TWO-VARIABLE LOGIC WITH TWO ORDER RELATIONS

THOMAS SCHWENTICK [a] AND THOMAS ZEUME [b]

[a,b] TU Dortmund University
    e-mail address: {thomas.schwentick, thomas.zeume}@cs.uni-dortmund.de

ABSTRACT. It is shown that the finite satisfiability problem for two-variable logic over structures with one total preorder relation, its induced successor relation, one linear order relation and some further unary relations is EXPSPACE-complete. Actually, EXPSPACE-completeness already holds for structures that do not include the induced successor relation. As a special case, the EXPSPACE upper bound applies to two-variable logic over structures with two linear orders. A further consequence is that satisfiability of two-variable logic over data words with a linear order on positions and a linear order and successor relation on the data is decidable in EXPSPACE.

As a complementing result, it is shown that over structures with two total preorder relations as well as over structures with one total preorder and two linear order relations, the finite satisfiability problem for two-variable logic is undecidable.

1. INTRODUCTION

First-order logic restricted to two-variables (*two-variable logic* or FO^2 in the following) is generally known to be reasonably expressive for many purposes. In contrast to full first-order logic, its satisfiability and its finite satisfiability problem are decidable [Mor75], in fact they are NEXPTIME-complete [GKV97].

However, if one is interested in (finite or general) satisfiability of FO^2 over structures with a particular property P these general results can only be applied if P is expressible in FO^2. Unfortunately, there are some simple properties like transitivity of a binary relation that cannot be expressed in FO^2. In particular, in FO^2 it can neither be expressed that a given binary relation is a linear order nor that it is an equivalence relation. Thus, the results from [Mor75, GKV97] do not help for satisfiability of FO^2 over (finite or general) structures with linear orders or equivalence relations,

In [Ott01] it was shown that it can be decided in NEXPTIME whether a given FO^2 sentence has a model (or whether it has a finite model) in which a particular relation symbol

is interpreted by a linear order. On the other hand, in the presence of eight binary symbols that have to be interpreted as linear orders it is undecidable. Note that in these results the formulas might use further relation symbols of arbitrary arity for which the possible interpretations are unrestricted.

The problem of deciding finite satisfiability[1] was shown to be NEXPTIME-complete over structures with one equivalence relation and undecidable over structures with three equivalence relations in [KO05]. In [KT09] it was shown that over structures with two equivalence relations the problem is decidable in triply exponential nondeterministic time.

In this article we study two-variable logic over structures with linear orders and total preorders. A total preorder $\precsim$ is basically an equivalence relation $\sim$ whose equivalence classes are ordered by $\prec$. Total preorders can therefore encode equivalence relations as well as linear orders and in this sense they generalize both types of relations. It should be stressed that in our results, structures may have an arbitrary number of additional unary relations but *no* further non-unary relations.

Our motivation stems from the context of so-called data words. A *data word* is a word, that is, a finite sequence of symbols from a finite alphabet, but besides a symbol, every position also carries a value from a possibly infinite domain. An interest in data words and data trees arises from applications in database theory, where XML documents can be modeled by data trees in which the symbols correspond to the tags and the data values to text or attribute values. On the other hand, (infinite) data words can also be considered as traces of computations in a distributed environment, where symbols correspond to states of processes and data values encode process numbers. Recently many logics and automata models have been considered for data words and data trees (see [Seg06] for a gentle introduction).

First-order logic on data words (with a linear order on positions and equality on data values) is undecidable, even for formulas with three variables [BMS+06].

On the other hand, finite satisfiability of two-variable logic on data words is decidable. More precisely, it was shown in [BMS+06] that it is decidable even in the setting where data words are equipped with a linear order and its corresponding successor relation on positions and equality on data values. However, the complexity is unknown but basically equivalent to the open complexity of Petri net reachability. It was further shown in [BMS+06] that the problem is NEXPTIME-complete without the successor relation on the positions and that it becomes undecidable if the data values are equipped with a linear order (in the presence of linear order and successor relation on positions).

Data words are closely related with finite structures with order relations. More precisely, data words can be represented by

- some unary relations to encode symbols at positions,
- a linear order (on the positions), possibly its induced successor relation, and
- an equivalence relation corresponding to data equality between positions.

In this context, an additional linear order on data values can be represented by a total preorder $\precsim$. All positions with a particular value constitute an equivalence class and these classes are ordered by the linear order on the values. It is exactly this setting which triggered our study of structures with a linear order, a total preorder, and some unary relations.

---

[1]As this article only deals with finite structures we henceforth only mention results on finite satisfiability.

Results. We show that finite satisfiability of two-variable logic over structures with a linear order, a total preorder, and unary relations is EXPSPACE-complete. Consequently, finite satisfiability of FO^2 over data words with a linear order (but no successor relation) on the positions and a linear order on the data values can be decided in exponential space as well. As it can be expressed in FO^2 that a total preorder is a linear order, the corresponding problem with two linear orders (and no total preorder) is also solvable in EXPSPACE. Thus, the gap between one and eight linear orders that was left in the work of Otto [Ott01] is narrowed.

The upper bound even holds when the preorder is accompanied by a partial successor relation (the precise definition of this notion will be given in Section 2). As a corollary, satisfiability of FO^2 on data words is in EXPSPACE even if the linear order on the data values is accompanied with a corresponding partial successor relation.

The upper bounds are obtained by a reduction to finite satisfiability of *semi-positive* FO^2 sentences over sets of labeled points in the plane, where points can be compared by their relative position with respect to the directions $\nwarrow, \nearrow, \swarrow, \searrow, \leftarrow, \rightarrow$. Furthermore two points can be tested for being on successive horizontal lines. Semi-positive formulas are in negation normal form and do not contain negated "direction atoms". For a precise definition we refer to Section 3. Finite satisfiability of *semi-positive* FO^2 over such point sets can in turn be reduced in exponential time to a constraint problem for labeled points in the plane with PSPACE-complexity. The EXPSPACE lower bound is by a reduction from exponential width corridor tiling.

Finally, we show by reductions from the Post Correspondence Problem (PCP) problem that finite satisfiability of FO^2 over structures[2] with two total preorders and over structures with two linear orders and a total preorder is undecidable.

Organization. After some basic definitions in Section 2, we prove the EXPSPACE upper bound in Section 3 and all lower bounds in Section 4. We conclude with Section 5 where we discuss research directions and related work on compass and interval logics.

Related work. As mentioned before, also other logics for data words besides FO^2 have been studied. As an example we mention the "freeze"-extension of LTL studied, e.g. in [DL09] and [FS09]. The latter paper is more closely related to our work as it considers a restriction of LTL without the X-operator. Amaldev Manuel has recently proved decidability and undecidability results for FO^2-logic over structures with orders [Man10]. However, in his work structures have at least two successor relations but no linear orders, hence neither results nor techniques translate from his work to ours nor vice versa.

---

[2]Additional unary relations are again allowed.

2. Preliminaries

In this article, we only consider *finite* structures. We are interested in three kinds of finite structures: ordered structures, sets of labeled points in the plane and data words.

In the following, $\mathbb{N}$ denotes the set of natural numbers and $\mathbb{Q}$ the set of rationals.

Ordered Structures. We first fix our notation concerning order relations. A *total preorder* $\precsim$ is a transitive, total relation, that is, $u \precsim v$ and $v \precsim w$ implies $u \precsim w$ and for every two elements u, v of a structure $u \precsim v$ or $v \precsim u$ holds. In particular, every total preorder is reflexive, that is $u \precsim u$ holds for every u. A *linear order* $\leq$ is a antisymmetric total preorder, that is, if $u \leq v$ and $v \leq u$ then $u = v$. Thus, the essential difference between a total preorder and a linear order is that the former allows that for two distinct elements u and v both $u \precsim v$ and $v \precsim u$ hold. We call two such elements *equivalent with respect to* $\precsim$ and write $u \sim v$. We write $u \prec v$ if $u \precsim v$ and $u \not\sim v$.

A total preorder can be viewed as an equivalence relation $\sim$ whose equivalence classes are linearly ordered by $\prec$. Clearly, every linear order is a total preorder with equivalence classes of size one.

We define the *induced successor relation* $S_{\precsim}$ of a total preorder $\precsim$ as follows. For two elements u, v, $S_{\precsim}(u, v)$ if $u \prec v$, and there is no element w such that $u \prec w \prec v$. A *partial successor relation* S is a sub-relation of $S_{\precsim}$ such that $u \sim u'$, $v \sim v'$ and $S(u, v)$ imply $S(u', v')$. Thus, a partial successor relation is a sub-relation of $S_{\precsim}$ that is derived from a sub-relation of the successor relation on the equivalence classes of $\precsim$. We say that $S_{\precsim}$ is *complete* to distinguish it from (truly) partial successor relations.

We use binary relation symbols $<, <_1, <_2, \ldots$ that are always interpreted as linear orders, binary relation symbols $\precsim, \precsim_1, \precsim_2, \ldots$ that are interpreted as total preorders, and binary relation symbols $S, S_1, S_2, \ldots$ that are interpreted as (partial or complete) successor relations. We note that $\sim$ and $\prec$ can be expressed in two-variable logic, given $\preceq$.

In this article, an *ordered structure* is a finite structure with non-empty universe and some linear orders, some total preorders, some successor relations and some unary relations. We always allow an unlimited number of unary relations and specify the numbers of allowed linear orders and total preorders explicitly.

We denote classes of ordered structures by the notation $\mathrm{FinOrd}(\mathcal{O})$ where $\mathcal{O}$ indicates the orders and successor relations following the above conventions. Here, corresponding relations are grouped together by square brackets. For example, by $\mathrm{FinOrd}(\leq_1, [\precsim_2, S_2])$ we denote the set of finite structures with one linear order and one total preorder together with a corresponding partial successor relation.

Sets of Labeled Points. As mentioned before, we also consider finite sets of labeled points. Let $\mathrm{Prop} = \{e_1, \ldots, e_k\}$ be a set of propositions. A Prop-*labeled point* p is a point in $\mathbb{N}^2$ in which propositions $e_1, \ldots, e_k$ may or may not hold. We refer to the x-coordinate and the y-coordinate of a point p by $p.x$ and $p.y$, respectively. We simply say *point* if Prop is understood from the context. We do not allow different points $p \neq q$ at the same position, that is, if $p.x = q.x$ and $p.y = q.y$ then $p = q$. We say that a set Prop of labeled points is *contiguous*, if the y-coordinates of Prop constitute an interval[3] in $\mathbb{N}$.

---

[3]It is not required that every number occurs only once as the y-coordinate of a point.

For a finite set P of labeled points, we write $p \nwarrow q$ if $p.x > q.x$ and $p.y < q.y$, that is if q is in the northwest of p. Likewise for $\leftarrow, \swarrow, \downarrow, \searrow, \rightarrow, \nearrow, \uparrow$. We write $I_y(p,q)$ if $p.y + 1 = q.y$ and $I_x(p,q)$ if $p.x + 1 = q.x$. Let $\mathcal{D} = \{\nearrow, \uparrow, \nwarrow, \leftarrow, \swarrow, \downarrow, \searrow, \rightarrow\}$ denote the set of *directions*. We denote $\mathcal{D}$ without $\downarrow$ and $\uparrow$ by $\mathcal{D}_-$.

Point Sets versus Ordered Structures. There is a strong connection between sets of labeled points and ordered structures with two total preorders.

With every finite ordered structure $\mathcal{O}$ with universe U and relations $\precsim_1$, $\precsim_2$, their induced successor relations S_1, S_2 and some unary relations one can associate a finite point set M in the plane and a bijection π such that the following statements hold.

- $u \prec_1 v$ in $\mathcal{O}$ if and only if $\pi(u) \nearrow \pi(v)$ or $\pi(u) \rightarrow \pi(v)$ or $\pi(u) \searrow \pi(v)$ in M.
- $u \prec_2 v$ in $\mathcal{O}$ if and only if $\pi(u) \nwarrow \pi(v)$ or $\pi(u) \uparrow \pi(v)$ or $\pi(u) \nearrow \pi(v)$ in M.
- $u \sim_1 v$ in $\mathcal{O}$ if and only if $\pi(u) \uparrow \pi(v)$ or $\pi(u) \downarrow \pi(v)$ or $\pi(u) = \pi(v)$ in M.
- $u \sim_2 v$ in $\mathcal{O}$ if and only if $\pi(u) \leftarrow \pi(v)$ or $\pi(u) \rightarrow \pi(v)$ or $\pi(u) = \pi(v)$ in M.
- $S_1(u,v)$ in $\mathcal{O}$ if and only if $I_x(\pi(u), \pi(v))$ in M.
- $S_2(u,v)$ in $\mathcal{O}$ if and only if $I_y(\pi(u), \pi(v))$ in M.

To this end, we can assign to every equivalence class of $\precsim_1$ a natural number in increasing order, likewise for $\precsim_2$. Then, $\pi(u).x$ is just the number of u's $\sim_1$-class and $\pi(u).y$ is the number of its $\sim_2$-class. Note that this construction might yield a multiset as there could be elements $u \neq v$ such that $u \sim_1 v$ and $u \sim_2 v$. However, in the following this case will not occur as we will require that $\precsim_1$ is a linear order.

Similarly, from every labeled point set M in $\mathbb{N}^2$, an ordered structure $\mathcal{O}$ (with universe M) can be obtained by defining

- $u \precsim_1 v$ if $u.x \leq v.x$ and
- $u \precsim_2 v$ if $u.y \leq v.y$

and by defining S_1 and S_2 accordingly.

Thus, points are equivalent with respect to $\precsim_2$ if they have the same y-coordinate.

Data Words. We are interested in a particular kind of ordered structures, *data words with ordered data*. In a nutshell, a data word with ordered data is a string in which every position carries a label from a finite alphabet *and* a value from a potentially infinite, ordered domain. In this paper, this domain will always be $\mathbb{N}$. As only a finite number of values can occur in a finite data strings this is not a restriction. More formally, a *data word s* over alphabet Σ is a finite sequence $(\sigma_1, d_1), \ldots, (\sigma_n, d_n)$ where $\sigma_i \in \Sigma$ and $d_i \in \mathbb{N}$, for every i. Such a data word can be represented in a natural way by an ordered structure whose universe is the set $\{1, \ldots, n\}$ of positions of s and which is equipped with a linear order $\leq_1$ and a successor relation S_1 on the positions. To represent the linear order on the data values the structure may have a total preorder $\precsim_2$. Thus, $i \precsim_2 j$ if $d_i \leq d_j$. Note that if the same data value occurs at different positions, $\precsim_2$ may indeed have non-singleton equivalence classes. Furthermore, the structure has one unary relation for each symbol of Σ. In logical formulas we simply write $\sigma(x)$ to denote that position x carries symbol σ. Note that data words are a special kind of ordered structures as every position is contained in exactly one of the unary relations.

In this article, we do not consider data words with the successor relation on positions. However, we allow a partial successor relation S_2 of the total preorder which translates to the successor relation on the data values as $(i,j) \in S_2$ if and only if $d_i + 1 = d_j$.
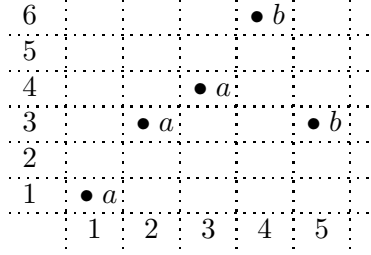
Figure 1: Finite set of labeled points corresponding to the data word $(a, 1)(a, 3)(a, 4)(b, 6)(b, 3)$. Columns are ordered by $<_1$ and rows are ordered by $\precsim_2$.

From the point of view of finite structures there is only one difference between ordered structures and data words: in data words every position carries exactly one symbol from some finite alphabet Σ, whereas ordered structures in general allow several unary relations that need not be disjoint. We thus add Σ to our notation of ordered structures to indicate that there is one unary relation per symbol in Σ and the unary relations partition the universe. For example, write $\text{FinOrd}(\Sigma, \leq_1, [\precsim_2, S_2])$ to denote predicate logic over data words with ordered data, alphabet Σ and domain $\mathbb{N}$ without successor on positions but with successor on values.

The correspondence between ordered structures and sets of labeled points naturally translates to a correspondence between data words with ordered data and sets of labeled points. Figure 1 shows the point set corresponding to the data word $w = \begin{pmatrix} a & a & a & b & b \\ 1 & 3 & 4 & 6 & 3 \end{pmatrix}$.

Logic. Two-variable logic is the restriction of predicate logic to formulas that only use (at most) two variables x and y. We denote two-variable logic by FO^2.

We denote two-variable logic on a restricted set $\mathcal{S}$ of structures by $\text{FO}^2(\mathcal{S})$. As an abbreviation, we denote $\text{FO}^2(\text{FinOrd}(\mathcal{O}))$, for a set $\mathcal{O}$ of predicates by $\text{FO}^2(\mathcal{O})$.

If $\mathcal{D}$ is a set of directions, we denote by $\text{FO}^2(\mathcal{D})$ two-variable logic with binary relations from $\mathcal{D}$ and unary relations from Prop. We interpret $\text{FO}^2(\mathcal{D})$ logic over non-empty sets of Prop-points in $\mathbb{N}^2$ where Prop has a proposition e_i, for every unary relation symbol U_i.

3. Upper Bounds for Two-variable Logic on Ordered Structures

The following theorem states our main result.

Theorem 3.1. Finite satisfiability of $\text{FO}^2(\leq_1, [\precsim_2, S_2])$ is EXPSPACE-complete.

Corollary 3.2. Finite satisfiability of FO^2 over data words with a linear order on the positions and a linear order and a corresponding successor relation on the data values can be decided in EXPSPACE.

It is worthwhile to compare Corollary 3.2 with the following result from [BMS$^+$06].

Proposition 3.3. Finite satisfiability of FO^2 over data words with a linear order and its corresponding successor relation on the positions and a linear order on the data values is undecidable.

The two results might appear contradictory at first sight. However, translated into the language of orders the two settings are indeed different. In Corollary 3.2 there is a total preorder with a corresponding successor relation plus a linear order, whereas in Proposition 3.3 there is a linear order with its underlying successor relation plus a total preorder. We can conclude that the additional total preorder yields more expressive power in that it allows to encode Post's correspondence problem in a two-variable fashion as shown in [BMS$^+$06]..

The lower bound of Theorem 3.1 will be shown in the following section as Theorem 4.1. The upper bound proof is given in this section and consists of three main steps.

(3.1) We first reduce finite satisfiability of $\mathrm{FO}^2(\leq_1, [\precsim_2, S_2])$ to finite satisfiability of $\mathrm{FO}^2(\mathcal{D})$-formulas of a syntactically restricted form in polynomial time.

(3.2) The latter problem is then reduced to the two-dimensional labeled point problem (2LPP), which will be defined below, in exponential time.

(3.3) Finally, we show that 2LPP can be solved in polynomial space.

3.1. **From ordered structures to labeled point sets.** By $\mathrm{FO}^2(\mathcal{D}_-, I_y)$ we denote two-variable logic with binary atoms $x \circ_d y$ using directions $\circ_d$ from $\mathcal{D}_-$ (in infix notation) and binary predicate I_y. A sentence φ of this logic is *semi-positive* if it fulfills the following conditions.

(i) φ is in negation normal form (NNF).

(ii) There are no negated literals $\neg(x \circ_d y)$ in φ. However, the occurrence of negated atoms $x \neq y$ is not restricted.

(iii) Literals $I_y(x, y)$ and $\neg I_y(x, y)$ only appear in conjunction with positive atoms $\circ_d \in \mathcal{D}_-$.

The correspondence between ordered structures and finite sets of labeled points can be exploited to show the following result.

Proposition 3.4. For each $\mathrm{FO}^2(\leq_1 [\precsim_2, S_2])$-sentence φ a semi-positive $\mathrm{FO}^2(\mathcal{D}_-, I_y)$-sentence ψ can be computed in polynomial time such that ψ and φ are equivalent with respect to finite satisfiability over structures with a linear order and a total preorder with induced successor relation.

Proof. We assume without loss of generality that φ is given in negation normal form. The target formula ψ is of the form $\chi \wedge \varphi'$ where χ ensures that no two points are on the same vertical line and φ' is the actual translation of φ into the setting of labeled points. More precisely,

$$\chi = \forall x \forall y \ \ x = y \vee \bigvee_{\circ_d \in \mathcal{D}_-} x \circ_d y,$$

and φ' is obtained from φ by translating its binary literals as follows.

φ	φ'
$x \leq_1 y$	$x = y \vee x \nearrow y \vee x \rightarrow y \vee x \searrow y$
$\neg(x \leq_1 y)$	$x \nwarrow y \vee x \leftarrow y \vee x \swarrow y$
$x \precsim_2 y$	$x = y \vee x \rightarrow y \vee x \leftarrow y \vee x \nearrow y \vee x \nwarrow y$
$\neg(x \precsim_2 y)$	$x \searrow y \vee x \swarrow y$
$S_2(x, y)$	$(I_y(x, y) \wedge x \nearrow y) \vee (I_y(x, y) \wedge x \nwarrow y)$
$\neg S_2(x, y)$	$(\neg I_y(x, y) \wedge x \nearrow y) \vee (\neg I_y(x, y) \wedge x \nwarrow y) \vee x = y \vee x \leftarrow y \vee x \rightarrow y \vee x \swarrow y \vee x \searrow y$

The translation of literals as $y \leq_1 x$ is analogous.

The correctness of this translation relies on the fact that χ ensures that no two points are on the same vertical line. For example, for $x \leq_1 y$, no disjuncts $x \uparrow y$ or $x \downarrow y$ are needed. It is easy to verify that ψ is indeed semi-positive. $\qquad\square$

3.2. **From Two-Variable Logic to Two-Dimensional Constraints.** We next define the 2-dimensional labeled point problem, 2LPP. For an alphabet Σ, a Σ-*labeled point* p is an element from $\mathbb{N}^2 \times \Sigma$. Of course, Prop-labeled points can be considered as Σ-labeled points with $\Sigma = 2^{\text{Prop}}$. However, the number of symbols is exponential in the number of propositions. Likewise, Σ-labeled points can be encoded as points over a suitable set of propositions but in that direction the encoding is less canonical. One could, e.g., use one proposition per symbol of Σ or a logarithmic number of propositions. We write $p.l$ for the label of a Σ-labeled point p.

A *directional constraint* is a pair $(\circ_d, s_d)$ from $\mathcal{D} \times \{S, \neg S\}$. We denote the set of directional constraints by $\mathcal{C}$. By $\mathcal{C}_-$ we denote the set of directional constraints $(\circ_d, s_d)$ with $\circ_d \in \mathcal{D}_-$.

We say a pair (p, q) of Σ-labeled points *satisfies* a directional constraint $d = (\circ_d, s_d)$, if $p \circ_d q$ and the following conditions are fulfilled.

- $p.y + 1 = q.y$, if $s_d = S$.
- $p.y + 1 \neq q.y$, if $s_d = \neg S$.

Note that the constraints $(\rightarrow, S)$ and $(\leftarrow, S)$ are not satisfied for any pair of points. By $\neq$ we denote an *inequality constraint* (and there is only one such constraint). A pair (p, q) satisfies the inequality constraint $\neq$ if $p \neq q$.

A *position constraint* is either a directional or an inequality constraint.

An *existential constraint ($\exists$-constraint)* is a pair (σ, E) where $\sigma \in \Sigma$ and E is a possibly empty set of pairs (τ, d), where $\tau \in \Sigma$ and d is a position constraint from $\mathcal{C}_- \cup \{\neq\}$. For a set $\mathcal{P}$ of Σ-labeled points and a point p, we say p *satisfies* an $\exists$-constraint (σ, E) if either $p.l \neq \sigma$ or there is $q \in \mathcal{P}$ such that, for some (τ, d) in E, $q.l = \tau$ and (p, q) satisfies d.

A *universal constraint ($\forall$-constraint)* is a tuple (σ, τ, d) where $\sigma, \tau \in \Sigma$ and d is a directional constraint. A pair (p, q) of points from M *satisfies* a $\forall$-constraint (σ, τ, d) if it is *not the case* that $p.l = \sigma$, $q.l = \tau$, and (p, q) satisfies d.

An input $L = (\Sigma, C_\exists, C_\forall)$ to the *two-dimensional labeled point problem (2LPP)* consists of an alphabet Σ, a set $C_\exists$ of existential constraints and a set $C_\forall$ of universal constraints. A non-empty set $M \subseteq \mathbb{N}^2 \times \Sigma$ is a *solution* of L if every point of M satisfies all constraints from $C_\exists$ and every pair of distinct points satisfies all constraints from $C_\forall$. It should be noted that $C_\exists$ specifies required patterns whereas $C_\forall$ specifies forbidden patterns.

Proposition 3.5. From every semi-positive $\text{FO}^2(\mathcal{D}_-, I_y)$-sentence φ an instance L of 2LPP can be computed in exponential time such that φ is finitely satisfiable if and only if L has a finite solution.

Proof.   First, φ can be translated into a semi-positive $\text{FO}^2(\mathcal{D}_-, I_y)$-sentence φ' in Scott normal form (SNF)

$$\forall x \forall y\ \psi(x, y) \wedge \bigwedge_{i=1}^{m} \forall x \exists y\ \psi_i(x, y),$$

that has a finite model if and only if φ has a finite model. The translation can be done in a way that ensures that

(1) ψ and all ψ_i are quantifier-free semi-positive formulas,
(2) φ' is of linear size in the size of φ,
In general, φ' uses more unary relation symbols than φ.

The translation mimics the proof of Theorem 2.1 in [GO99]. However, we need to be a bit careful to get φ' semi-positive.

The transformation is done in several rounds. After each round, a formula of the form $\theta_i \wedge \varphi_i$ is obtained that is equivalent to φ with respect to (finite) satisfiability. Here, θ_i is already in SNF and φ_i has i quantifiers less than φ. Furthermore, φ_i uses additional new unary predicates $P_1, \ldots, P_i$ and is semi-positive. Initially, we have $\theta_0 = \top$ and $\varphi_0 = \varphi$. In round i, a subformula $\chi_i(x)$ of the form $\exists y\, \rho_i(x, y)$ or the form $\forall y\, \rho_i(x, y)$ with quantifier-free formula ρ_i is chosen from φ_{i-1}. In the former case,

- $\theta_i = \theta_{i-1} \wedge \forall x \exists y (P_i(x) \to \rho_i(x, y))$ and
- φ_i is obtained from φ_{i-1} by replacing $\exists y\, \rho_i(x, y)$ with $P_i(x)$.

In the latter case,

- $\theta_i = \theta_{i-1} \wedge \forall x \forall y (P_i(x) \to \rho_i(x, y))$ and
- φ_i is obtained from φ_{i-1} by replacing $\forall y\, \rho_i(x, y)$ with $P_i(x)$.

Clearly, this process has as many rounds as the initial formula φ has quantifiers. It is easy to see that each θ_i is semi-positive and in SNF and each φ_i is semi-positive as well. Let $\theta_k \wedge \varphi_k$ be the resulting formula. φ_k does not contain any quantifiers and is equivalent to $\forall x \forall y\, \varphi_k$. Therefore, $\varphi' = \theta_k \wedge \forall x \forall y\, \varphi_k$ is in SNF (after combining all $\forall\forall$-conjuncts into one $\forall\forall$-formula). Furthermore, the size of φ' is linear in the size of φ. However, the signature has been extended by k unary relation symbols.

It remains to show that φ' is finitely satisfiable if and only if φ is finitely satisfiable. To this end, we show by induction on i that, for every $i > 0$, $\theta_i \wedge \varphi_i$ is finitely satisfiable if and only if $\theta_{i-1} \wedge \varphi_{i-1}$ is finitely satisfiable. Indeed, if $A_{i-1} \models \theta_{i-1} \wedge \varphi_{i-1}$, a model[4] $A_i = (A_{i-1}, P_i)$ for φ_i can be obtained from A_{i-1} by letting $P_i = \{a \mid A \models \chi_i(a)\}$. On the other hand, assume that $A_i = (A_{i-1}, P_i)$ is a model for $\theta_i \wedge \varphi_i$ in the case that $\chi_i(x)$ is of the form $\exists y\, \rho_i(x, y)$. Thus, $A_i \models \theta_i$ guarantees that for every a with $A_i \models P_i(a)$ the formula $\exists y\, \rho_i(a, y)$ is satisfied by A_i and thus also $A_{i-1} \models \exists y\, \rho_i(a, y)$ (as P_i does not occur in ρ_i). As φ_{i-1} is in NNF, the occurrence of $\exists y\, \rho_i(a, y)$ is not in the scope of any negations, therefore $A_i \models \varphi_i$ implies $A_{i-1} \models \varphi_{i-1}$. Furthermore, $A_i \models \theta_i$ also implies $A_{i-1} \models \theta_{i-1}$. If $\chi_i(x)$ is of the form $\forall y\, \rho_i(x, y)$ the reasoning is analogous. This completes the proof that φ' is finitely satisfiable if and only if φ is finitely satisfiable.

In order to continue the translation into a 2LPP-instance, we next describe how to obtain $\forall$-constraints from ψ. A *full atomic type for x and y* consists of

- a full atomic type σ for x,
- a full atomic type τ for y, and
- a conjunction of binary literals concerning x and y which can be either
  - $x = y$ or
  - the conjunction of a direction atom $x \circ_d y$ with $\circ_d \in \mathcal{D}$ and one of the formulas $S(x, y)$ or $S(y, x)$ or $\neg S(x, y) \wedge \neg S(y, x)$.

It should be noted that we do not allow $x \neq y$ in full atomic types as this literal is equivalent to the disjunction of all other possible relationships besides $x = y$.

---

[4]Here, (B, P) denotes the extension of a structure B by a relation P.

We further note that each full atomic type which does not contain the formula $x = y$ corresponds to a universal constraint. For instance, $\sigma(x) \wedge \tau(y) \wedge x \nearrow y \wedge \neg S(x,y) \wedge \neg S(y,x)$ corresponds to $(\sigma, \tau, (\nearrow, \neg S))$ and $\sigma(x) \wedge \tau(y) \wedge x \searrow y \wedge S(y,x)$ corresponds to $(\tau, \sigma, (\nwarrow, S))$. The reader should note the direction change and the switch of σ and τ in the latter example. It is due to the translation of the atom $S(y,x)$.

For every full atomic binary or unary type, ψ determines whether this type is allowed or forbidden in a model of φ'. We let Σ be the set of all *unary* atomic types σ of x for which $\sigma(x) \wedge \sigma(y) \wedge x = y$ is *allowed*. Furthermore, we let $C_\forall$ be the set of all $\forall$-constraints that are obtained from the *forbidden binary types*. Clearly, the size of $C_\forall$ is at most exponential in φ.

It should finally be noted that, although φ only uses directions from $\mathcal{D}_-$, the resulting universal constraints might contain the directions $\uparrow$ or $\downarrow$. This is for instance necessary in the case that ψ implies $\forall x \forall y \ (\sigma(x) \wedge \sigma(y) \rightarrow x = y)$, since here universal constraints $(\sigma, \sigma, \circ_d)$ are needed for all $\circ_d \in \mathcal{D}$.

By transforming every ψ_i into DNF and some additional simple steps, every conjunct $\forall x \exists y \ \psi_i(x,y)$ of φ' can be rewritten as

$$\forall x \bigwedge_{j=1}^{K} \left(\sigma_j(x) \rightarrow \exists y \bigvee_{\ell=1}^{M} (\tau_{j\ell} \wedge \psi_{j\ell}) \right),$$

where the σ_j describe pairwise distinct full atomic types, every $\tau_{j\ell}$ is a full atomic type and $\psi_{j\ell}$ is one of the following.
(1) a formula $x \circ_d y \wedge S(x,y)$ with $\circ_d \in \mathcal{D}_-$,
(2) a formula $x \circ_d y \wedge \neg S(x,y)$ with $\circ_d \in \mathcal{D}_-$,
(3) an atom $x = y$,
(4) a literal $x \neq y$.

For each $j \leq K$, the disjuncts with formulas $\psi_{j\ell}$ of types (1) and (3) can be combined into one $\exists$-constraint. Disjuncts with formulas of type (2) can be treated as follows. If $\sigma_j = \tau_{j,\ell}$ then the j-th conjunct can be completely removed as it becomes a tautology. If $\sigma_j \neq \tau_{j,\ell}$, the disjunct can be deleted as it can not be satisfied. If this results in an empty disjunction then σ_j is deleted from Σ. The numbers K and M are at most exponential in $|\varphi|$.

Altogether, we obtain an instance $L = (\Sigma, C_\exists, C_\forall)$ of 2LPP such that a Σ-labeled point set is a solution to L if and only if the corresponding labeled point set M is a model of φ. Furthermore, the size of L is at most exponential in $|\varphi|$ and the construction of L from φ can be done in exponential time. $\square$

3.3. **2LPP is in** PSPACE. In order to prove the upper bound of Theorem 3.1 it would be sufficient to give a polynomial space algorithm that decides whether a given 2LPP-instance has a solution where no two points are on the same vertical. However, there is an algorithm that decides the existence of *any* finite solution.

Proposition 3.6. Whether an instance L of 2LPP has a finite solution can be decided in polynomial space.

The algorithm follows a plane sweep approach, that is, it guesses a solution for L (horizontal) line by line. The crucial point is to show that this is possible in polynomial space.

With every line l of a set of labeled points we associate a *profile* that, roughly speaking, contains all neccessary information to validate the constraints from L for points on line l. Profiles respecting the constraints from L will be called *valid*. We will prove that a solution of L can be constructed from a sequence of valid profiles where every pair of successive profiles is *consistent*. Then, the algorithm guesses a sequence of profiles and verifies validity and consistency locally. Since the size of each profile is polynomial in the size of L, this is a polynomial space algorithm proving Proposition 3.6.

Now, we fill in the details of the proof, and start by giving some definitions.

Profile of a Horizontal Line. A *horizontal line* of a set $\mathcal{P}$ of labeled points is a number $r \in \mathbb{N}$ such that $\mathcal{P}$ contains points p and q with $p.y \leq r \leq q.y$. Intuitively, a horizontal line is just a line that lies between the top-most and the bottom-most point of $\mathcal{P}$. We say that a point from $\mathcal{P}$ *lies on* r if $p.y = r$. We associate with every horizontal line r of a set $\mathcal{P}$ a profile $\text{Pro}(r)$ that represents information about the points on r and on other points of $\mathcal{P}$ that are "important" with respect to $\exists$-constraints and $\forall$-constraints. Intuitively, for each symbol and each "direction" the profile contains the x-value of the leftmost and the rightmost point in that direction.

Let $D_P := \{\cdot\} \cup (\{\uparrow, \downarrow\} \times \{S, \neg S\})$ denote the set of *profile constraints*. For two rows r, s we denote by $d(r, s)$ the profile constraint of s relative to r, that is

$$
d(r, s) = \begin{cases}
(\uparrow, \neg S) & \text{if } s > r + 1, \\
(\uparrow, S) & \text{if } s = r + 1, \\
\cdot & \text{if } s = r, \\
(\downarrow, S) & \text{if } s = r - 1, \\
(\downarrow, \neg S) & \text{if } s < r - 1.
\end{cases}
$$

A *profile* is a sequence $P = A_1 \ldots A_k$ of non-empty sets of pairs from $\Sigma \times D_P$, such that
- every set A_i contains at most one pair of the form $(\sigma, \cdot)$,
- for each σ there are at most two indices i, j such that $(\sigma, \cdot) \in A_i$ and $(\sigma, \cdot) \in A_j$, and
- for every i there is some $(\sigma, d) \in A_i$ such that i is either the minimal or the maximal index of a set containing (σ, d).

The latter two conditions ensure that the number of sets in a profile is at most $10|\Sigma|$. A profile is a *top profile*, if none of its sets contains any pair $(\sigma, (\uparrow, S))$ or $(\sigma, (\uparrow, \neg S))$. Analogously a *bottom profile* is a profile that does not contain any pairs $(\sigma, (\downarrow, S))$ or $(\sigma, (\downarrow, \neg S))$.

Let, for the following, a finite labeled point set $\mathcal{P} \subseteq \mathbb{N}^2$ be fixed. Let $\mathcal{P}'$ be the set of all points p from $\mathcal{P}$ for which $p.x$ is minimal or maximal within all points with label $p.l$ in row $p.y$. More precisely let

$$
\begin{aligned}
\mathcal{P}' = &\{p \in \mathcal{P} \mid \neg \exists q \in \mathcal{P} : q.l = p.l, q.y = p.y, q.x < p.x\} \\
&\cup \{p \in \mathcal{P} \mid \neg \exists q \in \mathcal{P} : q.l = p.l, q.y = p.y, q.x > p.x\}
\end{aligned}
$$

As $\mathcal{P}'$ has, for each label σ, at most two σ-labeled points per row, it has at most $2|\Sigma|$ points per row altogether.

We describe next, how the profile $\text{Pro}(r)$ is constructed, for every horizontal row r of $\mathcal{P}'$. Let thus, a row r be fixed in the following. For each $\sigma \in \Sigma$ and every profile
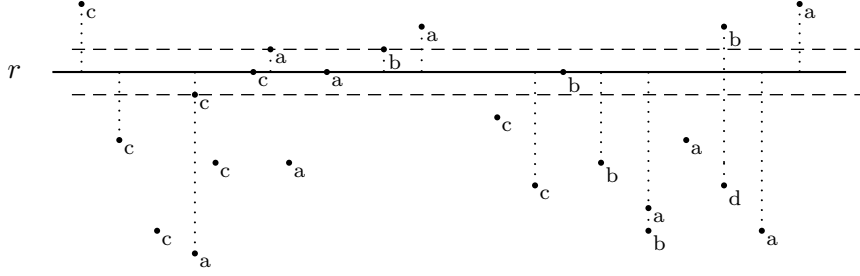
Figure 2: Line with profile $(c,\uparrow)(c,\downarrow)\{(a,\downarrow),(c,(\downarrow,S))\}(c,\cdot)(a,(\uparrow,S))(a,\cdot)(b,(\uparrow,S))$ $(a,\uparrow)(c,\downarrow)(b,\cdot)(b,\downarrow)(b,\downarrow)\{(b,\uparrow),(d,\downarrow)\}(a,\downarrow)(a,\uparrow)$. Dotted vertical lines indicate points contributing to the profiles. The dashed horizontal lines indicate the predecessor and successor line, respectively. Singleton sets in profiles are written without braces. Profile constraints $(\uparrow,\neg S)$ and $(\downarrow,\neg S)$ are abbreviated as $\uparrow$ and $\downarrow$, respectively. Thus, e.g., the first pair $(c,\uparrow)$ abbreviates the full notation $\{(c,(\uparrow,\neg S))\}$.

constraint $d \in D_P$, we choose a point $q = q(\sigma, d, \min, r) \in \mathcal{P}'$ with minimal value $q.x$ such that $d(r, q.y) = d$ and a point $q = q(\sigma, d, \max, r) \in \mathcal{P}'$ with maximal value $q.x$ such that $d(r, q.y) = d$. Note that $q(\sigma, d, \min, r)$ and $q(\sigma, d, \max, r)$ can be equal and do not necessarily exist for every σ, d and r.

Let $A'_{r,1}, \ldots, A'_{r,k_r}$ be the equivalence classes of points of the form $q = q(\sigma, d, \min, r)$ and $q = q(\sigma, d, \max, r)$ with arbitrary σ and d and the same value $q.x$, ordered by $q.x$. For every $i \le k_r$, we denote by $A'_{r,i}.x$ the value of the x-coordinate of the points in $A'_{r,i}$.

Even though the classes in $A'_{r,1}, \ldots, A'_{r,k_r}$ contain all points that are relevant for the validity of the given constraints, it might be necessary to add some more points for technical reasons, as follows.

For each $i \le k_r$, let the set $A''_{r,i}$ be defined as follows.

$$A''_{r,i} = A'_{r,i} \cup \{q \in \mathcal{P}' \mid \forall s(r < s \le q.y \rightarrow \exists j\, A'_{s,j}.x = A'_{r,i}.x)\}$$
$$\cup \{q \in \mathcal{P}' \mid \forall s(r > s \ge q.y \rightarrow \exists j\, A'_{s,j}.x = A'_{r,i}.x)\}$$

For each $i \le k_r$ we define the set $A_{r,i}$ as

$$A_{r,i} = \{(q.l, d(r,q)) \mid q \in A''_{r,i}\}$$

and, finally, we let

$$\mathrm{Pro}(r) = A_{r,1}, \ldots, A_{r,k_r}$$

For every $i \le k_r$, we call $A'_{r,i}.x$ the *x-value of* position i in $\mathrm{Pro}(r)$.

It is easy to see that $\mathrm{Pro}(r)$ is indeed a profile as defined above. We refer to Figure 2 for an example profile extraction.

Let $y_{\max}$ be the maximal y-value of a point in $\mathcal{P}'$ and $y_{\min}$ be the minimal y-value of a point in $\mathcal{P}'$. The *profile sequence* $\mathrm{PSeq}(\mathcal{P})$ of $\mathcal{P}$ is the sequence $\mathrm{Pro}(y_{\min})\mathrm{Pro}(y_{\min} + 1) \ldots \mathrm{Pro}(y_{\max})$.

Valid Profiles. Next we define some necessary conditions that a profile of a horizontal line in a solution of an 2LPP-instance L must fulfill.

Let $A = A_1 \ldots A_k$ be a profile. We say that a pair $(\sigma, \cdot) \in A_i$ *fulfills a directional constraint* $(\circ_d, s_d) \in \mathcal{D} \times \{S, \neg S\}$ *with respect to a pair* $(\tau, c) \in A_j$, if the following conditions are satisfied, where $c = (\circ_c, s_c)$ or $c = \cdot$:

- (Horizontal Conditions)
  - If $\circ_d \in \{\nwarrow, \leftarrow, \swarrow\}$ then $i > j$.
  - If $\circ_d \in \{\nearrow, \rightarrow, \searrow\}$ then $i < j$.
  - If $\circ_d \in \{\leftarrow, \rightarrow\}$ then $c = \cdot$.
  - If $\circ_d \notin \{\leftarrow, \rightarrow\}$ then $c \neq \cdot$ and $s_d = s_c$.
- (Vertical Conditions)
  - If $\circ_d \in \{\nwarrow, \uparrow, \nearrow\}$ then $\circ_c =\uparrow$.
  - If $\circ_d \in \{\swarrow, \downarrow, \searrow\}$ then $\circ_c =\downarrow$.
  - If $\circ_d \in \{\uparrow, \downarrow\}$ then $i = j$.

A pair $(\sigma, \cdot) \in A_i$ *fulfills an inequality constraint* $\neq$ *with respect to a pair* $(\tau, c) \in A_j$, if $i \neq j$ or $c \neq \cdot$.

A profile $A = A_1 \ldots A_k$ is *valid* with respect to an existential constraint (σ, E) if, for every i and every $(\sigma, \cdot) \in A_i$, there is some $(\tau, d) \in E$ and some (τ, c) in some A_j such that $(\sigma, \cdot)$ fulfills d with respect to (τ, c).

The profile A is *valid* with respect to a universal constraint (σ, τ, d) if there *do not exist* any i, j and pairs $(\sigma, \cdot) \in A_i$ and $(\tau, c) \in A_j$ such that $(\sigma, \cdot)$ fulfills d with respect to (τ, c).

A profile A is *L-valid* if it is valid with respect to all (existential and universal) constraints from L.

Example 3.7. Let L be an 2LPP instance and $A = A_1 \ldots A_k$ a profile. If L contains an $\exists$-constraint $(\sigma, \{(\tau_1, (\nwarrow, \neg S)), (\tau_2, \neq), (\tau_3, (\searrow, S))\})$ then for any $(\sigma, \cdot)$ in A_i there has to be

- an occurrence $(\tau_1, (\uparrow, \neg S))$ in A_j for some $j < i$,
- an occurrence of a pair (τ_2, c) in A_j with $c \neq \cdot$ if $j = i$, or
- an occurrence $(\tau_3, (\downarrow, S))$ in A_l for some $l > i$.

If L contains a $\forall$-constraint $(\sigma, \tau, (\swarrow, S))$ then there should be no $i > j$ with $(\sigma, \cdot) \in A_i$ and $(\tau, (\downarrow, S)) \in A_j$.

Consistent Pairs of Profiles. Let $A = A_1 \ldots A_k$ and $B = B_1 \ldots B_m$ be profiles. The pair (A, B) is said to be *consistent,* if there is a function

$$s : \big(\{a\} \times \{1, \ldots, k\} \cup \{b\} \times \{1, \ldots, m\}\big) \to \mathbb{Q}$$

that is strictly monotone in its second parameter and fulfills the following conditions (C1) and (C2) for every $\sigma \in \Sigma$.

(C1) (a) If for some i, A_i contains $(\sigma, (\uparrow, S))$ then there is some j, such that B_j contains $(\sigma, \cdot)$ and $s(a, i) = s(b, j)$.
   (b) If for some j, B_j contains $(\sigma, \cdot)$ then there is some i such that A_i contains $(\sigma, (\uparrow, S))$ and $s(a, i) = s(b, j)$.

(C2) (a) If for some i, A_i contains $(\sigma, (\uparrow, \neg S))$, there is some j such that B_j contains $(\sigma, (\uparrow, S))$ or $(\sigma, (\uparrow, \neg S))$ and $s(a, i) = s(b, j)$.
   (b) If for some j, B_j contains $(\sigma, (\uparrow, S))$ or $(\sigma, (\uparrow, \neg S))$ and j is minimal or maximal with this property then there is some i with $s(a, i) = s(b, j)$ such that A_i contains $(\sigma, (\uparrow, \neg S))$.

(c) If for some j, B_j contains $(\sigma, (\uparrow, S))$ or $(\sigma, (\uparrow, \neg S))$ and there is some i with $s(a, i) = s(b, j)$ then $(\sigma, (\uparrow, \neg S)) \in A_i$.

The analogous conditions hold with respect to occurrences of $(\sigma, (\downarrow, S))$ and $(\sigma, (\downarrow, \neg S))$ in B.

Intuitively, s "synchronizes" points in A with points in B by mapping them to the same "x-coordinate". The conditions (C1) ensure that successive roles fit together with respect to profile constraints of the form $(\uparrow, S)$ and $(\downarrow, S)$. More precisely, if $(\sigma, (\uparrow, S))$ occurs in A then B should have a σ-point at the same vertical position. The conditions of (C2) ensure consistency for profile constraints of the form $(\uparrow, \neg S)$ and $(\downarrow, \neg S)$. (C2a) ensures that the requirement that there is a σ-point above a certain position of A is reflected in B. (C2b) ensures that A contains the full information about minimal and maximal points above B. Finally, (C2c) guarantees that if a point u for A is synchronized with a point v for B then its profile carries the full information about the points above v.

If $s(a, i) = s(b, j)$ we say that position i of A and position j of B are *synchronized*. We additionally require for s that unless requested by (C1) or (C2) (or their downward counterparts) $s(a, i) \neq s(b, j)$ holds for every i and j. This requirement guarantees that, even though s is not uniquely determined, the set of pairs of synchronized positions is unique.

A sequence $\mathcal{S} = P_1 \ldots P_N$ of profiles is called *consistent*, if P_1 is a bottom profile, P_N is a top profile and (P_i, P_{i+1}) is consistent for every $i \in \{1, \ldots, N-1\}$.

If $\mathcal{S} = P_1 \ldots P_N$ is a consistent sequence of profiles, we say that a position k of P_i is *connected to* position ℓ of P_j, where $j > i$ if

(1) $j = i + 1$ and the two positions are synchronized or
(2) position k in P_i is synchronized with some position in P_{i+1} which in turn is connected to position ℓ in P_j.

The following proposition characterizes the 2LPP-instances with finite solutions in terms of existence of consistent sequences of valid profiles and is key to our algorithm.

Proposition 3.8. Let L be an instance of 2LPP. Then L has a finite solution if and only if there is a finite consistent sequence of L-valid profiles.

Proof. Let $\mathcal{P}$ be a model of L. We claim that $\mathrm{PSeq}(\mathcal{P}) = P_1 \ldots P_N$ is a consistent sequence of L-valid profiles. Let $\mathcal{P}'$ be defined as in the definition of $\mathrm{PSeq}(\mathcal{P})$. To this end, let $k \leq N$ be arbitrarily chosen such that $P = P_k = A_1, \ldots, A_\ell$, for some ℓ. We show that P is valid with respect to all existential and universal constraints from L.

Let first (σ, E) be an $\exists$-constraint from L and let $(\sigma, \cdot) \in A_i$, for some $i \leq \ell$. Let r be a horizontal line of $\mathcal{P}$ that corresponds to P_k and let p be the point on r that corresponds to $(\sigma, \cdot) \in A_i$. As $\mathcal{P}$ is a solution for L there is some $(\tau, d) \in E$ and a point $q \in \mathcal{P}$ with label τ such that (p, q) satisfies d. We first consider the case that d is a directional constraint $(\circ_d, s_d)$.

- If $\circ_d = \nearrow$ then q is to the northeast of p, in particular, $p.y < q.y$ and $p.x < q.x$. Let q' be $q(\tau, (\uparrow, s_d), \max, r)$. By definition of $q(\tau, (\uparrow, s_d), \max, r)$, it holds $p.x < q.x \leq q'.x$ and $p.y + 1 < q'.y$ in case $s_d = \neg S$ (in case, $s_d = S$, we get $p.y + 1 = q'.y$ instead). Therefore, $(\tau, (\uparrow, s_d))$ occurs in A_j, for some $j > i$. Clearly, $(\sigma, \cdot)$ and $(\tau, (\uparrow, s_d))$ are consistent with respect to d. Analogous arguments can be applied if $\circ_d \in \{\nwarrow, \swarrow, \searrow\}$.
- If $\circ_d = \rightarrow$ then q is to the right (east) of p, in particular, it holds $p.y = q.y$ and $p.x < q.x$. Without loss of generality, $q = q(\tau, \cdot, \max, r)$. By definition of $q(\tau, \cdot, \max, r)$, it holds

$p.y = q.y$ and $p.x < q.x$. Therefore, $(\tau, \cdot)$ occurs in A_j, for some $j > i$. Clearly, $(\sigma, \cdot)$ and $(\tau, \cdot)$ are consistent with respect to d. An analogous argument can be applied if $\circ_d = \leftarrow$. If d is an inequality constraint, then $q \neq p$ but the relative position of p and q can be arbitrary. If $p.y < q.y$ we can argue as above with $q' = q(\tau, (\uparrow, \neg S), \max, r)$ or $q' = q(\tau, (\uparrow, S), \max, r)$ (if the former does not exist). Note however, that the corresponding pair $(\tau, (\uparrow, S))$ or $(\tau, (\uparrow, \neg S))$ could be in A_i. However, as the second component of the pair is not $\cdot$ this is sufficient. If $p.y > q.y$ we can argue analogously. If $p.y = q.y$ and $p.x < q.x$ we can argue as above in the case $\circ_d = \rightarrow$. Likewise, we can argue as above in the case $\circ_d = \leftarrow$ if $p.y = q.y$ and $p.x > q.x$. Altogether we can conclude that P is valid with respect to (σ, E).

Let now (σ, τ, d) be a $\forall$-constraint from L and let $(\sigma, \cdot) \in A_i$, for some i, $p = q(\sigma, \cdot, m, r)$ for some $m \in \{\min, \max\}$ the corresponding point and $(\tau, c) \in A_j$ for some j. Towards a contradiction let us now assume that $(\sigma, \cdot) \in A_i$ and $(\tau, c) \in A_j$ are consistent with respect to $d = (\circ_d, s_d)$. We distinguish cases based on $\circ_d$.

- We first consider the case $\circ_d = \nearrow$. Thus, we can conclude that $i < j$, $c \neq \cdot$. Hence $c = (\circ_c, s_c)$ for some $\circ_c, s_c$ and we can further conclude $s_d = s_c$ and $\circ_c = \uparrow$. Let $q \in \mathcal{P}'$ be the point corresponding to $(\tau, c) \in A_j$. Clearly, $q.l = \tau$ and (p, q) satisfy d and therefore $\mathcal{P}$ is not a solution for L, the desired contradiction. For $\circ_d \in \{\nwarrow, \swarrow, \searrow\}$ we can argue analogously.

- We next consider the case $\circ_d = \rightarrow$. Thus, we can conclude that $i < j$ and $c = \cdot$. But then it is easy to see that $p \rightarrow q$, where q is the point corresponding to $(\tau, \cdot)$ in A_j and therefore $\mathcal{P}$ is not a solution for L, the desired contradiction. For $\circ_d = \leftarrow$ we can argue analogously.

- Finally, we consider the case $\circ_d = \uparrow$. Thus, we can conclude $c \neq \cdot$ and therefore $c = (\circ_c, s_c)$ for some $\circ_c, s_c$. Furthermore, $\circ_c = \uparrow$, $s_c = s_d$ and $i = j$. But then it is easy to see that $p \uparrow q$, where q is the point corresponding to $(\tau, (\uparrow, s_c))$ in A_j and therefore $\mathcal{P}$ is not a solution for L, the desired contradiction. For $\circ_d = \downarrow$ we can argue analogously.

Altogether we can conclude that P is valid with respect to (σ, τ, d) and therefore P and all profiles in $\mathrm{PSeq}(\mathcal{P})$ are L-valid.

We continue by showing that $\mathrm{PSeq}(\mathcal{P})$ is consistent. Clearly, P_1 is a bottom profile and P_N is a top profile as there are no points in $\mathcal{P}'$ below the row corresponding to P_1 or above the row corresponding to P_N. It thus remains to be shown that, for every $k < N$, P_k and P_{k+1} are consistent.

Let $P_k = A_1 \ldots A_\ell$ and $P_{k+1} = B_1 \ldots B_m$ be profiles and let r and $r+1$ be the horizontal rows of P_k and P_{k+1}, respectively.

We define the synchronizing function

$$s : \big(\{a\} \times \{1, \ldots, \ell\} \cup \{b\} \times \{1, \ldots, m\}\big) \rightarrow \mathbb{Q}$$

as follows. For $u \leq \ell$, we let $s(a, u)$ be the x-value of A_u and for $v \leq m$, we let $s(b, v)$ be the x-value of B_v. We need to show conditions (C1) and (C2).

Towards (C1a), let $(\sigma, (\uparrow, S)) \in A_i$, for some i. By construction of P_k, there is a σ-labeled point $q \in \mathcal{P}'$ with $q.y = r + 1$ and $q.x$ is the x-value of position i in P_k. Therefore, there is a j such that $(\sigma, \cdot) \in B_j$ and $s(a, i) = s(b, j)$. Towards (C1b), let $(\sigma, \cdot) \in B_j$ and let $q \in \mathcal{P}'$ be its corresponding point. As there are at most two σ-labeled points on $r + 1$, $q = (\sigma, (\uparrow, S), m, r)$, for some $m \in \{\min, \max\}$. Therefore, by definition of $\mathrm{Pro}(r)$, there is i such that $(\sigma, (\uparrow, S)) \in A_i$ and $s(a, i) = s(b, j)$. Analogously, for $(\sigma, (\downarrow, S)) \in A_i$.

Towards (C2a), let $(\sigma, (\uparrow, \neg S)) \in A_i$, for some i. Let $q = q(\sigma, (\uparrow, \neg S), m, r)$ be the corresponding point in $\mathcal{P}'$, for some $m \in \{\min, \max\}$. By definition of $\text{Pro}(r)$, either $q = q(\sigma, (\uparrow, \neg S), m, r+1)$ or $q = q(\sigma, (\uparrow, S), m, r+1)$. In either case (C2a) holds, by definition of $\text{Pro}(r+1)$.

In order to show (C2b), let j be minimal such that B_j contains $(\sigma, (\uparrow, S))$ or $(\sigma, (\uparrow, \neg S))$. Therefore, the point q corresponding to B_j is a leftmost point above $r+1$ with label σ. Thus, $q = q(\sigma, (\uparrow, \neg S), \min, r)$ and there is some i with $s(a, i) = s(b, j)$ and q induces a pair $(\sigma, (\uparrow, \neg S))$ in A_i.

Finally, (C2c) is guaranteed by the definition of $\text{Pro}(r)$.

Altogether, we can conclude that $\text{PSeq}(\mathcal{P})$ is consistent.

This concludes the first part of the proof that is, if L has a finite solution then there is a finite consistent sequence of L-valid profiles.

Now, let $\mathcal{S} = P_1 \dots P_N$ be a consistent sequence of L-valid profiles. There are sets $A_{k,i}$ and numbers m_k such that for each $k \leq 1$, $P_k = A_{k,1}, \dots, A_{k,m_k}$.

We construct a finite set $\mathcal{P}$ of labeled points with $\text{PSeq}(\mathcal{P}) = \mathcal{S}$ and prove that $\mathcal{P}$ is a model of L. Intuitively, the idea is that for every occurrence of $(\sigma, \cdot)$ in profile P_i the ith line of $\mathcal{P}$ contains a point p labeled with σ. That is, the points in $\mathcal{P}$ only have y-values from $\{1, \dots, N\}$. During the construction of $\mathcal{P}$ we will use rational, possibly non-integer values for x-coordinates. We note that if there is a finite solution for L with rational x-values, then there is also a finite solution for L with natural x-values which can be obtained by replacing, for every k, the k-th smallest x-value by the number k. Clearly, this transformation does not affect the successor relation on the y-values.

More formally, we assign to every set of a profile a point in $\mathbb{Q} \times \mathbb{N}$. To this end, we define, for every $k \in \{1, \dots, n\}$, a function $\pi_k : \{1, \dots, m_k\} \to \mathbb{Q}$ with the interpretation that the point $(\pi_k(i), k)$ is assigned to the set $A_{k,i}$. Whenever a set $A_{k,i}$ contains a pair $(\sigma, \cdot)$ we add a point p to $\mathcal{P}$ with $p.y = k$, $p.x = \pi_k(i)$ and $p.l = \sigma$. Therefore, $\mathcal{P}$ is defined as soon as the functions π_k are defined.

The function π_1 for the first row is defined by $\pi_1(i) = i$, for every $i \leq m_1$. Let us now assume that the function $\pi_1, \dots, \pi_{k-1}$ are already defined. Since (P_{k-1}, P_k) is consistent there is a synchronizing function s for them. For every pair (i, j) such that $A_{k-1,i}$ and $A_{k,j}$ are synchronized by s we define $\pi_k(j) = \pi_{k-1}(i)$. Let j_0, j_1 be such that $\pi_k(j_0)$ and $\pi_k(j_1)$ are already defined by the previous step, and $\pi_k(j)$ is not defined for every j with $j_0 < j < j_1$. We define $\pi_k(j)$ for all j, $j_0 < j < j_1$ by picking values such that

- $\pi_k(j_0) < \pi_k(j) < \pi_k(j') < \pi_k(j_1)$, for every pair j, j' with $j_0 < j < j' < j_1$, and
- for every j, $j_0 < j < j_1$, $\pi_k(j)$ is not in the range of π_ℓ for any $\ell < k$.

As the interval $(\pi_k(j_0), \pi_k(j_1))$ contains infinitely many points from $\mathbb{Q}$ such values can be found. We note that this choice of x-values ensures that two points can be on the same vertical only if they are connected. This concludes the construction of $\mathcal{P}$.

It remains to prove that $\mathcal{P}$ is indeed a solution for L.

To this end, we first prove the following claim.

Claim 1. Let $k \leq N$ and $i \leq m_k$.

(a) There is a τ-labeled point $q \in \mathcal{P}$ with $q.x = \pi_k(i)$, $q.y > k+1$ and $q.l = \tau$ if and only if $(\tau, (\uparrow, \neg S) \in A_{k,i}$.

(b) There is a τ-labeled point $q \in \mathcal{P}$ with $q.x = \pi_k(i)$, $q.y = k+1$ and $q.l = \tau$ if and only if $(\tau, (\uparrow, S) \in A_{k,i}$.

Furthermore, the corresponding statement with $\downarrow$ in place of $\uparrow$ hold as well.

Proof (of Claim 1). Let us first assume that $(\tau, (\uparrow, \neg S)) \in A_{k,i}$. By consistency of the profiles and induction it follows that for some $\ell > k + 1$ and $j \le m_\ell$, $(\tau, \cdot) \in A_{\ell,j}$ and position j of P_ℓ is connected to position i of P_k. By definition of π and induction we can conclude that $\pi_k(i) = \pi_\ell(j)$ and therefore, by construction of $\mathcal{P}$ from π, such a point q indeed exists. If $(\tau, (\uparrow, S)) \in A_{k,i}$ an analogous argument yields $(\tau, \cdot) \in A_{k+1,j}$ and again a point q with the desired properties. Thus, we can conclude that the "if"-statements in (a) and (b) hold. The corresponding conclusion for $\downarrow$ in place of $\uparrow$ naturally also holds.

Let us now assume that q is as in the statement of the claim. By construction of $\mathcal{P}$ there must be ℓ and j such that $(\tau, \cdot) \in A_{\ell,j}$, $\ell > k + 1$, and $\pi_\ell(j) = \pi_k(i)$. By definition of π and induction it follows that position j of P_ℓ and position i of P_k are connected. By consistency condition (C1) and the definition of π we can conclude that $(\tau, (\uparrow, S)) \in A_{\ell-1,j'}$, for some $j' \le m_{\ell-1}$ with $\pi_{\ell-1}(j') = \pi_\ell(j)$. This yields the "only if"-part of (b). By consistency condition (C2), definition of π and induction we can conclude that for every $u < \ell - 1$ there is some $v \le m_u$ such that $(\tau, (\uparrow, \neg S)) \in A_{u,v}$ and $\pi_u(v) = \pi_\ell(j)$. Thus, in particular, either $k = \ell - 1$ and $(\tau, (\uparrow, S)) \in A_{k,i}$ or $k < \ell - 1$ and $(\tau, (\uparrow, \neg S)) \in A_{k,i}$. This yields the "only if"-part of (a). The corresponding conclusion for $\downarrow$ in place of $\uparrow$ holds again. This completes the proof of Claim 1. $\square$

Now we can show that $\mathcal{P}$ is a solution for L.

Let first (σ, E) be an $\exists$-constraint from L and $p \in \mathcal{P}$ with $p.l = \sigma$. Let k be the number of the corresponding profile and i the position of the corresponding set in P_k. By construction of $\mathcal{P}$, $(\sigma, \cdot) \in A_{k,i}$. As $\mathcal{S}$ is L-valid, there is some $(\tau, d) \in E$, some $j \le m_k$ and some $(\tau, c) \in A_{k,j}$ such that $(\sigma, \cdot)$ and (τ, c) are consistent with respect to d.

- Let us first assume that $d = (\nearrow, s_d)$, for some s_d. From statements (a) and (b) of Claim 1 it follows that there exists a τ-labeled point q corresponding to (τ, c) in $A_{k,j}$ such that (p, q) satisfies d. For directions $\nwarrow, \swarrow, \searrow$ we can argue analogously.
- Let us now assume that $d = (\rightarrow, \neg S)$. By validity, $c = \cdot$ and thus there is a τ-labeled point q with $q.y = k$ and $q.x = \pi_k(j) > p.x$. Likewise, for $\leftarrow$.
- Let finally $d = \ne$. From validity it follows that $i \ne j$ or $c \ne \cdot$. Similarly as in the two previous cases, we can conclude that there is a τ-labeled point q in the row of p or somewhere else, possibly on the same vertical line as p but different from p.

Let now (σ, τ, d) be a $\forall$-constraint. For the sake of a contradiction, let us assume that there are p and q that do *not* satisfy this constraint. By definition this means that $p.l = \sigma$, $q.l = \tau$ and (p, q) satisfies d. Let $A_{k,i}$ and $A_{\ell,j}$ be the sets from which p and q were obtained.

- Let us first assume that $d = (\nearrow, S)$. Thus, $\ell = k + 1$. As $(\tau, \cdot) \in A_{\ell,j}$ and $q.x > p.x$, (C1) implies that $(\tau, (\uparrow, S)) \in A_{k,j'}$ for some $j' > i$ contradicting validity. An analogous argument applies to the directions $\nwarrow, \swarrow, \searrow$ in place of $\nearrow$.
- Let us next assume that $d = (\nearrow, \neg S)$. Thus, $\ell > k + 1$. Using consistency it can be shown by induction that, for every $u < \ell - 1$ there is some v such that $(\tau, (\uparrow, \neg S)) \in A_{u,v}$ and $\pi_u(v) \ge \pi_\ell(j)$. In particular, there is some j' such that $(\tau, (\uparrow, \neg S)) \in A_{k,j'}$ and $\pi_k(j') \ge \pi_\ell(j) > \pi_k(i)$, contradicting validity. An analogous argument applies to the directions $\nwarrow, \swarrow, \searrow$ in place of $\nearrow$.
- Let us next assume that $d = (\rightarrow, \neg S)$. Thus, $\ell = k$ and $(\tau, \cdot) \in A_{k,j}$, an immediate contradiction to validity. Likewise for $\leftarrow$ in place of $\rightarrow$.
- Let us finally assume that $d = (\uparrow, s_d)$, for some s_d. By Claim 1 it follows that $(\tau, (\uparrow, s_d)) \in A_{k,i}$, an immediate contradiction to validity.

As all possible cases yield a contradiction we can conclude that violations of $\forall$-constraints do not occur. Altogether we have shown that $\mathcal{P}$ is indeed a solution for L. $\square$

It should be noted that the proof implicitly shows that if an 2LPP-problem has a finite solution then it has a solution with at most $2|\Sigma|$ points per horizontal line.

Now we are ready to complete the proof of Proposition 3.6.

Proof (of Proposition 3.6). Proposition 3.8 allows for testing satisfiability of a labeled point problem L by checking whether there is a consistent sequence of L-valid profiles. We note that there is only an exponential number M of profiles, thus if there is a such sequence of profiles, there is also a sequence of length $\leq M$.

Whether such a sequence exists can be tested by a non-deterministic algorithm with polynomial space. The algorithm guesses a sequence $P_1, \ldots, P_N$ of profiles, for some $N \leq M$. It checks that every profile is L-valid and that the sequence is consistent. To this end, it only needs to store two profiles P_k, P_{k+1} at any time. Clearly, L-validity of a given profile can be tested in polynomial time, likewise consistency of two profiles P_k, P_{k+1} and whether P_1 is a bottom profile and P_N a top profile.

By Savitch's Theorem the problem can therefore be solved in polynomial space. $\square$

The proof of Theorem 3.1 can now be given by a simple combination of the results in this section.

Proof (of the upper bound of Theorem 3.1). Let φ be a $\text{FO}^2(\leq_1, [\precsim_2, S_2])$-sentence. By Proposition 3.4, a semi-positive $\text{FO}^2(\mathcal{D}_-, I_y)$-sentence φ' that is equivalent to φ with respect to finite satisfiability can be computed in polynomial time. From φ' an equivalent exponential size labeled point problem L can be obtained, by Proposition 3.5. By Proposition 3.6, finite satisfiability of L can be tested in polynomial space. Hence, testing finite satisfiability of a $\text{FO}^2(\leq_{,1}, [\precsim_2, S_2])$-sentence can be done in exponential space. $\square$

Already in Section 2 we saw a strong connection between finite ordered structures and data words. Hence the following theorem follows straightforwardly from Theorem 3.1.

Theorem 3.9. Finite satisfiability for $\text{FO}^2(\Sigma, \leq_1 [\precsim_2, S_2])$ is in EXPSPACE.

4. Lower Bounds for Two-variable Logic on Ordered Structures

4.1. One Linear Order and one Total Preorder.
The following result shows that the upper bound of Theorem 3.1 is sharp. We recall that $\text{FO}^2(\leq_1, \precsim_2)$ is an abbreviation for $\text{FO}^2(\text{FinOrd}(\leq_1, \precsim_2))$.

Theorem 4.1. Finite satisfiability for $\text{FO}^2(\leq_1, \precsim_2)$ is EXPSPACE-hard.

Proof. In the following, we use the notation $[i, j]$ to denote the set of natural numbers between i and j.

The proof of the theorem is by reduction from the EXPSPACE-complete problem ExpCorridorTiling. An input to ExpCorridorTiling is a tuple $I = (T, H, V, \alpha, \omega, n)$, where
- T is the set of *allowed tiles*,
- $V, H \subseteq T^2$ are the sets of horizontal and vertical constraints,
- $\alpha, \omega \in T$ are tiles for the bottom row and the top row, respectively, and
- n is a natural number, given in binary.

A *valid tiling for I* is a mapping $\lambda : [1, n] \times [1, m] \to T$, for some $m \geq 2$ such that the following constraints are satisfied:

(1) the bottom row starts with α, that is, $\lambda(1, 1) = \alpha$;
(2) the top row ends with ω, that is, $\lambda(n, m) = \omega$;
(3) all vertical constraints are satisfied, that is, for every $i \leq n$ and every $j < m$, $(\lambda(i, j), \lambda(i, j+1)) \in V$; and,
(4) all horizontal constraints are satisfied, that is, for every $i < n$ and every $j \leq m$, $(\lambda(i, j), \lambda(i + 1, j)) \in H$.

Deciding, whether an instance I for ExpCorridorTiling has a valid tiling is EXPSPACE-complete (see, e.g., [Boa97]).

We first describe how tilings for I can be encoded as structures with a linear order $\leq_1$, a total preorder $\precsim_2$ and some unary relations. Then we describe how an $\mathrm{FO}^2(\leq_1, \precsim_2)$-sentence φ_I can be constructed in polynomial time from I that describes necessary conditions for structures that encode a valid tiling. Finally, we show that from a model of φ_I a valid tiling for I can be constructed, thus establishing that $I \mapsto \varphi_I$ is a reduction from ExpCorridorTiling to finite satisfiability of $\mathrm{FO}^2(\leq_1, \precsim_2)$.

Let $I = (T, H, V, \alpha, \omega, n)$ be an ExpCorridorTiling-instance and let $\lambda : [1, n] \times [1, m] \to T$ be a valid tiling for I. For simplicity we assume that $n = 2^k$, for some integer k. We define a structure $M(\lambda)$ as follows. The universe P of $M(\lambda)$ consists of points from $\mathbb{N} \times \mathbb{N}$. Each position (i, j) of the tiling is represented in $M(\lambda)$ by the two elements

- $p_-(i, j) = (2jn + i, j)$ and
- $p_+(i, j) = ((2j + 3)n + i, j)$.

It should be noted that $x \neq x'$ for any two elements $(x, y) \neq (x', y')$ from P. Therefore, we can define a linear order $\leq_1$ on P by

$$(x, y) \leq_1 (x', y') \ \Leftrightarrow_{\mathrm{def}} \ x \leq x'.$$

We define the total preorder $\precsim_2$ by

$$(x, y) \precsim_2 (x', y') \ \Leftrightarrow_{\mathrm{def}} \ y \leq y'.$$

Thus, the equivalence classes of $\precsim_2$ are just maximal sets of points of P on the same horizontal line.

Furthermore, $M(\lambda)$ has the set $\mathcal{C} \cup \mathcal{S} \cup \{R_-, R_+\}$ of unary relations, where

- the relations from $\mathcal{C} = \{C_1, \ldots, C_k\}$ encode the column number of elements via $C_p = \{(i, j) \mid \text{the } p\text{-th bit of the binary representation of } i \text{ is } 1\}$;
- the relations from $\mathcal{S} = \{S_t \mid t \in T\}$ encode the actual tiling via $S_t = \{p_-(i, j), p_+(i, j) \mid \lambda(i, j) = t\}$; and
- R_- contains all elements of the form $p_-(i, j)$ and R_+ all elements of the form $p_+(i, j)$. We call the former *negative* and the latter *positive* elements.

See Figure 3 for an outline of the construction.

Next, we describe the construction of φ_I. As we keep the intuition of the previous section that elements of a structure with a linear order $\leq_1$ and a total preorder $\precsim_2$ can be considered as points in the plane, we will refer to the elements of such a structure as *points* and to the equivalence classes with respect to $\sim_2$ as *horizontal lines* or simply *lines*[5].

Let $\varphi_{\mathrm{general}}$ be a formula that expresses that

- every point p is in exactly one of R_+ and R_-, and

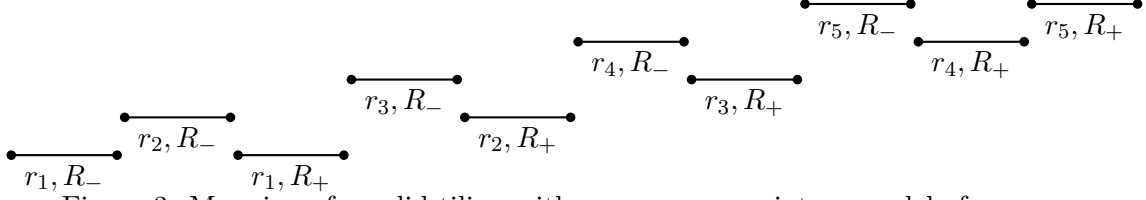[5] We avoid the term *row* for horizontal lines to avoid confusion with the rows of a tiling.

Figure 3: Mapping of a valid tiling with rows $r_1, \ldots, r_5$ into a model of φ.

- for every point p there is exactly one t such that $p \in S_t$. In the following, we refer to this t by $t(p)$.

In the following, we associate with every point p a column number $c(p)$ that is the number whose bit string has a 1 at position i if and only if $p \in C_i$ (where we count the least significant bit as position 1).

We use a formula $\psi_{+1}(x, y)$ to express that the column number of y is the column number of x plus one:

$$\psi_{+1}(x, y) = \bigvee_{i=1}^{k} \left(\neg C_i(x) \wedge C_i(y) \wedge \right.$$
$$\bigwedge_{j=1}^{i-1} C_j(x) \wedge \neg C_j(y) \wedge$$
$$\left. \bigwedge_{j=i+1}^{k} C_j(x) \leftrightarrow C_j(y) \right)$$

The next formula φ_{line} expresses that the points of every horizontal line are as intended. More precisely, φ_{line} expresses that for every point p

- there is a point $q \in R_-$ with column number 0 in the same line such that $q \leq_1 p$;
- there is a point $q \in R_+$ with column number 0 in the same line such that $(q \leq_1 p \iff p \in R_+)$;
- there is a point q in the same line with $c(q) = c(p)$, $t(q) = t(p)$ and $(p \in R_- \iff q \in R_+)$;
- unless $c(p) = n - 1$, there is a point q in the same line with $c(p) + 1 = c(q)$, $p <_1 q$, $(p \in R_- \iff q \in R_-)$ and $(t(p), t(q)) \in H$; and
- there is no point $q \neq p$ in the same line with the same column number and $(p \in R_- \iff q \in R_-)$.

The last formula $\varphi_{\text{next-line}}$ expresses that for every line l which is not a top line, there is another line l' such that the R_--part of l' is to the northwest of the R_+-part of l and such that the points in l and l' are compatible with respect to V. That is, $\varphi_{\text{next-line}}$ expresses that

- for every point $p \in R_+$ with $c(p) = 0$ there is a point q such that
  - q is in the same line as p, $c(q) = n - 1$ and $t(q) = \omega$, or
  - $q <_1 p$, $p \prec_2 q$ and $c(q) = n - 1$; and
- for every point $p \in R_+$ and every point $q \in R_-$ with $q <_1 p$, $p \prec_2 q$ and $c(q) = c(p)$ it holds $(t(p), t(q)) \in V$.

The formula φ_{start} expresses that the smallest point with respect to $\leq_1$ has the tile α.
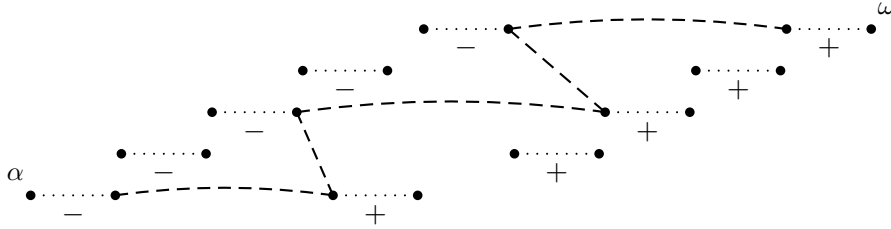
Figure 4: A sketch of the extraction of the rows of a valid tiling from a model of formula φ. The dotted lines indicate rows represented by (half) a line. The dashed lines indicate the extracted tiling.

Finally,

$$\varphi =_{\text{def}} \varphi_{\text{general}} \wedge \varphi_{\text{start}} \wedge \varphi_{\text{line}} \wedge \varphi_{\text{next-line}}.$$

It is easy to check that $M(\lambda) \models \varphi$ if λ is a valid tiling.

It remains to show that from every finite model M of φ we can construct a valid tiling λ_M for I. Let thus M be a finite model of φ. The fact that $M \models \varphi_{\text{general}}$ guarantees that every point represents exactly one tile and is in either R_- or R_+. From $M \models \varphi_{\text{line}}$ we can conclude that

- every class of $\sim_2$ consists of exactly $2n$ points, one for each possible combination of column number and being in R_- or R_+;
- points with adjacent column numbers have tiles that are compatible with respect to H; and
- every R_--point has the same tile as the R_+-point with the same column number.

Thus, in particular, every class of $\sim_2$ represents a horizontally valid row.

From $M \models \varphi_{\text{start}}$ it follows that there is a line that represents a valid bottom row of a tiling. We finally show that, for every line l of M it holds that l represents a valid top row of a tiling or there is another line l' above l that represents a tiling row that is consistent with the tiling row of l with respect to V.

It should be noted that we can not guarantee by an FO^2-formula that for every point p in R_+ and column number 0 there is *exactly* one (full or partial) line in northwestern direction. However, the first part of $\varphi_{\text{next-line}}$ guarantees that there is *at least* one line l' such that all R_--points of l' are in northwestern direction of p (and therefore in northwestern direction of all R_+-points in the line of p). Thus, the second part of $\varphi_{\text{next-line}}$ guarantees that the tiling row represented by l' is compatible with the row represented by p's line.

Therefore, M has a line encoding a valid bottom row of a tiling and every line either encodes a valid top row or has a line above that encodes a compatible row with respect to V. As M is finite, we can conclude that there exists a valid tiling for I. The extraction of a valid tiling from a model of φ is illustrated in Figure 4. $\qquad \square$

4.2. **Undecidable Extensions.** Next, we show that the approach of Theorem 3.1 fails if the linear order is replaced by a (second) total preorder. It can be concluded that Proposition 3.6 does not hold any more if we allow vertical directions in $\exists$-constraints of 2LPP.

Theorem 4.2. Finite satisfiability for $\text{FO}^2(\precsim_1, \precsim_2)$ is undecidable.

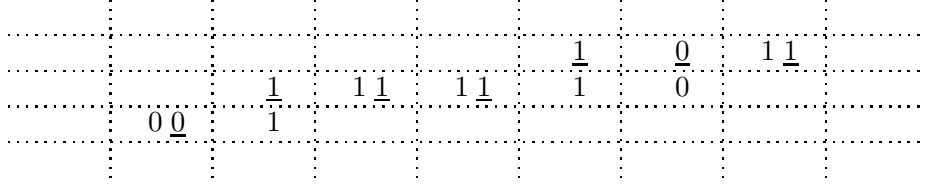|		1	1 1	1 1	1	0	1 1
		1	1 1	1 1	1	0	
	0 0	1					

Figure 5: How the valid sequences $u := 01|1110|1$ and $v := 0|111|101$ are represented in a model for the $\mathrm{FO}^2(\precsim_1, \precsim_2)$-formula φ. Columns represent equivalence classes of $\precsim_1$ and rows represent equivalence classes of $\precsim_2$. Letters from v are underlined.

Proof. Let $\Sigma = \{0,1\}$. We give a reduction from the Post Correspondence Problem PCP to finite satisfiability for $\mathrm{FO}^2(\precsim_1, \precsim_2)$, where PCP is defined as follows

> *Problem:* PCP
> *Input:* A sequence $(u_1, v_1), \ldots, (u_k, v_k)$, where every $u_i, v_i \in \{0,1\}^*$.
> *Question:* Is there a non-empty, finite sequence $\vec{i} = i_1, \ldots, i_m$ such that
> $u_{i_1} \ldots u_{i_m} = v_{i_1} \ldots v_{i_m}$?

Let $I = (u_1, v_1), \ldots, (u_k, v_k)$ be an instance of the PCP.

We first describe how solutions for I can be encoded as structures with two total preorders $\precsim_1, \precsim_2$ and some unary relations. Then we describe how an $\mathrm{FO}^2(\precsim_1, \precsim_2)$-sentence φ_I can be constructed from I that describes properties that a model that encodes a solution to I should fulfill. Finally, we show that from a model of φ_I a solution to I can be extracted, thus establishing that $I \mapsto \varphi_I$ is a reduction from PCP to finite satisfiability of $\mathrm{FO}^2(\precsim_1, \precsim_2)$-sentences.

For the first step, let $\vec{i} = i_1, \ldots, i_m$ be a solution to I with $u := u_{i_1} \ldots u_{i_m} = v_{i_1} \ldots v_{i_m} =: v$. The universe of the intended model consists of all positions of u and all positions of v and therefore has $|u| + |v| = 2|u|$ elements. We refer to the i-th position of u and v by $u[i]$ and $v[i]$, respectively.[6]

The equivalence classes of $\precsim_1$ are simply the sets $S_i = \{u[i], v[i]\}$, for $i \leq |u|$, and, for all $i, j \leq |u|$, if $i < j$ then $S_i \precsim_1 S_j$.

The total preorder $\precsim_2$ has one equivalence class C_j for every $j \leq m$ containing all positions corresponding to u_{i_j} in u and v_{i_j} in v. Again, if $i < j$ then $C_i \precsim_2 C_j$ for all $i, j \leq m$.

The construction of $\precsim_1$ and $\precsim_2$ is illustrated in Figure 5.

Furthermore, the intended model uses unary relation symbols from $\{A_0, A_1\} \cup W \cup P \cup \{U, V\}$, where

- $W = \{W_1, \ldots, W_k\}$, and
- $P = \{P_1, \ldots, P_\ell\}$, where ℓ is the length of the longest word in $\{u_1, \ldots, u_k, v_1, \ldots, v_k\}$.

The relation U consists of all positions of u and V consists of all positions of v. Every position with symbol 0 is in A_0 and every position with symbol 1 is in A_1. Furthermore, if in $u = u_{i_1} \ldots u_{i_m}$, position i of u corresponds to the p-th position of u_{i_k} and $i_k = j$ then $u[i]$ is put into the relations W_j and P_p. Likewise, if in $v = v_{i_1} \ldots v_{i_m}$, position i of v corresponds to the p-th position of v_{i_k} and $i_k = j$ then $v[i]$ is put into the relations W_j and P_p.

This completes the description of the construction of the intended model $M(\vec{i})$.

---

[6]It should be stressed that $u[i]$ does not denote a symbol but a position in a string.

Next, we give some necessary conditions that $M(\vec{i})$ has to fulfill.

It is easy to construct an FO^2-formula φ_{symbols} expressing that

- in every equivalence class of $\precsim_1$, there is exactly one element labeled from U and one element from V, and
- the elements of every equivalence class of $\precsim_1$ coincide on the predicates A_0 and A_1.

It should be noted here that in general it can not be expressed by any FO^2 formula that all classes of a given equivalence relation have size two. However, in our setting it is sufficient to express that no two different elements from U are $\sim_1$-equivalent and likewise for V.

It is also not hard to construct an FO^2-formula φ_{words} expressing that for every equivalence class E of $\precsim_2$, there is some j such that

- all elements of E are in W_j,
- for each $p \leq |u_j|$ there is exactly one element e in $E \cap P_p \cap U$ and e is in A_σ if and only if $u_j[p]$ is labeled with σ, and
- for each $p \leq |v_j|$ there is exactly one element e in $E \cap P_p \cap V$ and e is in A_σ if and only if $u_j[p]$ is labeled with σ.

Finally, an FO^2-formula φ_{order} can be constructed expressing that the two total preorders are consistent with each other, that is,

- for all elements $e_1, e_2 \in U$, if $e_1 \prec_2 e_2$, then $e_1 \prec_1 e_2$, and
- for all elements $e_1, e_2 \in U$, if $e_1 \prec_1 e_2$ then
  - either $e_1 \prec_2 e_2$
  - or $e_1 \sim_2 e_2$ and $e_1 \in P_k$, $e_2 \in P_\ell$, for some $k < \ell$.

Likewise for v-positions.

We let $\varphi_I = \varphi_{\text{symbols}} \wedge \varphi_{\text{words}} \wedge \varphi_{\text{order}}$. Clearly, if $\vec{i}$ is a solution for I, then $M(\vec{i}) \models \varphi_I$.

It remains to show that I has a solution if there is a structure M with $M \models \varphi_I$. To this end, let M be a model for φ_I. As $M \models \varphi_{\text{words}}$, we can associate with every equivalence class E of $\precsim_2$ an index $i(E)$ such that its positions are in $W_{i(E)}$. Thus, the linear order $E_1 \prec_2 \cdots \prec_2 E_m$ induces a sequence $\vec{i} = i_1, \ldots, i_m$, via $i_j := i(E_j)$. We show now that $\vec{i}$ is a solution to I.

Let $a \neq b$ two elements from U. Formula φ_{symbols} ensures that they are not in the same $\sim_1$-class. Thus, $\prec_1$ induces a linear order on the U-positions and therefore these positions naturally constitute a string u. Likewise, the V-positions constitute a string v. Formula φ_{order} ensures that $a \prec_1 b$ if $a \prec_2 b$ or if $a \sim_2 b$ and there are $k < l$ with $a \in P_k$ and $b \in P_l$. Thus, $u = u_{i_1} \cdots u_{i_m}$ and, likewise, $v = v_{i_1} \cdots v_{i_m}$. Finally, φ_{symbols} guarantees that $|u| = |v|$ and, for every i, $u[i]$ and $v[i]$ carry the same symbol. This completes the proof of the correctness of the reduction. Clearly, $I \mapsto \varphi_I$ can be computed (even in polynomial time). $\square$

Next we prove undecidability in case one total preorder is replaced by two linear orders in the previous theorem.

Theorem 4.3. Finite satisfiability for $\text{FO}^2(\leq_1, \leq_2, \precsim_3)$ is undecidable.

Proof. As in the previous proof we reduce from PCP. However, given a PCP-instance I we first translate it into a modified instance I' over the alphabet $\hat{\Sigma} = \{0, 1, 0', 1'\}$. The idea is to replace every symbol σ in I by $\sigma\sigma'$ in I', e.g., to transform the instance $(0, 01), (10, 1)$ into $(00', 00'11'), (11'00', 00')$. More formally, $I' = h(I)$ where h is the homomorphism defined by $h(0) = 00'$ and $h(1) = 11'$.
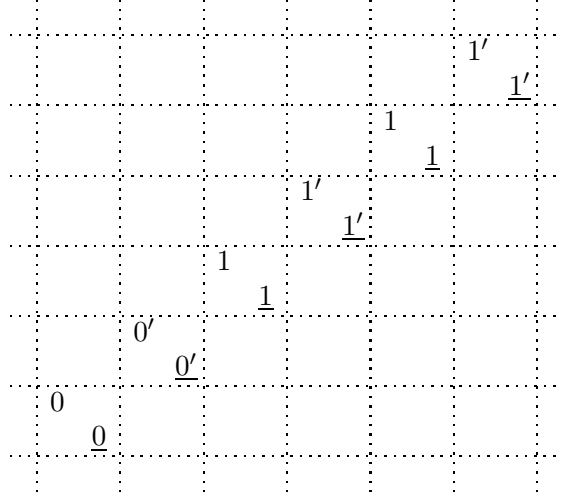
Figure 6: How the intended linear orders $\leq_1, \leq_2$ for a valid sequences $u := 00'|11'11'$ and $v := 00'11'|11'$ look like. Columns are ordered by $<_1$ and rows are ordered by $<_2$. Letters from v are underlined.

We call letters of the form σ' *marked*. Clearly, I' has a solution if and only if I has a solution.

From I' we will construct a $\mathrm{FO}^2(\leq_1, \leq_2, \precsim_3)$-formula φ_I with the intention that φ_I has a finite model if and only if I has a solution. The construction of φ_I is similar as in the proof of Theorem 4.2. However, the role of $\precsim_1$ in that proof will be mimicked by $\leq_1$ and $\leq_2$, here.

As before, we first describe how solutions for I' can be encoded as structures with two linear orders $\leq_1, \leq_2$, a total preorder $\precsim_3$, and some unary relations. Then we define φ_I and show that $I \mapsto \varphi_I$ is a reduction from PCP to finite satisfiability of $\mathrm{FO}^2(\leq_1, \leq_2, \precsim_3)$.

For the first step, let $\vec{i} = i_1, \ldots, i_m$ be a solution to I with $u := u_{i_1} \ldots u_{i_m} = v_{i_1} \ldots v_{i_m} =: v$. As before the intended model $M(\vec{i})$ consists of all positions of u and all positions of v and therefore has $2|u|$ elements.

The total preorder $\precsim_3$ is defined exactly as $\precsim_2$ in the previous proof. The model has additional unary relations $\{A_0, A_1, M\} \cup W \cup P \cup \{U, V\}$ where U, V and the relations from $W \cup P$ are defined as before. Positions with symbol $\sigma \in \{0, 1\}$ are in A_σ, positions with symbol $\sigma' \in \{0', 1'\}$ are in A_σ and in M.

The linear orders $<_1$ and $<_2$ are defined by

$$u[1] <_1 v[1] <_1 u[2] <_1 v[2] <_1 \ldots <_1 u[|u|] <_1 v[|u|]$$

and

$$v[1] <_2 u[1] <_2 v[2] <_2 u[2] <_2 \ldots <_2 v[|u|] <_2 u[|u|]$$

See 6 for an illustration of the construction of $\leq_1$ and $\leq_2$.

Now we describe necessary conditions that $M(\vec{i})$ has to fulfill.

First of all, it should satisfy formula φ_{words} of the previous proof, referring to $\precsim_3$ in place of $\precsim_2$.

Let $\varphi_3(x, y)$ be the formula which expresses that

- either $x \prec_3 y$
- or $x \sim_3 y$ and $x \in P_i$ and $y \in P_j$, for some $i < j$.

It is not hard to see that in every model of φ_{words} the formula φ_3 induces a linear order on the positions in U and on the positions in V. Furthermore, φ_{words} guarantees that both orders begin with unmarked positions, alternate between marked and unmarked positions and end with a marked position (just because the strings u_i, v_i do so). In the following, we write $x <_3 y$ as an abbreviation for $\varphi_3(x, y)$.

Using φ_3, we can define a formula $\varphi_{\text{2-order}}$ expressing that $\leq_1$, $\leq_2$ and $<_3$ are consistent on positions in U and on positions in V. That is, if $x, y \in U$ then $x <_1 y$ if and only if $x <_2 y$ if and only if $x <_3 y$ and likewise for $x, y \in V$. In the following, we denote these two orders on the positions of U and V simply by $\leq$.

Finally, we construct a formula φ_{symbols} with a similar intention as in the proof of Theorem 4.3.

To this end, let $\varphi_{\text{bi}}(x, y)$ be a formula expressing that $x \leq_1 y$ and $y \leq_2 x$.

The formula φ_{symbols} expresses that the following conditions hold.

- $\varphi_{\text{bi}}(x, y)$ only holds for positions $x \neq y$ if
  - $x \in U$ and $y \in V$, and
  - both carry the same symbol from $\hat{\Sigma}$.
- For every position x, there is a position y such that $\varphi_{\text{bi}}(x, y)$ holds and vice versa.

We let $\varphi_I = \varphi_{\text{symbols}} \wedge \varphi_{\text{words}} \wedge \varphi_{\text{2-order}}$ and it is again easy to see that, if $\vec{i}$ is a solution for I, then $M(\vec{i}) \models \varphi_I$.

It remains to show that I has a solution if there is a structure M with $M \models \varphi_I$. To this end, let M be a model for φ_I. As $M \models \varphi_{\text{words}}$, we can associate with every equivalence class E of $\precsim_3$ an index $i(E)$ such that its positions are in $W_{i(E)}$, just as in the previous proof. Again, the linear order $E_1 \prec_3 \cdots \prec_3 E_m$ induces a sequence $\vec{i} = i_1, \ldots, i_m$, via $i_j := i(E_j)$. We show now that $\vec{i}$ is a solution to I.

For this purpose, it is sufficient to show that

(1) $\varphi_{\text{bi}}(x, y)$ defines a bijection between the positions in U and the positions in V, and
(2) that this bijection is compatible with $\leq$, that is, if $\varphi_{\text{bi}}(a_1, b_1)$ and $\varphi_{\text{bi}}(a_2, b_2)$ hold for $a_1 \neq a_2 \in U$ and $b_1 \neq b_2 \in V$ then $a_1 < a_2$ if and only if $b_1 < b_2$.

Indeed, from (1) and (2) it follows that the bijection induced by φ_{bi} pairs the i-th position of U with the i-th position of V, for every i. As φ_{symbols} guarantees that corresponding positions carry the same symbol, it follows that $\vec{i}$ is a solution.

We next show (1), that is, there are no positions $a_1 \neq a_2 \in U$ and $b \in V$ such that $\varphi_{\text{bi}}(a_1, b)$ and $\varphi_{\text{bi}}(a_2, b)$ hold (and, correspondingly, for a, b_1, b_2). Towards a contradiction let us assume the existence of such elements $a_1 \neq a_2 \in U$ and $b \in V$ for which $\varphi_{\text{bi}}(a_1, b)$ and $\varphi_{\text{bi}}(a_2, b)$ hold. Let, without loss of generality $a_1 < a_2$ and let us assume that both are marked.[7] Then there is an unmarked U-position a with $a_1 < a < a_2$. But then we can conclude that $a \leq_1 b$ and $b \leq_2 a$, a contradiction as φ_{symbols} holds in M and a is unmarked and b is marked.

To show (2) let us assume, again towards a contradiction, that there are $a_1 \neq a_2 \in U$ and $b_1 \neq b_2 \in V$ such that $\varphi_{\text{bi}}(a_1, b_1)$ and $\varphi_{\text{bi}}(a_2, b_2)$, $a_1 < a_2$ and $b_2 < b_1$. But then

- $a_1 <_1 a_2 \leq_1 b_2$ and
- $b_2 <_2 b_1 \leq_2 a_1$

and therefore $\varphi_{\text{bi}}(a_1, b_2)$ holds, contradicting (1). Similarly, from $a_1 > a_2$ and $b_2 > b_1$ it follows that $\varphi_{\text{bi}}(a_2, b_1)$ holds, again in contradiction to (1). $\qquad\square$

---

[7]We recall that φ_{symbols} guarantees that a_1, a_2 and b are either all marked or all unmarked.

Data \ Positions	Successor	Linear Order	Successor & Linear Order
Successor	?	in EXPSPACE (this work)	undecidable [MZ11]
Linear Order	?	in EXPSPACE (this work)	undecidable [BMS$^+$06]
Successor & LO	?	in EXPSPACE (this work)	undecidable [BMS$^+$06]
single-occ. Succ	in 2NEXPTIME [Man10]	in EXPSPACE (this work)	decidable [MZ11]
single-occ. LO	in EXPSPACE (this work)	in EXPSPACE (this work)	in EXPSPACE (this work)

Table 1: Summary of results on Finite Satisfiability of FO^2 on data words with ordered data values. "single-occ." refers to the case where each data values occurs at most once in a data word.

5. Conclusion

The context of our results was already discussed in the introduction. Table 1 summarizes the results of this paper and previous results for data words with different kinds of orders on positions and data values.

We mention some possible lines for extensions and further research. We recall that FO$^2(\leq_1, \precsim_2)$ is an abbreviation for FO$^2(\text{FinOrd}(\leq_1, \precsim_2))$.

Two-Variable Logic. The lower bound for finite satisfiability of FO$^2(\leq_1; \precsim_2)$ from Theorem 4.1 does not immediately carry over to FO$^2(\Sigma, \leq_1, [\precsim_2, S_2])$ as the translation of a FO$^2(\leq ; \precsim)$ formula φ into a FO$^2(\Sigma, \leq, \precsim)$ formula might yield an alphabet of exponential size in $|\varphi|$. Thus, there remains a gap between the EXPSPACE upper bound from Corollary 3.2 and the NEXPTIME lower bound from [BMS$^+$06]. Further there is still a gap between the "eight orders" undecidability result of [Ott01] and the decidability for FO^2 with two linear orders in this paper.

In the context of automated verification it would be interesting to generalize our results from data words to data ω-words.

Other Logics. There are connections between the results of this paper and some temporal logics, Compass Logic and Interval Temporal Logic. Some of these connections have been made precise in the conference paper underlying this article [SZ10].

Compass Logic is a two-dimensional temporal logic, whose operators allow for moving north, south, east and west along a grid [Ven90]. Satisfiability for compass logic is known to be undecidable [MR97]. Compass logic can be extended in two directions. Up to now, only complete grids have been considered as underlying structures. Partial grids, i.e. grids where not all crossings need to exist, can be considered as underlying structures as well. Furthermore, operators northeast, northwest, southeast and southwest can be considered. The results of this article can be used, after some appropriate modifications, to yield decidability for compass logic with these extensions when only the operators northwest, northeast, southwest, southeast, west and east are used.

Interval Temporal Logic can reason about intervals of time with the help of operators as 'after', 'during', 'begins' etc. Expressions such as 'Immediately after we finished writing the paper, we will go to the beach' can be captured. To this end, propositions, like "writing the paper" or "go to the beach", are assigned to time intervals. In conventional interval temporal logic, all possible intervals are considered as part of a structure, that is, reasoning is always with respect to all intervals. In a setting where structures may consist of a subset of the set of all intervals, decidability of reasoning with the operators 'ends', 'later' and 'during' as well as their duals can be obtained from our results, again after appropriate modifications.

We believe that partial models for compass and interval logic deserve some further investigations.

Besides relations to compass logic and interval logic we conjecture that there are connections to spatial reasoning that is done in the context of Geographical Information Systems (for a survey, see [CH01]).

References

[BMS⁺06] Mikolaj Bojanczyk, Anca Muscholl, Thomas Schwentick, Luc Segoufin, and Claire David, *Two-variable logic on words with data*, LICS, 2006, pp. 7–16.

[Boa97] Peter Van Emde Boas, *The convenience of tilings*, In Complexity, Logic, and Recursion Theory, Marcel Dekker Inc, 1997, pp. 331–363.

[CH01] Anthony G. Cohn and Shyamanta M. Hazarika, *Qualitative spatial representation and reasoning: An overview*, Fundam. Inform. **46** (2001), no. 1-2, 1–29.

[DL09] Stéphane Demri and Ranko Lazic, *LTL with the freeze quantifier and register automata*, ACM Trans. Comput. Log. **10** (2009), no. 3, 16:1–16:30.

[FS09] Diego Figueira and Luc Segoufin, *Future-looking logics on data words and trees*, MFCS, 2009, pp. 331–343.

[GKV97] Erich Grädel, Phokion G. Kolaitis, and Moshe Y. Vardi, *On the decision problem for two-variable first-order logic*, Bulletin of Symbolic Logic **3** (1997), no. 1, 53–69.

[GO99] Erich Grädel and Martin Otto, *On logics with two variables*, Theor. Comput. Sci. **224** (1999), no. 1-2, 73–113.

[KO05] Emanuel Kieronski and Martin Otto, *Small substructures and decidability issues for first-order logic with two variables*, LICS, 2005, pp. 448–457.

[KT09] Emanuel Kieronski and Lidia Tendera, *On finite satisfiability of two-variable first-order logic with equivalence relations*, LICS, 2009, pp. 123–132.

[Man10] Amaldev Manuel, *Two variables and two successors*, MFCS, 2010, pp. 513–524.

[Mor75] M. Mortimer, *On languages with two variables*, Zeitschr. f. math. Logik u. Grundlagen d. Math. **21** (1975), 135–140.

[MR97] Maarten Marx and Mark Reynolds, *Undecidability of compass logic*, Journal of Logic and Computation **9** (1997), 897–941.

[MZ11] Amaldev Manuel and Thomas Zeume, *Personal communication.*, 2011.

[Ott01] Martin Otto, *Two variable first-order logic over ordered domains*, J. Symb. Log. **66** (2001), no. 2, 685–702.

[Seg06] Luc Segoufin, *Automata and logics for words and trees over an infinite alphabet*, CSL, 2006, pp. 41–57.

[SZ10] Thomas Schwentick and Thomas Zeume, *Two-variable logic with two order relations - (extended abstract)*, CSL, 2010, pp. 499–513.

[Ven90] Yde Venema, *Expressiveness and completeness of an interval tense logic*, Notre Dame Journal of Formal Logic **31** (1990), no. 4, 529–547.