A Short Note on Two-Variable Logic with a Linear Order Successor and a Preorder Successor

Amaldev Manuel
LIAFA, Université Paris Diderot
amal@liafa.univ-paris-diderot.fr

Thomas Schwentick
TU Dortmund University
thomas.schwentick@cs.uni-dortmund.de

Thomas Zeume
TU Dortmund University
thomas.zeume@cs.uni-dortmund.de

Saturday 27th September, 2014

Abstract

The finite satisfiability problem of two-variable logic extended by a linear order successor and a preorder successor is shown to be undecidable.

The decidability of the finite satisfiability problem of two-variable logic extended by orders and preorders as well as their corresponding successor relations has been currently investigated by several papers [2, 5, 3]. This short note extends [3] which is work under submission. In the setting of [3] only two cases remained open. One of them, namely the extension of two-variable logic by one successor relation of a linear order and one successor relation of a preorder, is settled here. We refer to [2] and [5] for more background and motivation.

In the rest of this note we first introduce the necessary notation and then prove that the finite satisfiability problem of the extension of two-variable logic by one successor relation of a linear order and one successor relation of a preorder is undecidable.

1 Notations

A binary relation $\leq_p$ over a finite set[1] A is a *preorder* if it is reflexive, transitive and total, that is, if for all elements u,v and w from A (i) $u \leq_p u$ (ii) $u \leq_p v$ and $v \leq_p w$ implies $u \leq_p w$ and (iii) $u \leq_p v$ or $v \leq_p u$ holds. A *linear order* $\leq_l$ on A is an antisymmetric total preorder, that is, if $u \leq_l v$ and $v \leq_l u$ then $u = v$.

Thus, the essential difference between a total preorder and a linear order is that the former allows for two distinct elements u and v that both $u \leq_p v$ and $v \leq_p u$ hold. We call two such elements *equivalent with respect to* $\leq_p$ and denote this by $u \sim_p v$. Hence, a total preorder can be seen as an equivalence relation $\sim_p$ whose equivalence classes are linearly ordered by a linear order. Clearly, every linear order is a total preorder with equivalence classes of size one. We write $u <_l v$ if $u \leq_l v$ but not $v \leq_l u$, analogously for a preorder order $\leq_p$. Further, if C and C' are the equivalence classes of u and v, respectively, then we write $C \leq_p C'$ if $u \leq_p v$.

For a linear order $\leq_l$ an induced *successor relation* $+1_l$ can be defined in the usual way, namely by letting $+1_l(u, v)$ if and only if $u <_l v$ and there is no w with $u <_l w <_l v$. Similarly a preorder $\leq_p$ induces a successor relation $+1_p$ based on the linear order on its equivalence classes,

[1] In this note all sets are finite.

i.e. $+1_p(u,v)$ if and only if $u <_p v$ and there is no w with $u <_p w <_p v$. Thus an element can have several successor elements in $+1_p$.

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. Linear orders and their induced successor relations will be denoted by $\leq_l, \leq_{l_1}, \leq_{l_2}, \dots$ and $+1_l, +1_{l_1}, +1_{l_2}, \dots$, respectively. Analogously, preorders and their induced successor relations will be denoted by $\leq_p, \leq_{p_1}, \leq_{p_2}, \dots$ and $+1_p, +1_{p_1}, +1_{p_2}, \dots$, respectively. For a set of binary relation symbols O, an O-structure is a finite structure with some unary relations and some binary relations of types indicated by O. For example an $(+1_l, +1_p)$-structure is a structure with some unary relations and, following the conventions from above, a linear order successor and a preorder successor.

Two-variable logic FO^2 is the restriction of first-order logic to formulas with at most two distinct variable x and y. By $\mathrm{FO}^2(O)$ we denote two-variable logic over a vocabulary that contains some unary relation symbols and binary relation symbols from O which have to be interpreted by O-structures. For example, formulas in $\mathrm{FO}^2(+1_l, +1_p)$ can use some unary relation symbols and the binary relation symbols $+1_l$ and $+1_p$, where $+1_l$ and $+1_p$ have to be interpreted as a linear successor and a preorder successor.

2 Result

In this section we prove the only result of this note.

Theorem 1. *Finite satisfiability of* $\mathrm{FO}^2(+1_l, +1_p)$ *is undecidable.*

The proof is by a reduction from the non-emptiness problem for Minsky counter automata. Following [1], a *Minsky counter automaton* (short: CA) is essentially a finite state automaton without input but equipped with a finite set of counters which can be incremented, decremented and tested for zero. More formally, a CA $\mathcal{M}$ is a tuple (Q, C, Δ, q_I, F), where the set Q of states, the initial state $q_I \in Q$ and the set $F \subseteq Q$ of final states are as in usual finite state automata, and C is a finite set (the *counters*). The transition relation Δ is a subset of

$$Q \times \{\mathrm{inc}(c), \mathrm{dec}(c), \mathrm{ifzero}(c) | c \in C\} \times Q$$

A *configuration* of a CA is a pair $(p, \vec{n})$ where p is a state and $\vec{n} \in \mathbb{N}^C$ gives a value n_c for each counter c in C. Transitions with $\mathrm{inc}(c)$ can be always applied, whereas transitions with $\mathrm{dec}(c)$ can only be applied to configurations with $n_c > 0$ and transitions with $\mathrm{ifzero}(c)$ can only be applied if $n_c = 0$. Applying a transition $(p, \mathrm{inc}(c), q)$ to a configuration $(p, \vec{n})$ yields a configuration $(q, \vec{n}')$ where $\vec{n}'$ is obtained from $\vec{n}$ by incrementing n_c and keeping all other values unchanged. Analogously, applying a (applicable) transition $(p, \mathrm{dec}(c), q)$ to a configuration $(p, \vec{n})$ yields a configuration $(q, \vec{n}')$ where $\vec{n}'$ is obtained from $\vec{n}$ by decrementing n_c. Applying an (applicable) transition $(p, \mathrm{ifzero}(c), q)$ to a configuration $(p, \vec{n})$ yields configuration $(q, \vec{n})$. A *run* is a sequence of configurations consistent with Δ. A run is *accepting*, if it starts at configuration $(q_I, \vec{0})$ and ends in some configuration $(q_F, \vec{n})$ with $q_F \in F$. Without loss of generality we require that the counter values in the last configuration of an accepting run are equal to zero.

The emptiness problem for CA is the question whether a given CA has an accepting run. It is well known that the emptiness problem for CA with two counters is undecidable [4].

Proof (of Theorem 1). We use the following notions. An element v is a $(+1_l, +1_p)$-*successor* of an element u if $+1_l(u,v)$ and $+1_p(u,v)$. Observe that there does not need to be a $(+1_l, +1_p)$-successor and that it is unique if it exists. Similarly v is a $(+1_l, +1_p)$-*predecessor* of u if $+1_l(v,u)$ and $+1_p(v,u)$.

The proof is by a reduction from the non-emptiness problem for Minsky counter automata with two counters only.

Intuitively, from a given CA $\mathcal{M}$ with set $C = \{R, B\}$ of counters (we refer to R and B as the red and blue counter in the following), we construct a $\mathrm{FO}^2(+1_l, +1_p)$-formula $\varphi_{\mathcal{M}}$ such that for every accepting run ρ of $\mathcal{M}$ there is a $(+1_l, +1_p)$-structure $\mathcal{A}_\rho$ that satisfies $\varphi_{\mathcal{M}}$, and such that from every model of $\varphi_{\mathcal{M}}$ an accepting run of $\mathcal{M}$ can be constructed. The formula $\varphi_{\mathcal{M}}$ uses as propositions the set Δ of transitions of $\mathcal{M}$, the propositions $\{R, B\}$. We will often say color instead of proposition.

An accepting run $\rho = \rho_0, \dots, \rho_n$ of $\mathcal{M}$ will be encoded as a $(+1_l, +1_p)$-structure $\mathcal{A}_\rho$ as follows. The idea is to represent configurations by equivalence classes with respect to $\sim_p$, to encode transitions by propositions and counter values by the number of red and blue elements in a class.

We will always assume in the following that $\mathcal{M}$ is non-trivial and thus ρ has at least one step. More precisely, let, for every $i \in \mathbb{N}_0$, $\rho_i = (p_i, (b_i, r_i))$ with $p_i \in Q$ and $b_i, r_i \in \mathbb{N}$ and δ_i be the transition applied in the i-th step. Let k be large enough such that for every configuration the value of the red counter plus the value of the blue counter is at most k. The intended structure $\mathcal{A}_\rho$ has exactly one $+1_p$-class τ_i for each configuration ρ_i. For every i, τ_i has exactly k elements of which b_i elements carry the proposition B and r_i elements carry the proposition R. The sets of B-elements and of R-elements are disjoint. The relation $+1_l$ induces bijections between successive $+1_p$-classes with the following additional properties. As the counter values of $\mathcal{M}$ can change by at most one in one step, the number of blue (or red) elements in successive classes at most differs by one. The relation $+1_l$ can thus be chosen such that

- it is a bijection between the blue elements in τ_i and τ_{i-1}, if δ_i does not change the blue counter,
- it is a bijection between the blue elements of τ_i and the blue elements of τ_{i+1} minus one, in case δ_i increments the blue counter, and
- it is a bijection between the blue elements of τ_i minus one and the blue elements of τ_{i+1}, in case δ_i decrements the blue counter

Likewise for the red counter and red elements. Furthermore, each element of τ_i carries the transition δ_i that yielded it from the previous configuration (with the exception of τ_0).

We now state conditions (T1-T5), (B1-B5) and (R1-R5) that hold in a structure $\mathcal{A}$ if and only if it is (isomorphic to) a structure $\mathcal{A}_\rho$, for some accepting run ρ. The conditions (T1-T5) ensure the general structure of $\mathcal{A}$, the consistency of successive transitions and the initial and final state. Conditions (B1-B5) ensure that the counter values, i.e. the number of B-labeled elements, are consistent with the transitions. Likewise for (R1-R5).

(T1) Every element that is not in the first $+1_p$-class, carries exactly one label from Δ, and all elements of a $+1_p$-class carry the same Δ-label.

(T2) Elements of the second $+1_p$-class are labeled with a transition starting from the start state.

(T3) Elements of the last $+1_p$-class are labeled with a transition leading to a final state.

(T4) Transitions of successive $+1_p$-classes are consistent, i.e. if u, v are labeled with (p, op, q) and (p', op', q'), and $+1_p(u, v)$, then $q = p'$.

(T5) Every element that is not in the last $+1_p$-class has a $(+1_l, +1_p)$-successor and every element that is not in the first class has a $(+1_l, +1_p)$-predecessor.

(B1) Neither there are B-labeled elements in the first $+1_p$-class nor in the last $+1_p$-class.

(B2) If some element u carries a transition from Δ that does not change the B-counter then $+1_l$ induces a bijection between the B-labeled elements of the class of u and the B-labeled elements of its $+1_p$-predecessor class.

(B3) If some element u carries a transition from Δ that increments the B-counter, then

- there is exactly one B-labeled element v with $v \sim_p u$ whose $+1_l$-predecessor is not B-labeled and
- $+1_l$ induces a bijection between the set of all other B-labeled elements of the class of u and the B-labeled elements of its $+1_p$-predecessor class.

(B4) If some element u carries a transition from Δ that decrements the B-counter, then

- there is exactly one B-labeled element v with $+1_p(v,u)$ whose $+1_l$-successor is not B-labeled and
- $+1_l$ induces a bijection between the set of all other B-labeled elements of the $+1_p$ predecessor class of u and the B-labeled elements of u's class.

(B5) If some element u carries a transition that tests whether the B-counter is equal to zero, then there is no B-labeled element v with $+1_p(v,u)$.

(R1)-(R5) are as (B1)-(B5) with "blue" replaced by "red", throughout.

For proving the correctness of this construction, we first observe that the $(+1_l, +1_p)$-structure $\mathcal{A}_\rho$ constructed from a given accepting run ρ as explained above, clearly satisfies all these conditions.

Now, let $\mathcal{A}$ be a structure that satifies all conditions. Let $\tau_0, \dots, \tau_n$ be the $+1_p$-classes of $\mathcal{A}$ in increasing order. By property (T1) there is, for every $i > 0$, a single transition $\delta_i = (p_i, op_i, q_i)$ that labels all elements from τ_i. Let, for every i, b_i and r_i be the number of B- and R-labeled elements in τ_i. We claim that $\rho = \rho_0, \dots, \rho_n$ with $\rho_0 = (p_0, (b_0, r_0))$ and, for every $i > 0$, $\rho_i = (q_i, (b_i, r_i))$, is an accepting run of M. The states q_i and p_{i+1} are consistent with δ_i for all $i \in \{1, \dots, m-1\}$; and p_0 is an initial state and q_m is a final state, by properties (T2)-(T4).

Thus it remains to verify that the counter values in ρ are consistent with the transitions. By (B1), both counters are zero in ρ_0. By induction over $i \in \{1, \dots, n\}$, it can be shown that ρ_i is consistent with ρ_{i-1} with respect to the counter values. Condition (B2) ensures that $b_i = b_{i-1}$ if δ_i does not change the blue counter. Likewise, (B3) and (B4), respectively, ensure that $b_i = b_{i-1} + 1$ if δ_i increments the counter and $b_i = b_{i-1} - 1$ if δ_i decrements the counter. Finally, (B5) ensures that zero test transitions for the blue counter are only taken if $b_{i-1} = 0$. The correctness of the values of the red counter can be proved analogously.

It remains to show that (T1-T5), (B1-B5) and (R1-R5) can be expressed by $\mathrm{FO}^2(+1_l, +1_p)$ formulas.

Conditions (T2)-(T5), (B1), (B2) and (B5) can be easily expressed by such formulas.

We describe next how to construct a formula $\varphi(x)$ that expresses that in the $+1_p$-class of x some element exists for which some unary formula ψ holds. The idea is simply to state that for some $+1_p$-predecesssor of a $+1_p$-successor of x it holds ψ or for some $+1_p$-successsor of a $+1_p$-predeccessor of x it holds ψ. Here, we make use of our assumption that $\mathcal{A}$ has at least two $+1_p$-classes. That is, we define $\varphi_{\text{succ}}(x)$ as $\exists y \, (+1_p(x,y) \wedge \exists x(+1_p(x,y) \wedge \psi(x)))$. It should be stressed that the quantification might bind x to some other element in the class of the "original" x. Likewise, we let $\varphi_{\text{precc}}(x)$ be $\exists y \, (+1_p(y,x) \wedge \exists x(+1_p(y,x) \wedge \psi(x)))$ and $\varphi(x) = \varphi_{\text{prec}} \vee \varphi_{\text{succ}}$.

In this way, a formula for (T1) is readily definable. For the second part of (T1) it is stated that there is no element in the current class with some other transition proposition.

For the construction of the other formulas, we assume that (T1) and (T5) hold.

The challenge when expressing (B3) is to make sure that there is exactly one B-labeled element in the current class that does not take part in the bijection induced by $+1_l$. However, this can be expressed by stating that

(1) every B-labeled $+1_p$-predecessor of the current element has a B-labeled $+1_l$-successor, and

(2) there is a $+1_p$-predecessor of the current element u that is not B-labeled but has a B-labeled $+1_l$-successor, but

(3) all B-labeled $+1_p$-successors of u besides u's $+1_l$-successor have a B-labeled $+1_l$-predecessor.

This condition can be easily expressed by a $\mathrm{FO}^2(+1_l, +1_p)$ formula. Condition (B4) can be expressed analogously. □

We strongly conjecture that finite satisfiability for the other remaining open case from [3], namely the extension of two variable logic by one preorder relation and one successor relation of a linear order, is decidable. We are actually working on the details of the proof. However, we felt that the result presented in this note should be made public without further delay.

References

[1] Mikoaj Bojańczyk, Anca Muscholl, Thomas Schwentick, and Luc Segoufin. Two-variable logic on data trees and XML reasoning. *J. ACM*, 56(3):1–48, 2009.

[2] Amaldev Manuel. Two orders and two variables. In *MFCS*, volume 6281 of *Lecture Notes in Computer Science*, pages 513–524, 2010.

[3] Amaldev Manuel and Thomas Zeume. Two-variable logic on 2-dimensional structures. *Submitted to CSL*, 2013.

[4] Marvin L. Minsky. *Computation: finite and infinite machines.* Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1967.

[5] Thomas Schwentick and Thomas Zeume. Two-variable logic with two order relations. *Logical Methods in Computer Science*, 8(1), 2012.