduction from UNFO with inequalities, which is known to be undecidable [27]: $x \neq y$ can be then expressed as $x < y \lor y < x$. See also [1]. To the best of our knowledge,
the decidability of the (finite) satisfiability problem for UNFO with non-strict linear orders is open.

Capturing expressive description logics

F.1 Constants

To show a small model property, and establish the decidability of UNFO+ S with constants, UNFO+ SO, we are not going to design any new transformations of models. We will just use Thm. 22 and Thm. 10. Our plan is to simulate constants with freshly introduced unary predicates. Such predicates will be called pseudoconstants. Of course, our transformations of models from Section F do not respect the uniqueness of interpretations of pseudoconstants. We thus introduce a simple quotient construction which given a structure, for every pseudoconstant, shrinks all its interpretations into a single element. There is a potential danger here: the shrinking operation may lead to some new patterns of connections. E.g., if a a sends a T a-edge to an interpretation of a pseudoconstant, c, and b receives a T a from another incarnation c′ of the same pseudoconstant, then in the resulting model, a and b become T a-connected, even though they need not be T a-connected in the original model. To make this operation safe we will perform some manipulations on the input formula.

Pseudoconstants. Let ϕ be a UNFO+ S formula with constants c1,...,cK. Take a set of fresh unary symbols σconst := {C1,...,CK} and let ϕconst be a formula obtained from ϕ by simulating constants with symbols from σconst. Namely, for every relational symbol R, and every atom R(x1,...,xs,c1,...,cs), where the xi are its variables and the ci are its constants, we replace this atom with ∃yi1...yisR(x1,...,xs,y1,...,ys), where the yi are fresh variables. Note that ϕconst remains a UNFO+ S formula.

Let EXI := ∨i xiCi ∈ (there exists an interpretation of every pseudoconstant) and UNI := ∀xy(Ci x ∧ Ci y → x = y) (every pseudoconstant is uniquely interpreted). Notice that EXI is in UNFO, but UNI is not. Clearly, ϕ is (finitely) satisfiable iff ϕconst ∧ EXI ∧ UNI is (finitely) satisfiable. Furthermore, assuming that ϕconst is a normal form of ϕconst, due to Lemma 15 we can check (finite) satisfiability of ϕconst ∧ EXI ∧ UNI instead of ϕ.

So, w.l.o.g., we will consider the finite satisfiability problem for formulas of the form ϕ ∧ EXI ∧ UNI, where ϕ is a UNFO+ S formula in normal form, over some signature σ = σbase ∪ σaux ∪ σconst, with σconst consisting of auxiliary unary relation symbols.

Shrinking. Let us now introduce our shrinking operation. Let A be a structure. Define a relation ∼ on A by setting a ∼ b if A |= ϵ(a,b), where ϵ(x,y) = (x = y) ∨ (∃i Xi x ⇔ Ci y) (x and y correspond to the same constant or they are the same non-constant). Observe that ∼ is an equivalence relation and that ϵ is in UNFO. Let A0 := A/∼ and q : A → A0 be the corresponding quotient map. If A |= R a for some relation symbol R and a tuple a, then put A0 |= Rq(a). Let A be the result of applying the transitive closure to all the T a in A0. Observe that q : A → A is a homomorphism, A |= UNI and if A |= UNI then q : A → A is an isomorphism. A triple of the form (A, A, q) will be called a shrinking triple. We will sometimes refer to the intermediate model A0 (as usual, in some arguments involving transitive connections in A corresponding to paths in A0).

Modifications of ϕ. Now we introduce a modification of the given formula ϕ. Let ∀¬¬xϕ(x) be the universal conjunct of ϕ. Note that a Tu-connection may appear in the shrinking A of a given model A either as a direct copy of some 2-type from A or as the transitive closure of a Tu-path that goes through some constants. To capture this second possibility we replace every atom T u xz′ in ϕ0 by τu(x, z′) := ∃x0,y1,...,xK,ε(x0,ε(y1,x1)) ∧ ∨k=1,...,K ε(yk,xk) ∧

∀¬¬xϕ(x)
Proof. (i) $\varepsilon(y_{K+1}, z') \wedge T_u x_0 y_1 \wedge \bigwedge_{i=1}^K (T_u x_i y_{i+1} \vee x_i = y_{i+1})$. We also replace any non-transitive atom in $R z_1 \ldots z_n$ in $\varphi_0$ by $\rho_R(z_1, \ldots, z_n) := \exists x_1 \ldots x_n (\bigwedge_i \varepsilon(x_i, z_i) \wedge R x_1 \ldots x_n)$. Let us denote by $\varphi^\text{path}$ the formula resulting from such operations on $\varphi_0$, and by $\varphi^\text{path}$ the result of substituting the $\forall$-conjunction of $\varphi$ with $\forall \bar{x} \neg \varphi_0^\text{path} (\bar{x})$. Note that $\varphi^\text{path}$ is (equivalent to) a UNFO+$\mathcal{S}$ formula.

For technical reasons we introduce two additional formulas. Let CON be a formula saying that the $\sim$-equivalent elements have the same 1-types. Let TYPE be a formula whose aim is to prevent the 1-types from enlarging after the application of the shrinking operation, $\text{TYPE} := \bigwedge_u \forall x (\neg T_u x x \rightarrow \neg \tau_u(x, x))$. Clearly both CON and TYPE can be treated as UNFO+$\mathcal{S}$ formulas.

Observe that for any structure $\mathfrak{A}$, if $\mathfrak{A} \models \text{CON}$ then there are at most $K$ equivalence classes of the relation $\sim$ containing some pseudoconstants.

Let us collect some basic properties of our transformation.

\textbf{Claim 31.} Let $\mathfrak{A}$ be an arbitrary structure. Then for any $a, b \in A$, $\bar{a} \subseteq A$, non-transitive symbol $R$ and transitive symbol $T_u$

(i) If $\mathfrak{A} \models T_u ab$ then $\mathfrak{A} \models \tau_u(a, b)$.
(ii) If $\mathfrak{A} \models R \bar{a}$ then $\mathfrak{A} \models \rho_R(\bar{a})$.
(iii) If $\mathfrak{A} \models \text{TYPE}$ then $\mathfrak{A} \models T_u \bar{a} \bar{a}$ iff $\mathfrak{A} \models \tau_u(a, a)$.
(iv) If $\mathfrak{A} \models \text{CON}$ then $\mathfrak{A} \models R a \ldots a$ if $\mathfrak{A} \models \rho_R(a, a, \ldots, a)$.
(v) If $\mathfrak{A} \models \text{CON} \wedge \text{TYPE}$ and $\mathfrak{A} \models \varphi_0^\text{path}(\bar{a})$ then $\mathfrak{A} \models \varphi_0(\bar{a})$.
(vi) In particular, if $\mathfrak{A} \models \text{CON} \wedge \text{TYPE}$ and $\mathfrak{A} \models \varphi^\text{path}$ then $\mathfrak{A} \models \varphi$.

\textbf{Proof.} (i) If $\mathfrak{A} \models T_u ab$ then substitute $a$ for $x_0$ and $b$ for all the other quantified variables in $\tau_u(a, b)$ to obtain that $\mathfrak{A} \models \tau_u(a, b)$.
(ii) Substitute the existentially quantified variables in $\rho_R(\bar{a})$ with $\bar{a}$.
(iii) $\Rightarrow$ follows from (i), $\Leftarrow$ is exactly TYPE.
(iv) $\Rightarrow$ follows from (ii), $\Leftarrow$ We have then some $a' \in A$ such that $a \sim a'$ and $\mathfrak{A} \models Ra a' \ldots a'$, so, from the fact that $\mathfrak{A} \models \text{CON}$ it follows that $\mathfrak{A} \models Ra a' \ldots a$.
(v) $\varphi_0$ is in UNFO+$\mathcal{S}$, so its non-unary atoms cannot be negated. So by (i)–(iv) the thesis follows.
(vi) The formulas $\varphi$ and $\varphi^\text{path}$ differ only on their $\forall$-conjuncts. So the thesis follows, since by (v), if $\mathfrak{A} \models \forall \bar{x} \neg \varphi_0(\bar{x})$ then $\mathfrak{A} \models \forall \bar{x} \neg \varphi_0^\text{path}(\bar{x})$.

\textbf{Claim 32.} Let $\mathfrak{A} \models \text{UNI}$. Then for any $a, b \in A$, $\bar{a} \subseteq A$ and $\bar{a} \subseteq A$

(i) For any $1 \leq i \leq 2k$, if $\mathfrak{A} \models \tau_u(a, b)$ then $\mathfrak{A} \models T_u ab$. In particular $\mathfrak{A} \models \text{TYPE}$.
(ii) For any non-transitive symbol $R$, if $\mathfrak{A} \models \rho_R(\bar{a})$ then $\mathfrak{A} \models R \bar{a}$.
(iii) If $\mathfrak{A} \models \varphi_0^\text{path}(\bar{a})$ then $\mathfrak{A} \models \varphi_0(\bar{a})$.
(iv) If $\mathfrak{A} \models \varphi$ then $\mathfrak{A} \models \varphi^\text{path}$.

\textbf{Proof.} (i) Since $\mathfrak{A} \models \text{UNI}$, the relation $\sim$ is the identity and $\varepsilon$ defines the identity.

Therefore, since $\mathfrak{A} \models \tau_u(a, b)$ we have that $\mathfrak{A} \models \exists x_0 \ldots x_{K+1} a = x_0 \wedge x_{K+1} = b \wedge T_u x_0 x_1 \wedge \bigwedge_{i=1}^K (T_u x_i x_{i+1} \vee x_i = x_{i+1})$. Since $\mathfrak{A}$ is transitively closed, $\mathfrak{A} \models T_u ab$.
(ii) Analogous to (i).
(iii) Use (i), (ii) and Claim 31(iii), (iv) ($\mathfrak{A} \models \text{TYPE}$ by (i) and $\text{UNI}$ clearly implies $\text{CON}$) to mimic the proof of Claim 31(v).
(iv) Follows from (iv) in such a way as Claim 31(vi) follows from Claim 31(v).

Next, let us now see how our formula transformation interacts with the shrinking operation.
\(\triangleright\) **Claim 33.** Let \((\tilde{\mathfrak{A}}, \mathfrak{A}, q)\) be a shrinking triple such that \(\tilde{\mathfrak{A}} \models \text{CON} \land \text{TYPE}\). Let \(a, b \in \tilde{\mathfrak{A}}\), let \(\tilde{a} \subseteq \tilde{\mathfrak{A}}\). Then

(i) For any \(1 \leq u \leq 2k\), if \(\mathfrak{A} \models \tau_u(q(a), q(b))\) then \(\tilde{\mathfrak{A}} \models \tau_u(a, b)\).

(ii) For any non-transitive symbol \(R\), if \(\mathfrak{A} \models \rho_R(q(\tilde{a}))\) then \(\tilde{\mathfrak{A}} \models \rho_R(\tilde{a})\).

(iii) For any \(1 \leq u \leq 2k\), \(\mathfrak{A} \models \tau_u(q(a), q(a))\) iff \(\tilde{\mathfrak{A}} \models \tau_u(a, a)\).

(iv) For any non-transitive symbol \(R\), \(\mathfrak{A} \models \rho_R(q(a), \ldots, q(a))\) iff \(\tilde{\mathfrak{A}} \models \rho_R(a, \ldots, a)\).

(v) If \(\mathfrak{A} \models \varphi^{\text{path}}_0(q(\tilde{a}))\) then \(\tilde{\mathfrak{A}} \models \varphi^{\text{path}}_0(\tilde{a})\).

(vi) In particular, if \(\mathfrak{A} \models \forall \tilde{x} \neg \varphi^{\text{path}}_0(\tilde{x})\), then \(\tilde{\mathfrak{A}} \models \forall \tilde{x} \neg \varphi^{\text{path}}_0(\tilde{x})\).

**Proof.** (i) Assume \(\mathfrak{A} \models \tau_u(q(a), q(b))\). Recall the intermediate model \(\mathfrak{A}^0\) used in the definition of shrinking. Then there exist elements \(a_0', a_1', \ldots, a_N' \in \mathfrak{A}^0\) such that

\[\mathfrak{A}^0 \models q(a) = a_0' \land \bigwedge_v T_v(q(a_i')(q(a_{i+1}'))) \land a_N' = q(b)\]

This path can be 'lifted' to \(\tilde{\mathfrak{A}}\), that is, there exist elements \(b_0, a_0, b_1, a_1, \ldots, a_{N-1}, b_N, a_N \in \tilde{\mathfrak{A}}\) such that \(b_0 = a, a_N = b\), and for all \(i\) we have \(q(a_i) = q(b_i) = a_i'\), and \(\tilde{\mathfrak{A}} \models T_v(a_i,b_{i+1})\). Assume that \(N\) is the smallest possible.

Observe, that we prove that \(N \leq K + 1\) and \(\tilde{\mathfrak{A}} \models \tau_u(a,b)\) (just substitute the existentially quantified variables in \(\tau_u\) with \(a_0, b_1, a_1, \ldots, a_{N-1}, b_N, b, b, \ldots, b\) respectively).

We claim that indeed \(N \leq K + 1\). Note first that for each \(1 \leq i \leq N - 1\) element \(a_i\) is an interpretation of a pseudoconstant and that \(b_i\) is (a different incarnation of) the same pseudoconstant. Otherwise \(a_i = b_i\), so we can remove them both from the sequence to obtain a shorter sequence satisfying the required properties (recall that \(\tilde{\mathfrak{A}}\) is transitively closed). Furthermore, all the \(a_i\), for \(1 \leq i \leq N - 1\), are different pseudoconstants -- if they were not, that is, for some \(i < i'\) we had \(a_i \sim a_i'\), we could cut it out \(b_i, a_i, \ldots, b_{i' - 1}, a_{i' - 1}\) and obtain a shorter sequence, with the required properties. As we have at most \(K\) constants it follows that \(N - 1 \leq K\).

(ii) Analogous to (i), yet simpler. It just suffices to "lift" \(q(\tilde{a})\) to some \(\tilde{a}^* \subseteq \tilde{\mathfrak{A}}\) such that \(\tilde{\mathfrak{A}} \models R(\tilde{a}^*)\).

(iii) \((\Rightarrow)\) Follows from (i). (\(\Leftarrow\)) By \(\tilde{\mathfrak{A}} \models \text{TYPE}\) we have that \(\tilde{\mathfrak{A}} \models T_0 aa\), so by the definition of shrinking \(\mathfrak{A} \models T_v q(a)q(a)\). Since \(\mathfrak{A} \models \text{UNI}\), it satisfies \text{CON} and by Claim \(\boxed{32}\) (i) it satisfies \text{TYPE}, by Claim \(\boxed{31}\) (i) \(\mathfrak{A} \models \tau_v(q(a), q(a))\).

(iv) \((\Rightarrow)\) Follows from (ii). (\(\Leftarrow\)) If there exists \(a' \sim a\) such that \(\tilde{\mathfrak{A}} \models Ra' \ldots a'\) then by the definition of shrinking \(\mathfrak{A} \models Rq(a') \ldots q(a')\). Since \(q(a) = q(a')\) we have that \(\mathfrak{A} \models \varepsilon(q(a), q(a'))\), so \(\mathfrak{A} \models \rho_R(q(a), q(a'))\).

(v) Follows from (i)--(iv) as Claim \(\boxed{31}\) (v) follows from Claim \(\boxed{31}\) (i)--(iv).

(vi) Recall that \(q\) is onto \(A\) and use (v).

\<

\(\triangleright\) **Lemma 34.** Let \((\tilde{\mathfrak{A}}, \mathfrak{A}, q)\) be a shrinking triple. Assume that \(\tilde{\mathfrak{A}} \models \varphi\text{path} \land \text{CON} \land \text{EXI} \land \text{TYPE}\). Then \(\mathfrak{A} \models \varphi\text{path} \land \text{CON} \land \text{EXI} \land \text{TYPE}\).

**Proof.** \(q\) preserves the \(1\)-types. Assume to the contrary that for some \(a \in \tilde{M}\) we have \(\text{atp}^{\mathfrak{A}}(a) \neq \text{atp}^{\tilde{\mathfrak{A}}}(q(a))\). Then, by the definition of shrinking and the fact that \(\tilde{\mathfrak{A}} \models \text{CON}\), we have that in fact the \(\subseteq\) inclusion holds and that the inequality follows from some transitive atom \(T_{u,xx}\) belonging to the latter and not to the former. But this means that, by Claim \(\boxed{32}\) (ii) we have that \(\tilde{\mathfrak{A}} \models \tau_u(a, a)\). Since \(\mathfrak{A} \models \text{TYPE}\) we have that \(\tilde{\mathfrak{A}} \models T_{u,aa}\). Contradiction.

\(\mathfrak{A}\) satisfies the \(\forall \exists\) conjuncts of \(\varphi\text{path}\). Follows from the following facts: \(q\) preserves the \(1\)-types and the set of positive atoms of arity greater than \(1\) can only be enlarged when pushed using \(q\). So it suffices to just push the witnesses through \(q\).
$\mathfrak{A} \models CON \land EXI \land UNI$. Straightforward, by the definition of shrinking.

$\mathfrak{A}$ satisfies the $\forall$-conjuncts of $\varphi$ and $\varphi^{path}$. Use Claim 33(vi).

$\mathfrak{A} \models TYPE$. Use Claim 32(i).

Let us now prove the main result of this subsection.

**Theorem 35.** The finite satisfiability problem for UNFO+$SO$ is 2-ExpTime-complete. If a UNFO+$SO$ formula has a finite model then it has a model of size triply exponential in its length.

**Proof.** Let us first show the second part of this theorem. Take a finitely satisfiable UNFO+$SO$ formula $\varphi^*$ and let $\varphi$ be its UNFO+$S$ version with constants simulated by pseudoconstants. Let $\mathfrak{A}$ be a finite model of $\varphi \land EXI \land UNI$. Thus, we also have $\mathfrak{A} \models CON \land TYPE$, and by part (iv) of Claim 32 it holds $\mathfrak{A} \models \varphi^{path}$. Note that the last formula has size at most quadratic in $|\varphi|$. We now take a triply exponentially bounded model $\mathfrak{B}$ of $\varphi^{path} \land EXI \land CON \land TYPE$, guaranteed by Thm. 25. Let $\mathfrak{B}$ be its shrinking. By Lemma 34 we have $\mathfrak{B} \models \varphi^{path} \land EXI \land UNI \land CON \land TYPE$. By part (i) of Claim 32 we have $\mathfrak{B} \models \varphi \land EXI \land UNI$ and thus $\mathfrak{B} \models \varphi^*$.

Regarding the complexity, as explained above, $\varphi^*$ has a finite model if $\varphi^{path} \land EXI \land CON \land TYPE$ has a finite model. As the latter is a UNFO+$S$ formula its satisfiability can be checked in 2-ExpTime by Thm 10.

**F.2 Constants and binary inclusions**

Here we show how to extend the proofs and techniques from the previous subsection to cover simultaneously constants and inclusions of binary relation. In this subsection we assume that, similarly to transitive symbols from $\sigma_{\text{dist}}$, also binary symbols from $\sigma_{\text{base}}$ come in pairs $B, B^{-1}$ and that the symbols from such pairs are always interpreted as inverses of each other. This can be done w.l.o.g., since all our constructions from section F.1 respect this property.

The (finite) satisfiability problem for the unary negation fragment with constants and inclusions of binary relations, UNFO+$SOH$, is defined as follows. Given a UNFO+$SOH$ formula $\varphi$ and a set $\mathcal{H}$ of inclusions of the form $B \subseteq B'$, for binary symbols $B, B'$, check if there exists a (finite) model of $\varphi$ in which for every $B \subseteq B' \in \mathcal{H}$ the interpretation of $B$ is contained in the interpretation of $B'$.

As in the previous subsection we simulate constants by pseudoconstants and search for finite models of a formula $\varphi \land UNI \land EXI \land \mathcal{H}$, with normal form $\varphi$, over a purely relational signature $\sigma = \sigma_{\text{base}} \cup \sigma_{\text{dist}} \cup \sigma_{\text{const}}$.

We first remark that all our constructions from Sections E and F.1 without literally any changes, respect inclusions of the form $T \subseteq T'$, $B \subseteq B'$ and $B \subseteq T$ for any $T, T' \in \sigma_{\text{dist}}$ and $B, B' \in \sigma_{\text{base}}$. The only problematic inclusions are those of the form $T \subseteq B$ for $B \in \sigma_{\text{base}}$ and $T \in \sigma_{\text{dist}}$. To deal with them we will introduce an operation of taking the pseudotransitive closure.

For a given set of inclusions $\mathcal{H}$ let $\mathcal{H}^+$ denote the smallest set such that (i) $\mathcal{H} \subseteq \mathcal{H}^+$, (ii) if $B_1 \subseteq B_2 \in \mathcal{H}^+$ then $B_1^{-1} \subseteq B_2^{-1} \in \mathcal{H}^+$, (iii) if $B_1 \subseteq B_2 \in \mathcal{H}^+$ and $B_2 \subseteq B_3 \in \mathcal{H}^+$ then $B_1 \subseteq B_3 \in \mathcal{H}^+$. For any structure $\mathfrak{A}$ we have that $\mathfrak{A} \models \mathcal{H}$ iff $\mathfrak{A} \models \mathcal{H}^+$. So, w.l.o.g., from now we assume that $\mathcal{H}$ itself satisfies (i)–(iii). Denote by $\mathcal{H}_0$ the subset of "safe" inclusions of $\mathcal{H}$, $\mathcal{H}_0 := \{B_1 \subseteq B_2 \in \mathcal{H} : B_1 \notin \sigma_{\text{dist}} \lor B_2 \notin \sigma_{\text{base}}\}$. If $T \subseteq B \in \mathcal{H}$, for $T \in \sigma_{\text{dist}}$ and $B \in \sigma_{\text{base}}$ then $B$ is called pseudo-transitive. Pseudo-transitive relations must be treated in a special way. Let $\mathfrak{A}$ be a structure. Then we define its pseudotransitive closure as the structure $\mathfrak{A}'$ as follows: $A = A'$, for all $a, b \in A$ and $B \in \sigma_{\text{base}}$, $\mathfrak{A} \models Bab$ iff $\mathfrak{A}' \models Bab \lor \bigvee_{T \subseteq B \in \mathcal{H}, T \in \sigma_{\text{dist}}} Tab$. 
for all the other relations we copy their interpretations from $\mathfrak{A}$ to $\mathfrak{A}'$. Observe that if $\mathfrak{A} \models H_0$ then $\mathfrak{A}' \models H$, if $\mathfrak{A} \models H$ then $\mathfrak{A} = \mathfrak{A}'$, and that the identity map $\iota : A \to A'$ is a homomorphism.

To deal simultaneously with constants and inclusions, we plan to apply both the shrinking operation and the pseudotransitive closure. More precisely, starting from a model respecting $H_0$ we first apply to it the shrinking operation and then, to the result, the pseudotransitive closure, obtaining a model satisfying both $H$ and UNI. As in the previous subsection, to ensure that the above transformations respect the given formula $\varphi$ we will need to perform some syntactic manipulations, and formulate a counterpart of Lemma 34 taking into account both types of operations. To make this work for the pseudotransitive closure, slightly different manipulations and slightly stronger assumptions on the initial model will be needed.

In particular, for technical reasons, we need a simple condition on the realized 1-types, namely, that they respect the inclusions from $H$. Formally, this is captured by a UNFO formula $RES := \forall x \bigwedge_{B_1 \subseteq B_2 \in H} (B_1 xx \rightarrow B_2 xx)$. Observe that for any model $\mathfrak{A}$ such that $\mathfrak{A} \models H$, $\mathfrak{A} \models RES$.

Given $\varphi$ in normal form with universal conjunct $\forall \bar{x} \neg \varphi_0(\bar{x})$ we now describe its modifications corresponding to path modifications from Section F.1. The modifications are performed on formula $\varphi_0$ for the atoms $T_{wry}$ and $R\bar{e}$ with $R$ of arity other than 2, replace them, as before, with the formulas $\tau_u(x, y)$ and $\rho_R(\bar{e})$, respectively. In the case of atoms $R\bar{e}y$ (with $R \in \sigma_{2\text{var}}$), replace them with $\omega_R(x, y) := p_R(x, y) \lor \bigvee_{T_u \subseteq R \in H} \tau_u(x, y)$. Let $\varphi_0^{\text{path-inc}}$ be the result of this transformation, and let $\varphi^{\text{path-inc}}$ be the result of substituting the conjunct $\forall \bar{x} \neg \varphi_0(\bar{x})$ of $\varphi$ with $\forall \bar{x} \neg \varphi_0^{\text{path-inc}}$. Note that $\varphi^{\text{path-inc}}$ is (equivalent to) an UNFO+$\mathcal{S}$ formula.

From now on we can proceed analogously as we did for the path modification. Let us now collect some basic properties of this syntactic transformation and see how it interplays with the operations of shrinking and taking the pseudotransitive closures.

$\triangleright$ Claim 36. Let $\mathfrak{A}$ be an arbitrary structure, $a, b \in A$ and $R \in \sigma_{\text{base}}$ be a relational symbol of arity 2. Then

(i) If $\mathfrak{A} \models Rab$ then $\mathfrak{A} \models \omega_R(a, b)$

(ii) If $\mathfrak{A} \models \text{CON} \land \text{TYPE} \land RES$ then $\mathfrak{A} \models Raa$ if $\mathfrak{A} \models \omega_R(a, a)$

(iii) If $\mathfrak{A} \models \text{CON} \land \text{TYPE} \land RES$ then if $\mathfrak{A} \models \varphi^{\text{path-inc}}$ then $\mathfrak{A} \models \varphi$

Proof. (i) Follows from Claim 31(ii) and the definition of $\omega_R$.

(ii) (⇒) Follows from (i). (⇐) By the definition of $\omega_R$ and Claim 31(iii), (iv) we have that $\mathfrak{A} \models Rab$ or $\mathfrak{A} \models Tab$ for some $T \subseteq R \in H$. Using the fact that $\mathfrak{A} \models RES$ the thesis follows.

(iii) Use Claim 31(i)–(iv) and (i), (ii) as usual.

$\triangleright$ Claim 37. Let $\mathfrak{A} \models \text{UNI} \land H$. Then for any $a, b \in A$

(i) If $\mathfrak{A} \models \omega_R(a, b)$ then $\mathfrak{A} \models Rab$

(ii) If $\mathfrak{A} \models \varphi$ then $\mathfrak{A} \models \varphi^{\text{path-inc}}$.

Proof. (i) By the definition of $\omega_R$ and Claim 31(i), (ii) we have that $\mathfrak{A} \models Rab$ or $\mathfrak{A} \models Tab$ for some $T \subseteq R \in H$. Since $\mathfrak{A} \models H$, the thesis follows.

(ii) Use Claims 32(i), (ii), 31(iii), (iv) ($\mathfrak{A} \models \text{UNI}$ so it satisfies the required assumptions), 36(ii) ($\mathfrak{A} \models H$ implies $\mathfrak{A} \models RES$) and (i) as usual.
Furthermore, if \( \exists \mathbf{A} \) with the role of \( \mathbf{A} \) exists some \( \varepsilon \in \mathbf{D} \).

Claim 38. Let \( \mathbf{A}, \mathbf{A}, \mathbf{q} \) be a shrinking triple such that \( \mathbf{A} \models \mathbf{CON} \land \mathbf{TYPE} \land \mathbf{RES} \). Let \( a, b \in \mathbf{A} \) and \( R \) be a binary non-transitive symbol. Then

(i) If \( \mathbf{A} \models R(q(a), q(b)) \) then \( \mathbf{A} \models R(a, b) \)

(ii) If \( \mathbf{A} \models R^2(a, a) \) iff \( \mathbf{A} \models R(q(a), q(a)) \).

(iii) If \( \mathbf{M} \models \forall \bar{x} \neg \nu^\text{path-inc}_0(\bar{x}) \), then \( \mathbf{M} \models \forall \bar{x} \neg \nu^\text{path-inc}_0(\bar{x}) \).

<Proof.>

(i) Use the definition of \( \nu_R \) and Claims 33(i), (ii).

(ii) Use the definition of \( \nu_R \) and Claims 33(i), (ii).

(iii) Follows from (i), (ii) and Claim 33(i)–(iv) as usual.

Claim 39. Let \( \mathbf{A}' \) be the pseudotransitive closure of \( \mathbf{A}, \mathbf{A}, \mathbf{H} \) and \( a, b \in \mathbf{A}(= \mathbf{A}') \subseteq \mathbf{A} \). Then

(i) For any \( 1 \leq u \leq 2k \) we have \( \mathbf{A}' \models \varepsilon_u(a, b) \) iff \( \mathbf{A} \models \varepsilon_u(a, b) \) and \( \mathbf{A}' \models \rho_R(\bar{a}) \) iff \( \mathbf{A} \models \rho_R(\bar{a}) \).

(ii) For any non-transitive symbol \( R \) of arity \( 2 \) we have \( \mathbf{A}' \models \rho_R(a, b) \) iff \( \mathbf{A} \models \rho_R(a, b) \).

(iii) For any \( 1 \leq u \leq 2k \) we have \( \mathbf{A}' \models \tau_u(a, a) \) iff \( \mathbf{A} \models \tau_u(a, a) \). For any non-transitive symbol \( R \) of arity other than \( 2 \) it holds that \( \mathbf{A}' \models \rho_R(a, \ldots, a) \) iff \( \mathbf{A} \models \rho_R(a, \ldots, a) \).

(iv) For any binary non-transitive symbol \( R \) we have \( \mathbf{A}' \models \rho_R(a, b) \) iff \( \mathbf{A} \models \rho_R(a, b) \).

(v) If \( \mathbf{A} \models \nu^\text{path-inc}_0 \) then \( \mathbf{A}' \models \nu^\text{path-inc}_0 \).

<Proof.>

(i) Follows from the fact that the process of the pseudotransitive closure does not change neither the transitive relations nor the non-transitive relations of arity other than \( 2 \) nor the equalities (and these are the types of atoms that \( \rho_R \) and \( \tau_u \) consist of).

(ii) \( \vDash \) If \( \mathbf{A}' \models \neg \rho_R(a, b) \) then use (i). Otherwise there exist \( a', b' \) such that \( \mathbf{A}' \models \varepsilon(a, a') \land \varepsilon(b, b') \land R(a', b') \), so \( \mathbf{A} \models R(a, b) \) or \( \mathbf{A} \models T_u a'b' \) for some transitive \( T_u \) such that \( T \subseteq R \subseteq \mathcal{H} \). Thus, since the interpretation of \( \varepsilon \) is not changed by the application of the pseudotransitive closure, by Claim 33(i), (ii) we have that \( \mathbf{A} \models \rho_R(a, b) \) or \( \mathbf{A} \models \tau_u(a, a, b) \) and therefore \( \mathbf{A} \models \rho_R(a, b) \). \( \vDash \) If \( \mathbf{A} \models \tau_u(a, a, b) \) for some \( T_u \subseteq R \subseteq \mathcal{H} \) then by (i) we have \( \mathbf{A}' \models \tau_u(a, b) \). If \( \mathbf{A} \models \rho_R(a, b) \) then, since the interpretation of \( R \) in \( \mathbf{A} \) is contained in the interpretation of \( R \) in \( \mathbf{A}' \) and \( \varepsilon \) is interpreted identically in \( \mathbf{A} \) and \( \mathbf{A}' \) we have that \( \mathbf{A}' \models \rho_R(a, b) \).

(iii) Follows from (i).

(iv) Follows from (ii).

(v) Follows from (i)–(iv) as usual.

Now we are ready to state the counterparts for Lemma 34.

> Lemma 40. Let \( \mathbf{A}' \) be the pseudotransitive closure of \( \mathbf{A} \). Assume that \( \mathbf{A} \models \phi^\text{path-inc} \land \mathbf{EXI} \land \mathbf{CON} \land \mathbf{TYPE} \land \mathbf{RES} \land \mathbf{H}_0 \). Then \( \mathbf{A}' \models \phi^\text{path-inc} \land \mathbf{EXI} \land \mathbf{CON} \land \mathbf{TYPE} \land \mathbf{RES} \land \mathbf{H} \). Furthermore, if \( \mathbf{A} \models \mathbf{UNI} \) then \( \mathbf{A}' \models \mathbf{UNI} \).

<Proof.>

Let \( \iota : A \to A' \) be the inclusion map.

\( \iota \) preserves the 1-types. The only possibility for non-preservation of the 1-types is when there exists some \( a \in A \) and non-transitive \( R \) such that \( Rxx \in atp^\mathbf{A'}(a) \setminus atp^\mathbf{A}(a) \), appearing due to the fact that \( \mathbf{A} \models Taa \) for some transitive \( T \), and \( T \subseteq R \subseteq \mathcal{H} \). But then \( Txx \in atp^\mathbf{A}(a) \) and since \( \mathbf{A} \models \mathbf{RES} \) we get that \( Rxx \in atp^\mathbf{A}(a) \). Contradiction.

\( \mathbf{A'} \) satisfies the \( \forall \exists \) conjunct of \( \phi^\text{path-inc} \), \( \mathbf{CON}, \mathbf{EXI} \). Exactly as in the proof of Lemma 34 with the role of \( q \) played now by \( \iota \).

\( \mathbf{A}' \models \mathbf{RES} \). \( \mathbf{RES} \) uses only unary atoms and \( \iota \) respects 1-types.
\( \mathfrak{A} \) satisfies the \( \forall \) conjunct of \( \varphi^{\text{path-inc}} \). Use Claim 39(v).

\( \mathfrak{A} \models \text{TYPE}. \) The formula TYPE uses only some transitive symbols, some symbols of arity 1 and equalities, whose interpretation is not changed after the application of the pseudotransitive closure.

\[ \textbf{Lemma 41.} \] Let \((\mathfrak{A}, \mathfrak{A}', q)\) be a shrinking triple. Assume that \( \mathfrak{A} \models \varphi^{\text{path-inc}} \land \text{EXI} \land \text{CON} \land \text{TYPE} \land \text{RES} \land \mathcal{H}_0 \). Then \( \mathfrak{A}' \models \varphi^{\text{path-inc}} \land \text{EXI} \land \text{UNI} \land \text{CON} \land \text{TYPE} \land \text{RES} \land \mathcal{H}_0 \).

\[ \textbf{Proof.} \] \( \mathfrak{A} \models \mathcal{H}_0 \). By the definition of the shrinking operation, \( \mathfrak{A}^0 \models \mathcal{H}_0 \). Clearly, taking the transitive closure to obtain \( \mathfrak{A} \) does not spoil this condition.

\( q \) preserves the 1-types. Exactly as in the proof of Lemma 54.

\( \mathfrak{A} \) satisfies the \( \forall \exists \) conjuncts of \( \varphi^{\text{path-inc}} \), and the formulas \( \text{CON, EXI, TYPE, UNI} \). Exactly as in the proof of Lemma 54.

\( \mathfrak{A} \models \text{RES}. \) RES uses only unary atoms and \( q \) respects 1-types.

\( \mathfrak{A} \) satisfies the \( \forall \) conjunct of \( \varphi^{\text{path-inc}} \). Use Claim 38(iii).

Now we are ready to put the pieces together and prove the main result for the UNFO+SOH. The first part of the following theorem has already been stated as Thm. 11.

\[ \textbf{Theorem 42.} \] The finite satisfiability problem for UNFO+SOH is 2-ExpTime-complete. If a UNFO+SOH formula has a finite model then it has a model of size triply exponential in its length.

\[ \textbf{Proof.} \] Let us first show the second part of this theorem. Take a finitely satisfiable UNFO+SOH formula \( \varphi^* \) and let \( \varphi \) be its UNFO+S version with constants simulated by pseudoconstants. Let \( \mathcal{H} \) be its inclusions. Let \( \mathfrak{A} \) be a finite model of \( \varphi \land \text{EXI} \land \text{UNI} \land \mathcal{H} \). Therefore we also have \( \mathfrak{A} \models \text{CON} \land \text{TYPE} \land \text{RES} \land \mathcal{H}_0 \), and by part (ii) of Claim 37 it holds \( \mathfrak{A} \models \varphi^{\text{path-inc}} \). Note that the last formula has size at most quadratic in \(|\varphi|\). We now take a triply exponentially bounded model \( \mathfrak{B} \) of \( \varphi^{\text{path-inc}} \land \text{EXI} \land \text{CON} \land \text{TYPE} \land \text{RES} \land \mathcal{H}_0 \), guaranteed by Thm. 25 (recall our previous observation, that the inclusions from \( \mathcal{H}_0 \) are satisfied by all the transformations of models from Section 3). Let \( \mathfrak{B} \) be its shrinking. By Lemma 41 we have \( \mathfrak{B} \models \varphi^{\text{path-inc}} \land \text{EXI} \land \text{UNI} \land \text{CON} \land \text{TYPE} \land \text{RES} \land \mathcal{H}_0 \). Let \( \mathfrak{B}' \) be the pseudotransitive closure of \( \mathfrak{B} \). By Lemma 10 we have that \( \mathfrak{B}' \models \varphi^{\text{path-inc}} \land \text{EXI} \land \text{UNI} \land \text{CON} \land \text{TYPE} \land \text{RES} \land \mathcal{H} \). By part (iii) of Claim 36 we have \( \mathfrak{B} \models \varphi \land \text{EXI} \land \text{UNI} \land \mathcal{H} \) and thus \( \mathfrak{B} \models \varphi^* \).

Regarding the complexity, as explained above \( \varphi^* \) has a finite model iff \( \varphi^{\text{path-inc}} \land \text{EXI} \land \text{CON} \land \text{TYPE} \land \text{RES} \land \mathcal{H}_0 \) has a finite model. The latter is a conjunction of a UNFO+S formula and 'safe' inclusions and its satisfiability can be checked in 2-ExpTime by an adaptation of the algorithm from the proof of Thm 10 forcing it to search for models satisfying \( \mathcal{H}_0 \) just by ensuring that the structures built on the downward families respect \( \mathcal{H}_0 \).

As mentioned in the Introduction, UNFO+SOH captures several interesting description logics. This implies that we can solve FOMQA problem for them. In particular, we have the following corollary, which, up to our knowledge is the first decidability result for FOMQA in the case of a description logic with both transitive roles and role hierarchies.

\[ \textbf{Corollary 43 (restating of Cor. 12).} \] Finite ontology mediated query answering, FOMQA, for the description logic \( \text{SHOIQ} \) is decidable and 2-ExpTime-complete.
some related logics are considered, e.g., in [11]. For more about FOMQA for description logics with transitivity see [12]. For more about OMQA for description logics see, e.g., references in [12].

\section{Towards guarded negation fragment with transitivity}

\subsection{1-dimensional guarded negation fragment with transitivity}

Guarded negation fragment, GNFO [5], is a common decidable extension of UNFO and GF in which negated subformula $\psi$ must be used in conjunction with an atom, called a guard, containing all the free variables of $\psi$. Formally, it is defined by the following grammar:

$$\varphi = B\bar{x} \mid x = y \mid \varphi \land \varphi \mid \varphi \lor \varphi \mid \exists x.\varphi \mid \gamma(\bar{x}, \bar{y}) \land \neg \varphi(\bar{y}),$$

where $\gamma$ is an atomic formula, called a guard. Equality statements of the form $x = x$ can be used as guards, so UNFO can be seen as a fragment of GNFO.

The (finite) satisfiability problem for GNFO with transitive relations is undecidable, since already the two-variable guarded fragment with transitive relations, GF2+TR, is undecidable, [18, 17]. Recall that the decidability of the general satisfiability problem is regained when transitive symbols are admissible only on non-guard positions [1]. We call this decidable variant the base-guarded negation fragment with transitivity, BGNFO, and recall that it embeds UNFO+S. We do not solve its finite satisfiability problem here, but, analogously to the extension with equivalence relations, UNFO+EQ [7], we are able to lift our results to its one-dimensional restriction, BGNFO1+S. We say that a first-order formula is one-dimensional if its every maximal block of quantifiers leaves at most one variable free. E.g., $\neg\exists z Bxyz$ is one-dimensional, and $\neg\exists z Bxyz$ is not.

\begin{theorem}[rastating of Thm 13] The finite satisfiability problem for BGNFO1+S is 2-ExpTime-complete.
\end{theorem}

The required modifications in our constructions are analogous to those described in [7], in the case of equivalences. We put them here for the reader’s convenience.

Using a natural adaptation of the standard Scott translation [25] we can transform any sentence belonging to BGNFO1+S into a normal form sentence $\varphi$ of the shape as in [1], where the $\varphi_i$ are quantifier-free GNFO formulas. Assume that some finite structure $\mathfrak{A}$ is a model of $\varphi$. First, we need a slightly stronger version of condition (a1) in Lemma 1—each of the considered homomorphisms should additionally be an isomorphism when restricted to a guarded substructure. We need to extend the notion of a declaration so that it treats subformulas of the form $\gamma(\bar{x}, \bar{y}) \land \neg \varphi'(\bar{y})$ like non-transitive atomic formulas. This allows us to perform surgery making the transitive paths bounded and then to construct a regular tree-like model $\mathfrak{A}' \models \varphi$ as it is done in the proofs of Lemma 22 and Lemma 24 respectively. The key facts are that Lemma 21 holds (with the new declarations) and that $\varphi_0$ is equivalent to a disjunction of some formulas generated by declarations. Finally we apply, without any changes, the construction from the proof of Lemma 26 to $\mathfrak{A}'$ and $\varphi$ obtaining eventually a finite structure $\mathfrak{A}''$. Note that during the step of providing witnesses we build isomorphic copies of partial witness structures, which means that we preserve not only positive atoms but also their negations. Thus the elements of $A''$ have all witness structures required by $\varphi$. Consider now the conjunct $\forall x_1, \ldots, x_t.\neg \varphi_0(\bar{x})$, and take arbitrary elements $a_1, \ldots, a_t \in A''$. From Lemma 26 we know that there is a homomorphism $h : \mathfrak{A''} \{a_1, \ldots, a_t\} \to \mathfrak{A}'$ preserving 1-types. If $\gamma(\bar{x}, \bar{y}) \land \neg \varphi'(\bar{y})$ is a subformula of $\varphi_0$ with $\gamma$ a $\sigma_{base}$-guard and $\mathfrak{A}'' \models \gamma(\bar{b}, \bar{c}) \land \neg \varphi'(\bar{c})$
for some $\overline{b}, \overline{c} \subseteq \overline{a}$ then, by our construction, all elements of $\overline{b} \cup \overline{c}$ are members of the $\varphi$-witness structure for some element. As mentioned above such witness structures are isomorphic copies of substructures from $\mathfrak{A}$ and $\mathfrak{b}$ works on them as an isomorphism, and thus $\mathfrak{b}$ preserves on $\overline{c}$ not only 1-types and positive atoms but also negations of atoms in witnesses structures. Since $\mathfrak{A}' \models \neg \varphi_0(h(a_1), \ldots, h(a_t))$ this means that $\mathfrak{A}'' \models \neg \varphi_0(a_1, \ldots, a_t)$.

The algorithm for checking finite satisfiability presented in the proof of Thm. 10 also works without any changes and, moreover, its correctness proof does not need any modifications. It is the case since a key role is played here by Lemma 21 that still holds with the new version of declarations.

### G.2 Undecidability with inclusions of binary relations

Note that $\text{BGNFO}_{1}+S$ can express inclusions $B \subseteq T$ and $B \subseteq B'$, and our constructions respect the inclusions of the form $T \subseteq T'$ for $B, B' \in \sigma_{\text{base}}$ and $T, T' \in \sigma_{\text{dist}}$. However, it turns out that if we extend it with inclusions of the form $T \subseteq B$ then the (finite) satisfiability problem becomes undecidable. This can be easily shown by a reduction from the already-mentioned (finite) satisfiability problem for $\text{GF}^2+TR$. Indeed, all the negations in a $\text{GF}^2+TR$ formula $\varphi$ can be guarded by the guards of quantifiers. If $\varphi$ uses a transitive guard $T$ then we can add an inclusion $T \subseteq B_T$, for a fresh $B_T \in \sigma_{\text{base}}$ and then use $B_T$ to guard negations. More precisely, subformulas of $\varphi$ of the form $\exists y (Txy \land \psi(x, y))$ are replaced by $\exists y (Txy \land \psi^*(x, y))$, where $\psi^*$ is obtained by replacing negated subformulas $\neg \varsigma(x, y)$ (not in the scope of a deeper quantifier) of $\psi$ by, properly base-guarded, $B_T xy \land \neg \varsigma(x, y)$. (In subformulas of the original formula of the form $\exists y (Bxy \land \psi(x, y))$, for non-transitive $B$, we just add the guard $B_Txy$ to all binary negations in $\psi$, not in the scope of a deeper quantifier.) Note that since $\text{GF}^2+TR$ uses only two variables all its formulas are one-dimensional, which is not changed by the described reduction. We conclude:

**Theorem 45.** The (finite) satisfiability problem for $\text{BGNFO}_{1}+S$ with inclusions of binary relations is undecidable.