

# Counter Automata and Classical Logics for Data Words

Amaldev Manuel

Let  $\Sigma$  be a finite alphabet and  $\Gamma$  be an infinite set in which membership and equality are decidable. We call finite sequences of elements of the set  $\Sigma \times \Gamma$  *data words*. Formally a data word  $w$  is in  $(\Sigma \times \Gamma)^*$ .

The course of study of data languages so far has been driven by two important questions, which are (1) what is a suitable class of automata for recognizing data languages? (2) what is a suitable logical language for expressing data languages? The contributions of this thesis are to be seen in the light of these two questions which we discuss briefly below.

Our approach to the automaton problem involves enhancing finite state automata with counters. Counters are a primitive and minimal computational device where the operations are increment, decrement and checking for zero. Yet it is long been known that automata with two counters are as powerful as Turing machines. Hence it is necessary to restrict the operations on the counters. There are standard restrictions in the literature. Some of them are (1) disallowing the decrement operation (2) removing the two-way branching on a zero test (3) allowing counter values to be negative etc.

We now briefly describe the class of machines we call Class Counting Automata. A class counting automaton  $A = (Q, \Sigma, \Delta, I, F)$  is a finite state automaton with  $|\Gamma|$ -many counters where  $Q$  is the finite set of states,  $\Delta$  is the transition function and  $I \subseteq Q$  and  $F \subseteq Q$  are the set of initial and final set of states. A configuration of the automaton is of the form  $(q, h)$  where  $q \in Q$  and  $h : \Gamma \rightarrow \mathbb{N}$  is a function holding the counter values. The transition of the automaton are of the form  $(p, a, \varphi(x), u, q)$  where  $p, q$  are the entry and exit states of the transition,  $\varphi(x)$  is a univariate linear inequality and  $u$  is from the set  $\{\text{inc}, \text{reset}\}$ . The intended semantics of the transition is that on a configuration  $(p, h)$  of the automaton on the pair  $(a, d)$  the transition  $(p, a, \varphi(x), u, q)$  can be taken if  $\varphi(h(d))$  is true. The resulting configuration will be  $(q, h')$  where  $h'$  is  $h$  for all but  $d$  where  $h'(d) = h(d) + 1$  if  $u$  is *inc* and  $h'(d) = 0$  if *reset*.

This thesis focuses on classical logics on data words. For this purpose, data words can be represented as a first order structure  $w = ([n], \Sigma, <, +1, \sim)$  extending the corresponding representation for words due to Büchi. Here  $[n]$  denotes the set  $\{1, \dots, n\}$ , and  $\Sigma$  stands for the unary relations indicating the alphabet labelling. The binary relations  $<, +1$  are interpreted as the natural linear order and successor relations on the set  $[n]$ . The binary relation  $\sim$  denotes the equivalence relation on  $[n]$  given by the data values based on equality. That is to say,  $i \sim j$  if  $d_i = d_j$ . In addition if we have a linear order  $<_\Gamma$  on the data values then this uniquely defines a total preorder  $<_p$  (a total preorder is a reflexive, transitive and total binary relation) on the positions  $[n]$ . We denote the successor relation of  $<_p$  by  $+1_p$ . In the following we denote linear orders and their successor relations by  $<_{l_1}, <_{l_2}, \dots$  and by  $+1_{l_1}, +1_{l_2}, \dots$ .

It is easy to see that satisfiability and finite satisfiability of first order logic on data words,  $\text{FO}(\Sigma, <, \sim)$  is undecidable. The problems remain undecidable even for the fragment  $\text{FO}^3(\Sigma, <, \sim)$ , the set of formulas which uses at most 3 variables. Hence for decidability one has to look for suitable restrictions which are sufficiently expressive. Two-variable fragment is a natural candidate. It is known that satisfiability problem for first order logic with two variables is decidable [Mor75, GKV97]. Since it is not expressible in  $\text{FO}^2$  that a binary relation  $R$  is a linear order

(or an equivalence relation or a preorder), the above theorem does not imply satisfiability of  $\text{FO}^2$  over data words or over ordered data words. In a landmark paper [BMS<sup>+</sup>06] it was shown that,

**Theorem 1** ([BMS<sup>+</sup>06]). *Finite satisfiability of  $\text{FO}^2(\Sigma, <_{l_1}, +1_{l_1}, \sim)$  is decidable.*

Note that  $+1_{l_1}$  is not definable in terms of  $<_{l_1}$  using two-variables over words. This prompts us to add both the order and successor relations of the linear order to the vocabulary. [As a side note, it is also the case that  $+1_{l_1}$  is not definable in terms of  $<_{l_1}$  and  $\sim$  using two-variables over words.] Decidability holds even when  $<_{l_1}$  is the ordinal  $\omega$ . Status of the infinite satisfiability problem is not known.

However, the theorem fails for ordered data words;

**Proposition 1** ([BMS<sup>+</sup>06]). *Finite satisfiability problems of  $\text{FO}^2(\Sigma, <_{l_1}, +1_{l_1}, <_{p_2})$  and  $\text{FO}^2(\Sigma, <_{l_1}, +1_{l_1}, +1_{p_2})$  are undecidable. In fact, undecidability persists even when the equivalence classes of  $<_{p_2}$  are of size at most 2.*

This implies that in the presence of a total order on data values to get back decidability either  $<_{l_1}$  or  $+1_{l_1}$  has to be dropped from the vocabulary. The former case was undertaken in [SZ10] where it was shown that  $\text{FO}^2(\Sigma, <_{l_1}, <_{p_2}, +1_{p_2})$  is decidable. We consider the latter case when the preorder is in fact a linear order (in the case of data words it corresponds to the scenario when all the data values are distinct) and show that,

**Theorem 2.** *Finite satisfiability of  $\text{FO}^2(\Sigma, +1_{l_1}, +1_{l_2})$  is decidable.*

**Proposition 2.** *Finite satisfiability of  $\text{FO}^2(\Sigma, <_{l_1}, +1_{l_1}, <_{l_2}, +1_{l_2})$  is undecidable.*

Our proof is automata theoretic and makes use of Presburger automata. Concurrently, it was shown that removing at least one successor relation also results in decidability [SZ10], that is;

**Theorem 3** ([SZ10]). *Finite satisfiability of  $\text{FO}^2(\Sigma, <_{l_1}, +1_{l_1}, <_{l_2})$  is decidable.*

This raises the question whether  $\text{FO}^2$  is decidable if one of the order relations is absent from the vocabulary. This question is answered in the positive. In fact, a more general theorem is proved which says that  $\text{FO}^2(\Sigma, +1_{l_1}, <_{p_2}, +1_{p_2})$  is decidable where  $+1_{l_1}$  is a successor of a linear order and  $<_{p_2}, +1_{p_2}$  are a total preorder and its successor relation where the equivalence classes of the preorder is bounded by a constant. Note that it is to be contrasted with Proposition 1.

**Theorem 4.** *Finite satisfiability of  $\text{FO}^2(\Sigma, +1_{l_1}, <_{p_2}, +1_{p_2})$  is decidable when classes of  $<_{p_2}$  are of size at most  $k$ .*

For the proof, the notion of data automata are generalized so that they accept ordered data words. A translation from the above logic to these automata is established and finally the non-emptiness of these automata are shown to be decidable by reduction to reachability problem in vector addition systems. Since it is definable in  $\text{FO}^2$  that  $<_{p_2}$  is a linear order, this implies the answer to the previous question.

**Corollary 1.** *Finite satisfiability of  $\text{FO}^2(\Sigma, +1_{l_1}, <_{l_2}, +1_{l_2})$  is decidable.*

Though it is decidable, it is proved that the problem is as hard as reachability in vector addition systems.

## References

- [BMS<sup>+</sup>06] Mikolaj Bojanczyk, Anca Muscholl, Thomas Schwentick, Luc Segoufin, and Claire David. Two-variable logic on words with data. In *LICS*, pages 7–16. IEEE Computer Society, 2006.
- [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(1):53–69, 1997.
- [Mor75] M. Mortimer. On languages with two variables. *Zeitschr. f. math. Logik u. Grundlagen d. Math.*, 21:135–140, 1975.
- [SZ10] Thomas Schwentick and Thomas Zeume. Two-variable logic with two order relations. In *CSL*, volume 6247 of *LNCS*, pages 499–513, 2010.