Weighted One-Deterministic-Counter Automata

Prince Mathew
Indian Institute of Technology Goa, India

Vincent Penelle
Univ. Bordeaux, CNRS, Bordeaux INP, LaBRI, UMR 5800, F-33400, Talence, France

Prakash Saivasan
The Institute of Mathematical Sciences, HBNI, India
CNRS UMI ReLaX, India

A.V. Sreejith
Indian Institute of Technology Goa, India

Abstract

We introduce weighted one-deterministic-counter automata (ODCA). These are weighted one-counter automata (OCA) with the property of counter-determinacy, meaning that all paths labelled by a given word starting from the initial configuration have the same counter-effect. Weighted ODCAs are a strict extension of weighted visibly OCAs, which are weighted OCAs where the input alphabet determines the actions on the counter.

We present a novel problem called the co-VS (complement to a vector space) reachability problem for weighted ODCAs over fields, which seeks to determine if there exists a run from a given configuration of a weighted ODCA to another configuration whose weight vector lies outside a given vector space. We establish two significant properties of witnesses for co-VS reachability: they satisfy a pseudo-pumping lemma, and the lexicographically minimal witness has a special form. It follows that the co-VS reachability problem is in P.

These reachability problems help us to show that the equivalence problem of weighted ODCAs over fields is in P by adapting the equivalence proof of deterministic real-time OCAs [3] by Böhm et al. This is a step towards resolving the open question of the equivalence problem of weighted OCAs. Finally, we demonstrate that the regularity problem, the problem of checking whether an input weighted ODCA over a field is equivalent to some weighted automaton, is in P. We also consider boolean ODCAs and show that the equivalence problem for (non-deterministic) boolean ODCAs is in PSPACE, whereas it is undecidable for (non-deterministic) boolean OCAs.

2012 ACM Subject Classification Theory of computation → Automata extensions

Keywords and phrases One-counter automata, Equivalence, Weighted automata, Reachability

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2023.39

Related Version *Full Version*: `https://arxiv.org/abs/2301.13456`

Funding *A.V. Sreejith*: supported by SERB for the project "Probabilistic Pushdown Automata" [MTR/2021/000788].

Acknowledgements The authors would like to thank Rahul C S for his intuitive suggestions that helped in proving Lemma 14.

1 Introduction

This paper investigates a restriction on weighted one-counter automata (OCA). Like weighted finite automata, weighted OCAs recognise functions - every word over a finite alphabet is mapped to a weight. We say that a weighted OCA has *counter-determinacy* (see Definition 6) if "all paths labelled by a given word, starting from the initial configuration, have the same counter-effect". Weighted one-deterministic-counter automata (ODCA) is a syntactic model equivalent to weighted OCA with counter-determinacy (see Definition 7). It consists of,

43rd IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2023).
Editors: Patricia Bouyer and Srikanth Srinivasan; Article No. 39; pp. 39:1–39:23
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1. Counter: A counter that stays non-negative and allows zero tests.
2. Counter structure: A finite state deterministic machine where the transitions depend only on its current state, the input letter, and whether the counter is zero. The counter structure can increment/decrement the counter by one, or leave it unchanged.
3. Finite state machine: A finite state weighted automaton whose transitions depend on its current state, the input letter, and whether the counter value is zero. This machine cannot modify the counter.

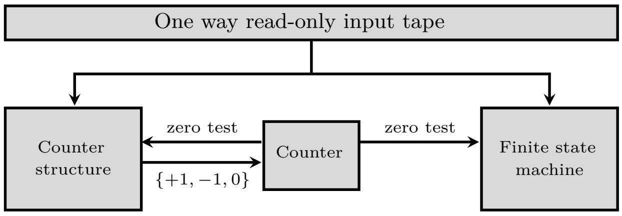


Figure 1 One-deterministic-counter automata.

The counter structure and the finite state machine run synchronously on any word. The finite state machine computes the weight associated with the word. Our first observation is:

▶ **Theorem 1.** *There is a polynomial time translation from a weighted* OCA *with counter-determinacy to a weighted* ODCA *and vice versa.*

In the following example, the functions `prefixAwareDecimal` and `equalPrefixPower` are recognised by weighted OCA with counter-determinacy.

▶ **Example 2.** *The functions are defined over the alphabet $\Sigma = \{a, b\}$. The transition weights of the* ODCA*s are from the field of rational numbers $\mathbb{Q}$.*

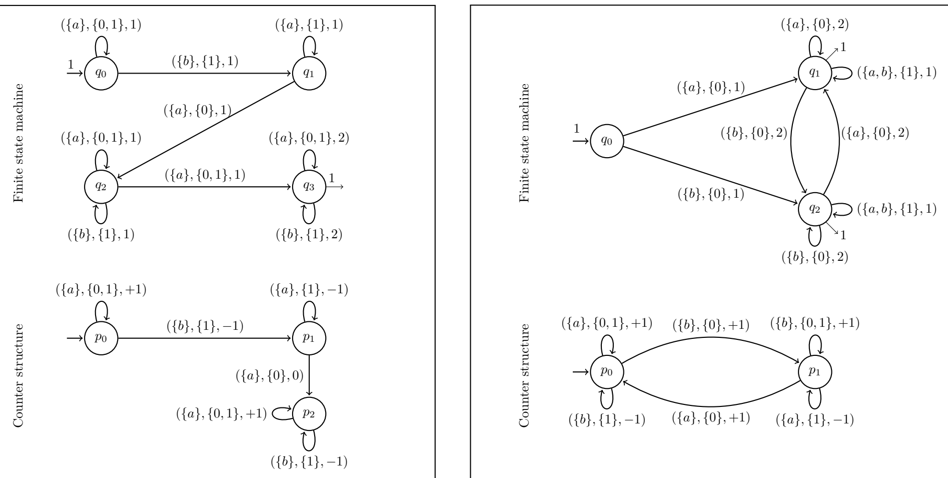
(a) *The function* `prefixAwareDecimal` $: \Sigma^* \to \mathbb{N}$ *is defined as follows:* `prefixAwareDecimal`$(w) =$ `decimal`(w_2) *if $w = w_1w_2$, $w_1 \in \{a^nba^n \mid n > 0\}$, and the number of a's $\geq$ number of b's for any prefix of w_2, and 0 otherwise. Here,* `decimal`(w_2) *represents the decimal equivalent of w_2 when interpreted as a binary number, where "a" is treated as a one and "b" as a zero.*

(b) *The function* `equalPrefixPower` $: \Sigma^* \to \mathbb{N}$ *is defined as follows: for all $w \in \Sigma^*$,* `equalPrefixPower`$(w) = 2^k$ *where k is the number of proper prefixes of w with equal number of a's and b's.*

The weighted ODCAs recognising these functions are given in Figure 2. In the figure, if a transition from p_i to p_j of the *counter structure* is labelled (A, R, D) and $(a, r, d) \in A \times R \times D$, then there is a transition from p_i to p_j on reading the symbol a with counter action d. If a transition from q_i to q_j of the *finite state machine* is labelled (A, R, s) and $(a, r, s) \in A \times R \times \mathbb{Q}$, then there is a transition from q_i to q_j on reading the symbol a with weight s. In both cases, r denotes the sign of the counter value. i.e., the current counter value should be 0 if $r = 0$ and greater than 0 if $r = 1$. For the finite state machine, the initial (resp. output) weight is marked using an inward (resp. outward) arrow. The weight of a path is the product of transition weights along that path. The accepting weight of a word is the sum of weights of all the paths from an initial state to an output state labelled by that word.

1.1 Comparisons with other models

Visibly pushdown automata (VPDA) were introduced by Alur and Madhusudan in 2004 [2]. They have received much attention as they are a strict subclass of pushdown automata suitable for program analysis. VPDAs enjoy tractable decidable properties, which are undecidable



(a) $\texttt{prefixAwareDecimal}(w_1w_2)$ = decimal value of w_2's binary interpretation, if $w_1 \in \{a^nba^n \mid n > 0\}$ and $\#a$'s $\geq \#b$'s for any prefix of w_2; 0 otherwise.

(b) $\texttt{equalPrefixPower}(w) = 2^k$ where k is the number of proper prefixes of w with equal number of a's and b's.

Figure 2 The figure shows weighted ODCAs recognising the functions given in Example 2.

in the general case. The visibly restriction, in essence, is that the stack operations are *input-driven*, i.e., only depends on the letter read. Weighted VPDA is a natural extension to the weighted setting. Counter-determinacy can be seen as a relaxation in the visibly constraint on OCAs, as the counter actions are no longer input-driven but are deterministic. The fact that weighted ODCAs are strictly more expressive than weighted visibly OCA can be noted from the fact that the functions in Example 2 are not recognised by a weighted visibly OCA.

Nowotka et al. [17] introduced height-deterministic pushdown automata, where the input string determines the stack height. Weighted ODCAs can be seen as weighted height-deterministic pushdown automata over a single stack alphabet and a bottom-of-stack symbol.

The reader might feel that a weighted ODCA is equivalent to a cartesian product of a deterministic OCA and a weighted finite automaton. However, one can note that the functions `prefixAwareDecimal` and `equalPrefixPower` in Example 2 are not definable by the cartesian product of deterministic OCA and a weighted automaton. The reason is that the weighted automaton cannot "see" the counter values, so its power is restricted.

1.2 Motivation

Probabilistic pushdown automata (pPDA) have been studied for the analysis of stochastic programs with recursion [14, 19]. They are equivalent to recursive Markov chains [8, 15]. pPDAs are also a generalisation of stochastic context-free grammars [1] used in natural language processing and many variants of one-dimensional random walks [7].

The decidability of equivalence of probabilistic pushdown automata is a long-standing open problem [10]. The problem is inter-reducible to multiplicity equivalence of context-free grammars. In fact, the decidability is only known for some special subclasses of pPDA. It is known that the equivalence problem for pPDA is in PSPACE if the alphabet contains only one letter and is at least as hard as polynomial identity testing [10]. There is a randomised polynomial time algorithm that determines the non-equivalence of two visibly pPDA over the alphabet triple $(\Sigma_{call}, \Sigma_{ret}, \Sigma_{int})$ where both machines perform push, pop, and no-action on the stack over the symbols in Σ_{call}, Σ_{ret}, and Σ_{int} respectively [12]. There is a polynomial-time reduction from polynomial identity testing to this problem. Hence it is highly unlikely that the problem is in P.

Since the equivalence problem for pPDA is unknown, the natural question to ask is the equivalence problem for probabilistic one-counter automata. However, this problem is also unresolved. In this paper, we identify a subclass of probabilistic OCAs (probabilistic ODCAs are also a superclass of visibly probabilistic OCAs) for which the equivalence problem is decidable. In particular, we show that the problem is in P. Note that our results are slightly more general since we consider weighted ODCAs where weights are from a field.

1.3 Our contributions on weighted ODCA (weights from a field)

The paper's primary focus is on the equivalence problem for weighted ODCAs where the weights are from a field (possibly infinite).

We first introduce a novel reachability problem on weighted ODCA, called the complement to a vector space (co-VS) reachability problem. The co-VS reachability problem (see Section 3) takes a weighted ODCA, an initial configuration, a vector space, a final counter state, and a final counter value as input. It asks, starting from the initial configuration, whether it is possible to reach a configuration with the final counter state, final counter value, and weight distribution over the states that is not in the vector space.

Let us call a word a *witness* if the run of the word "reaches" a configuration desired by the reachability problem. We identify two interesting properties of witnesses.

1. pseudo-pumping lemma (Lemma 10): If the run of a witness encounters a "large" counter value, then it can be pumped-down (resp. pumped-up) to get a run where the maximum counter value encountered is smaller (resp. larger). However, the lemma is distinct from a traditional pumping lemma, where the same subword can be pumped-down (or pumped-up) multiple times while maintaining reachability. In the case of a weighted ODCA, we only claim that a subword can be pumped, but the same subword may not be repeatedly pumped. It follows from the pseudo-pumping lemma that the co-VS reachability problem is in P (Theorem 13).
2. special-word lemma (Lemma 15): The lexicographically smallest witness is of the form $uy_1^{r_1}vy_2^{r_2}w$ where u, v, w, y_1 and y_2 are "small" words and $r_1, r_2 \in \mathbb{N}$. The length of the word uy_1vy_2w is bounded by a polynomial in the number of states of the ODCA, whereas r_1 and r_2 also depend on the counter values of the initial and final configurations.

Comparing the above properties with that of deterministic one-counter automata will be interesting. In a deterministic OCA, the reachability problem is equivalent to asking whether there is a path to a final state (rather than a weight distribution over states) and a counter value from an initial state and counter value. Let z be an arbitrary "long" witness. Consider the run on z of the deterministic OCA. By the Pigeonhole principle (see Valiant and Paterson [24]), there will be words u, y_1, v, y_2, and w such that $z = uy_1vy_2w$, and y_1 (and similarly y_2) starts and ends in the same state and the effect of y_1 on the counter is minus of the effect of y_2 on the counter. In short, y_1 and y_2 form loops with inverse counter-effects and can be pumped simultaneously. Therefore, for all $r \in \mathbb{N}$, the word $uy_1^rvy_2^rw$ is a witness. One can view this as a pumping lemma for OCA (see Ogden's lemma [18] for pushdown automata). Such a property does not hold in the case of weighted ODCA. The presence of weights at each state makes the problem inherently complex, necessitating a more sophisticated approach.

The proofs of Lemma 10 and Lemma 15 use linear algebra and combinatorics on words and are distinct from those employed for deterministic OCA. We also introduce a similar problem called co-VS coverability (see Section 3). The two properties of the witness and co-VS coverability are crucial along with the ideas developed by Böhm et al. [3, 4, 6] and Valiant and Paterson [24] in solving the equivalence problem. The proof is rather technical, Section 4 provides a high-level idea of the proof.

▶ **Theorem 3.** *There is a polynomial time algorithm that decides if two weighted ODCAs (weights from a field) are equivalent and outputs a word that distinguishes them otherwise.*

Finally, we consider the regularity problem - the problem of deciding whether a weighted ODCA is equivalent to some weighted automaton. The proof technique is adapted from the ideas developed by Böhm et al. [6] in the context of real-time OCA. The crucial idea in proving regularity is to check for the existence of infinitely many equivalence classes. The pseudo-pumping lemma (particularly pumping-up) is used in proving this. A detailed proof can be found in Appendix B.

▶ **Theorem 4.** *The regularity problem of weighted ODCA (weights from a field) is in* P.

1.4 Related work

Extensive studies have been conducted on weighted automata with weights from semirings. Tzeng [23] (also see Schützenberger [20]) gave a polynomial time algorithm to decide the equivalence of two probabilistic automata. The result has been extended to weighted automata with weights over a field. On the other hand, the problem is undecidable if the weights are over the semiring $(\mathbb{N}, \min, +)$ [13]. Unlike the extensive literature on weighted automata, the study on weighted versions of pushdown or one-counter machines is limited [9, 11, 14]. The undecidability of several interesting problems creates a major bottleneck.

Moving on to the non-weighted models, the equivalence problem for non-deterministic pushdown automata is known to be undecidable. On the other hand, from the seminal result by Sénizergues [21], we know that the equivalence problem for deterministic pushdown automata is decidable. It was later proved to be primitive recursive [22]. The language equivalence of synchronised real-time height-deterministic pushdown automata is in EXPTIME [17]. The equivalence problem for deterministic one-counter automata (with and without ϵ transitions), similar to that of deterministic finite automata, is NL-complete [5].

2 Preliminaries

2.1 Basic notations

In this paper, we fix an alphabet Σ. Given a word $w \in \Sigma^*$, we use $|w|$ to denote the length of the word w. For any set S, we use $|S|$ to denote the number of elements in S. We use the notation $[i, j]$ to denote the interval $\{i, i+1, \ldots, j\}$. We say that a word $u = a_1 \cdots a_k$ is a subword of a word w, if $w = u_0 a_1 u_1 a_2 \cdots a_k u_k$, where $a_i \in \Sigma$, $u_j \in \Sigma^*$ for all $i \in [1, k]$ and $j \in [0, k]$. We call u a proper subword of w if $u \neq w$. We say that a word u is a prefix of a word w if there exists $v \in \Sigma^*$ such that $w = uv$. Given a word $w = a_0 \cdots a_n$, we write $w[i \cdots j]$ to denote the factor $a_i \cdots a_j$. For a $d \in \mathbb{N}$, the sign of d (denoted by $sign(d)$) is defined as $sign(d) = 0$ if $d = 0$ and is 1 otherwise. For all $l \in \mathbb{N}$, we use $\Sigma^{\leq l}$ (resp. Σ^l) to denote the set of words over Σ having length less than or equal to l (resp. exactly equal to l).

2.2 Weighted one-deterministic-counter automata

In this section, we define weighted ODCA, where the weights are from a semiring. However, our results require that the weights come from some field $\mathcal{F}$ (not necessarily finite) except for Section 5, where the weights are from the boolean semiring. First, we define weighted one-counter automata.

▶ **Definition 5.** *A weighted one-counter automaton $\mathcal{A} = (Q, \boldsymbol{\lambda}, \delta_0, \delta_1, \boldsymbol{\eta})$, is defined over an alphabet Σ where, Q is a non-empty finite set of states, $\boldsymbol{\lambda} \in \mathcal{F}^{|Q|}$ is the initial distribution where the i^{th} component of $\boldsymbol{\lambda}$ indicates the initial weight on state $q_i \in Q$, $\delta_0 : Q \times \Sigma \times Q \times \{0, +1\} \to \mathcal{F}$ and $\delta_1 : Q \times \Sigma \times Q \times \{-1, 0, +1\} \to \mathcal{F}$ are the transition functions, and $\boldsymbol{\eta} \in \mathcal{F}^{|Q|}$ is the final distribution, where the i^{th} component of $\boldsymbol{\eta}$ indicates the output weight on state $q_i \in Q$.*

Note that the counter values do not go below zero. Let $p, q \in Q, a \in \Sigma, n \in \mathbb{N}, e \in \{-1, 0, +1\}$, and $s \in \mathcal{F}$. We say $(q, n) \hookrightarrow^{a|s} (p, n + e)$ if $\delta_{sign(n)}(q, a, p, e) = s$. Let $w = a_1 a_2 \cdots a_t \in \Sigma^*$ for some $t \in \mathbb{N}$. For a $q_0 \in Q$ and $n_0 \in \mathbb{N}$, we say $(q_0, n_0) \hookrightarrow^{w|s} (q_t, n_t)$ if for all $i \in [1, t]$, there are $q_i \in Q, n_i \in \mathbb{N}$, $s_i \in \mathcal{F}$ such that $(q_{i-1}, n_{i-1}) \hookrightarrow^{a_i|s_i} (q_i, n_i)$ and $s = \prod_{i=1}^{t} s_i$.

▶ **Definition 6.** *A weighted OCA with counter-determinacy is a weighted one-counter automaton $\mathcal{A} = (Q, \boldsymbol{\lambda}, \delta_0, \delta_1, \boldsymbol{\eta})$ with the following restriction: if $\boldsymbol{\lambda}[i]$ and $\boldsymbol{\lambda}[j]$ are non-zero for some $i, j \in [1, |Q|]$, then for all $w \in \Sigma^*$, if $(q_i, 0) \hookrightarrow^{w|s_1} (p_1, n_1)$ and $(q_j, 0) \hookrightarrow^{w|s_2} (p_2, n_2)$ for some $p_1, p_2 \in Q, n_1, n_2 \in \mathbb{N}$ and $s_1, s_2 \in \mathcal{F}$, then $n_1 = n_2$.*

We present a definition for weighted ODCA, which is an equivalent syntactic model.

▶ **Definition 7.** *A weighted ODCA, $\mathcal{A} = ((C, \delta_0, \delta_1, p_0), (Q, \boldsymbol{\lambda}, \Delta, \boldsymbol{\eta}))$ is defined over an alphabet Σ where,*

- *C is a non-empty finite set of counter states.*
- *$\delta_0 : C \times \Sigma \to C \times \{0, +1\}$, $\delta_1 : C \times \Sigma \to C \times \{-1, 0, +1\}$ are counter transitions.*
- *$p_0 \in C$ is the start state for the counter structure.*
- *Q is a non-empty finite set of states of the finite state machine.*
- *$\boldsymbol{\lambda} \in \mathcal{F}^{|Q|}$ is the initial distribution where the i^{th} component of $\boldsymbol{\lambda}$ indicates the initial weight on state $q_i \in Q$.*
- *$\Delta : \Sigma \times \{0, 1\} \to \mathcal{F}^{|Q| \times |Q|}$ gives the transition matrix. For all $a \in \Sigma$ and $d \in \{0, 1\}$, the component in the i^{th} row and j^{th} column of $\Delta(a, d)$ denotes the weight on the transition from state $q_i \in Q$ to state $q_j \in Q$ on reading symbol a from counter value n with $sign(n) = d$.*
- *$\boldsymbol{\eta} \in \mathcal{F}^{|Q|}$ is the final distribution, where the i^{th} component of $\boldsymbol{\eta}$ indicates the output weight on state $q_i \in Q$.*

Here, $(C, \delta_0, \delta_1, p_0)$ represents the counter structure and $(Q, \boldsymbol{\lambda}, \Delta, \boldsymbol{\eta})$ represents the finite state machine. Note that δ_0 and δ_1 are deterministic transition functions. The counter structure and the finite state machine run synchronously on any given word. A configuration $\mathtt{c}$ of an ODCA is of the form $(\mathbf{x}_{\mathtt{c}}, p_{\mathtt{c}}, n_{\mathtt{c}}) \in \mathcal{F}^{|Q|} \times C \times \mathbb{N}$. We use the notation WEIGHT-VECTOR($\mathtt{c}$) to denote $\mathbf{x}_{\mathtt{c}}$, COUNTER-STATE($\mathtt{c}$) to denote $p_{\mathtt{c}}$, and COUNTER-VALUE($\mathtt{c}$) to denote $n_{\mathtt{c}}$. The initial configuration is $(\boldsymbol{\lambda}, p_0, 0)$. A *transition* is a tuple $\tau = (\iota, d, a, \mathtt{ce}, \mathbb{A}, \theta)$ where $\iota, \theta \in C$ are counter states, $d \in \{0, 1\}$ denotes the sign of the counter value, $a \in \Sigma$, $\mathtt{ce} \in \{-1, 0, 1\}$ is the *counter-effect*, $\mathbb{A} \in \mathcal{F}^{|Q| \times |Q|}$ such that $\Delta(a, d) = \mathbb{A}$, and $\delta_d(\iota, a) = (\theta, \mathtt{ce})$. Given a transition $\tau = (\iota, d, a, \mathtt{ce}, \mathbb{A}, \theta)$ and a configuration $\mathtt{c} = (\mathbf{x}, n, p)$, we denote the application of τ to $\mathtt{c}$ as $\tau(\mathtt{c}) = (\mathbf{x}\mathbb{A}, \theta, n + \mathtt{ce})$ if $p = \iota$ and $d = sign(n)$; $\tau(\mathtt{c})$ is undefined otherwise. Note that the counter values are always non-negative.

Consider a sequence of transitions $T = \tau_0 \cdots \tau_\ell$ where $\tau_i = (\iota_i, d_i, a_i, \mathtt{ce}_i, \mathbb{A}_i, \theta_i)$ for all $i \in [0, \ell]$. We denote by $\mathtt{word}(T) = a_0 \cdots a_\ell$ the word labelling it, $\mathtt{we}(T) = \mathbb{A}_0 \cdots \mathbb{A}_\ell$ its weight-effect matrix, and $\mathtt{ce}(T) = \mathtt{ce}_0 + \cdots + \mathtt{ce}_\ell$ its counter-effect. For all $0 \leq i < j \leq \ell$, we use $T_{i \dots j}$ to denote the sequence of transitions $\tau_i \cdots \tau_j$ and $|T|$ to denote its length $\ell + 1$. We call T *floating* if for all $i \in [0, \ell - 1]$, $d_i = 1$ and *non-floating* otherwise.

A *run* π is an alternating sequence of configurations and transitions denoted as $\pi = \mathtt{c}_0 \tau_0 \mathtt{c}_1 \cdots \tau_{\ell-1} \mathtt{c}_\ell$ such that for every i, $\mathtt{c}_{i+1} = \tau_i(\mathtt{c}_i)$. The word labelling, length, weight-effect, and counter-effect of the run are those of its underlying sequence of transitions. Given a sequence of transitions $T = \tau_0 \cdots \tau_{\ell-1}$ and a configuration $\mathtt{c}$, we denote by $T(\mathtt{c})$ the run (if it is defined) $\mathtt{c}_0 \tau_0 \mathtt{c}_1 \cdots \tau_{\ell-1} \mathtt{c}_\ell$ where $\mathtt{c}_0 = \mathtt{c}$.

For any word w, there is at most one run labelled by w starting from a given configuration $\mathtt{c}_0$. We denote this run $\pi(w, \mathtt{c}_0)$. A run $\pi(w, \mathtt{c}_0) = \mathtt{c}_0 \tau_0 \mathtt{c}_1 \cdots \tau_{\ell-1} \mathtt{c}_\ell$ is also represented as $\mathtt{c}_0 \xrightarrow{w} \mathtt{c}_\ell$. We say $\mathtt{c}_0 \to^* \mathtt{c}_\ell$ if there is some word w such that $\mathtt{c}_0 \xrightarrow{w} \mathtt{c}_\ell$. For a weighted ODCA $\mathcal{A}$, the accepting weight of w is denoted by $f_{\mathcal{A}}(w, \mathtt{c}) = \boldsymbol{\lambda} \mathtt{we}(\pi(w, \mathtt{c})) \boldsymbol{\eta}^\top$, where $\mathtt{c}$ is the initial configuration of $\mathcal{A}$. Two weighted ODCAs $\mathcal{A}$ and $\mathcal{B}$ are equivalent if for all $w \in \Sigma^*$, $f_{\mathcal{A}}(w, \mathtt{c}) = f_{\mathcal{B}}(w, \mathtt{d})$ where $\mathtt{c}$ and $\mathtt{d}$ are the initial configurations of $\mathcal{A}$ and $\mathcal{B}$ respectively. Let $\mathtt{c}$ and $\mathtt{d}$ be configurations of ODCAs $\mathcal{A}$ and $\mathcal{B}$ respectively. We say that $\mathtt{c} \equiv_l \mathtt{d}$ if and only if for all $w \in \Sigma^{\leq l}$, $f_{\mathcal{A}}(w, \mathtt{c}) = f_{\mathcal{B}}(w, \mathtt{d})$ otherwise $\mathtt{c} \not\equiv_l \mathtt{d}$. An uninitialised weighted ODCA $\mathcal{A}$ is a weighted ODCA without an initial counter state and initial distribution. Weighted automata (WA) is a restricted form of weighted ODCA where the counter value is fixed at zero. The above notions of transitions, runs, acceptance, etc. are used for WA also. Given a weighted ODCA $\mathcal{A}$ and $M \in \mathbb{N}$, we define the M-unfolding weighted automata $\mathcal{A}^M$ as a finite state weighted automaton, where the accepting weight of any word whose run does not encounter counter values greater than M in $\mathcal{A}$ is equal in both $\mathcal{A}$ and $\mathcal{A}^M$. There is a polynomial time procedure to construct $\mathcal{A}^M$.

Consider the weighted ODCA $\mathcal{C}$ recognising the function $\mathtt{prefixAwareDecimal}$ given in Figure 2a. Here, $\boldsymbol{\lambda} = [1, 0, 0, 0]$ and $\boldsymbol{\eta} = [0, 0, 0, 1]$. The configuration $\mathtt{c}_0 = ([1, 0, 0, 0], p_0, 0)$ is the initial configuration of this machine. Let $w = abaaab$. The run of this machine on the word w can be written as:

$$\begin{aligned}\pi(w, \mathtt{c}_0) = ([1,0,0,0], p_0, 0) &\xrightarrow{a} ([1,0,0,0], p_0, 1) \xrightarrow{b} ([0,1,0,0], p_1, 0) \xrightarrow{a} ([0,0,1,0], p_2, 0) \\ &\xrightarrow{a} ([0,0,1,1], p_2, 1) \xrightarrow{a} ([0,0,1,2], p_2, 2) \xrightarrow{b} ([0,0,1,6], p_2, 1).\end{aligned}$$

The counter-effect of this run is $\mathtt{ce}(\pi(w, \mathtt{c}_0)) = 1$ and the weight-effect matrix is given by

$$\mathtt{we}(\pi(w, \mathtt{c}_0)) = \Delta(a,0)\ \Delta(b,1)\ \Delta(a,0)\ \Delta(a,1)\ \Delta(a,1)\ \Delta(b,1) = \begin{bmatrix} 0 & 0 & 1 & 6 \\ 0 & 0 & 1 & 14 \\ 0 & 0 & 1 & 46 \\ 0 & 0 & 0 & 64 \end{bmatrix}.$$

The accepting weight of the word w is $f_{\mathcal{C}}(w, \mathtt{c}_0) = \boldsymbol{\lambda} \mathtt{we}(\pi(w, \mathtt{c}_0)) \boldsymbol{\eta}^\top = 6$.

3 Reachability problems of weighted ODCA

In this section, we introduce the co-VS reachability and co-VS coverability problems for weighted ODCAs over a field $\mathcal{F}$. We fix a weighted ODCA $\mathcal{A} = ((C, \delta_0, \delta_1, p_0), (Q, \boldsymbol{\lambda}, \Delta, \boldsymbol{\eta}))$. We use $\mathcal{V} \subseteq \mathcal{F}^{|Q|}$ to denote a vector space and $\overline{\mathcal{V}}$ its complement. Let $S \subseteq C$ be a subset of the set of counter states, $X \subseteq \mathbb{N}$ a set of counter values, and $w \in \Sigma^*$. The notation $\mathtt{c} \xrightarrow{w} \overline{\mathcal{V}} \times S \times X$ denotes the run $\mathtt{c} \xrightarrow{w} \mathtt{d}$ where $\mathtt{d} \in \overline{\mathcal{V}} \times S \times X$ if it exists. We use $\mathtt{c} \xrightarrow{*} \overline{\mathcal{V}} \times S \times X$ to denote that there exists a word $u \in \Sigma^*$ such that $\mathtt{c} \xrightarrow{u} \overline{\mathcal{V}} \times S \times X$.

CO-VS REACHABILITY PROBLEM
INPUT: a weighted ODCA $\mathcal{A}$, an initial configuration $\mathtt{c}$, a vector space $\mathcal{V}$, a set of counter states S, and a counter value m.
OUTPUT: *Yes*, if there exists a run $\mathtt{c} \xrightarrow{*} \overline{\mathcal{V}} \times S \times \{m\}$ in $\mathcal{A}$. *No*, otherwise.

CO-VS COVERABILITY PROBLEM
INPUT: a weighted ODCA $\mathcal{A}$, an initial configuration $\mathtt{c}$, a vector space $\mathcal{V}$, and a set of counter states S.
OUTPUT: *Yes*, if there exists a run $\mathtt{c} \xrightarrow{*} \overline{\mathcal{V}} \times S \times \mathbb{N}$ in $\mathcal{A}$. *No*, otherwise.

Unlike the co-VS reachability problem, the final configuration's counter value is not considered part of the input for co-VS coverability problem. We assume that the vector space $\mathcal{V} \subseteq \mathcal{F}^{|Q|}$ is provided by giving a basis. We call $z \in \Sigma^*$ a *witness* of $(\mathtt{c}, \overline{\mathcal{V}}, S, X)$ if $\mathtt{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times X$. Furthermore, z is called a *minimal* witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, X)$ if for all $u \in \Sigma^*$ with $\mathtt{c} \xrightarrow{u} \overline{\mathcal{V}} \times S \times X$, $|u| \geq |z|$.

In the upcoming subsection, we give some interesting properties of minimal witnesses. In Section 3.2, we provide a pseudo-pumping lemma which helps us show that co-VS reachability and co-VS coverability are in P if the counter values are given in unary notation. Finally, in Section 3.3, we demonstrate that the lexicographically minimal witness has a canonical form. In the following subsections, $\mathcal{V}$ denotes a vector space, $\mathtt{c}$ a configuration, S a subset of counter states, and $X \subseteq \mathbb{N}$. We also denote by $K = |Q| \cdot |C|$, where C is the set of counter states, and Q is the set of states of the finite state machine.

3.1 Minimal witness and its properties

The following observation helps in breaking down the reachability problem into sub-problems. If $z \in \Sigma^*$ is a minimal witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, X)$, then for every z_1, z_2 such that $z = z_1 z_2$, there is a vector space $\mathcal{U}$ such that z_1 is a minimal witness for $(\mathtt{c}, \overline{\mathcal{U}}, \{p\}, \{n\})$ where p is the counter state and n is the counter value reached after reading z_1 from $\mathtt{c}$.

▶ **Observation 8.** *Consider arbitrary $z, z_1, z_2 \in \Sigma^*$ such that $z = z_1 z_2$. Let $\mathtt{d} = (\mathbf{x}_{\mathtt{d}}, p_{\mathtt{d}}, n_{\mathtt{d}})$ and $\mathtt{e} = (\mathbf{x}_{\mathtt{e}}, p_{\mathtt{e}}, n_{\mathtt{e}})$ be configurations such that $\mathtt{c} \xrightarrow{z_1} \mathtt{d} \xrightarrow{z_2} \mathtt{e}$ and $\mathbb{A} \in \mathcal{F}^{|Q| \times |Q|}$ be such that $\mathbf{x}_{\mathtt{d}} \mathbb{A} = \mathbf{x}_{\mathtt{e}}$. If z is a minimal witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, X)$, then z_1 is a minimal witness for $(\mathtt{c}, \overline{\mathcal{U}}, \{p_{\mathtt{d}}\}, \{n_{\mathtt{d}}\})$, where $\mathcal{U} = \{\mathbf{y} \in \mathcal{F}^{|Q|} \mid \mathbf{y}\mathbb{A} \in \mathcal{V}\}$.*

We aim to show that the length of a minimal witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, X)$ is polynomially bounded. The following lemma shows that if the counter values are polynomially bounded during the run of a minimal witness, then its length is also polynomially bounded.

▶ **Lemma 9.** *Let $z \in \Sigma^*$ be a minimal witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, X)$. If the number of distinct counter values encountered during the run $\mathtt{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times X$ is t, then $|z| \leq t \cdot K$.*

It now suffices to show that the counter values encountered during the run of a minimal witness are polynomially bounded.

3.2 Pseudo-pumping lemma

The pseudo-pumping lemma is a valuable tool in our analysis, allowing us to pump up or down a sufficiently long word while maintaining the reachability conditions.

▶ **Lemma 10** (pseudo-pumping lemma). *Let $m, R \in \mathbb{N}$, be such that* COUNTER-VALUE$(\mathtt{c}) = m$ *and $z \in \Sigma^*$ be such that $\mathtt{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times \{m\}$ is a floating run, and the maximum counter value encountered during this run is $m + R$. If $R > K^2$, then there exists $z_{sub}, z_{sup} \in \Sigma^*$ such that the following hold:*

1. *there exist $x, y, u, v, w \in \Sigma^*$ such that $z = xyuvw$, $z_{sub} = xuw$, $\mathtt{c} \xrightarrow{z_{sub}} \overline{\mathcal{V}} \times S \times \{m\}$ is a floating run, and the counter values encountered during this run are less than $m + R$, and*
2. *there exist $x, y, u, v, w \in \Sigma^*$ such that $z = xyuvw$, $z_{sup} = xy^2uv^2w$, $\mathtt{c} \xrightarrow{z_{sup}} \overline{\mathcal{V}} \times S \times \{m\}$ is a floating run, and the maximum counter value encountered in this run exceeds $m + R$.*

Proof. Let $z \in \Sigma^*$ be a witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, \{m\})$ and $\mathtt{e} \in \overline{\mathcal{V}} \times S \times \{m\}$ be such that $\mathtt{c} \xrightarrow{z} \mathtt{e}$ is a floating run, and the maximum counter value encountered in this run be $m + R$ where $R > K^2$. Let COUNTER-VALUE$(\mathtt{c}) = m$. There exist $z_1, z_2 \in \Sigma^*$ and configuration $\mathtt{f}$ such that $z = z_1 z_2$ and $\mathtt{c} \xrightarrow{z_1} \mathtt{f} \xrightarrow{z_2} \mathtt{e}$, where COUNTER-VALUE$(\mathtt{f}) = m + R$ (see Figure 3).

Let $\mathtt{c}_1 = \mathtt{c}$ and $\pi = \mathtt{c}_1 \tau_1 \mathtt{c}_2 \cdots \tau_{\ell-1} \mathtt{c}_\ell$ denote the run on word z from the configuration $\mathtt{c}_1$ and $T = \tau_1 \tau_2 \cdots \tau_{\ell-1}$ the sequence of transitions of π. For any $i \in [0, R]$, we denote by l_i and d_i the indices such that a configuration with counter value $m + i$ is encountered for the last (resp. first) time before (resp. after) reaching counter value $m + R$ in π. That is, COUNTER-VALUE$(\mathtt{c}_{l_i})$ = COUNTER-VALUE$(\mathtt{c}_{d_i}) = m + i$, and for any j where $l_i < j < d_i$, COUNTER-VALUE$(\mathtt{c}_j) > m + i$. To simplify the notation, we denote by $\mathtt{g}_i = \mathtt{c}_{l_i}$ and $\mathtt{g}'_i = \mathtt{c}_{d_i}$.

Consider the pairs of configurations $(\mathtt{g}_1, \mathtt{g}'_1), (\mathtt{g}_2, \mathtt{g}'_2), \ldots, (\mathtt{g}_R, \mathtt{g}'_R)$. Since $R > (|Q| \cdot |C|)^2$, by the Pigeonhole principle, there exist two counter states p, q, and a set of indices $I \subseteq [0, R]$ where $|I| = |Q|^2 + 1$ such that for all $h \in I$, COUNTER-STATE$(\mathtt{g}_h) = p$ and COUNTER-STATE$(\mathtt{g}'_h) = q$. For all $j \in I$, let $u_j, v_j, w_j \in \Sigma^*$ be such that $\mathtt{c}_1 \xrightarrow{u_j} \mathtt{g}_j \xrightarrow{v_j} \mathtt{g}'_j \xrightarrow{w_j} \mathtt{e}$. We use the following shorthand for any configuration $\mathtt{g}$: $\mathbf{x}_\mathtt{g}$ = WEIGHT-VECTOR$(\mathtt{g})$. For all $j \in I$, let matrix $\mathbb{A}_j$ and $\mathbb{B}_j$ be such that $\mathbf{x}_{\mathtt{g}'_j} = \mathbf{x}_{\mathtt{g}_j} \mathbb{A}_j$ and $\mathbf{x}_\mathtt{e} = \mathbf{x}_{\mathtt{g}'_j} \mathbb{B}_j$. Since $\mathbf{x}_\mathtt{e} \in \overline{\mathcal{V}}$, for all $j \in I$, $\mathbf{x}_{\mathtt{g}_j} \mathbb{A}_j \mathbb{B}_j \in \overline{\mathcal{V}}$. Let $r = |Q|^2 + 1$, and $i_1 < i_2 < \cdots < i_r$ be the indices in I. We prove the two cases separately.

1. Consider the sequence of matrices $\mathbb{A}_{i_r}, \mathbb{A}_{i_{r-1}}, \ldots, \mathbb{A}_{i_1}$. Since there can be at most $|Q|^2$ independent matrices, there exists $k \in [1, r]$ such that $\mathbb{A}_{i_k}$ is a linear combination of $\mathbb{A}_{i_r}, \ldots, \mathbb{A}_{i_{k+1}}$. Hence, there exists $h \in \{i_r, \ldots, i_{k+1}\}$ such that $\mathbf{x}_{\mathtt{g}_{i_k}} \mathbb{A}_h \mathbb{B}_{i_k} \in \overline{\mathcal{V}}$. Let $z_{sub} = u_{i_k} v_h w_{i_k}$. It is easy to observe that z_{sub} is a subword of z as mentioned in the lemma. To conclude the proof, it now suffices to show that z_{sub} is a witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, \{m\})$ and the counter values encountered during the run $\mathtt{c} \xrightarrow{z_{sub}} \mathtt{h}$ are less than $m + R$. Consider the floating run $\mathtt{g}_h \xrightarrow{v_h} \mathtt{g}'_h$. From the choice of $\mathtt{g}_h$ and $\mathtt{g}'_h$ we know that COUNTER-VALUE$(\mathtt{g}_h)$ = COUNTER-VALUE$(\mathtt{g}'_h) = m + h$ and for all j where $l_h < j < d_h$, COUNTER-VALUE$(\mathtt{c}_j) > m + h$. Since COUNTER-STATE$(\mathtt{g}_h)$ = COUNTER-STATE$(\mathtt{g}_{i_k})$, $\pi(v_h, \mathtt{g}_{i_k})$ is also a floating run $\mathtt{g}_{i_k} \xrightarrow{v_h} \mathtt{d}$ such that COUNTER-STATE$(\mathtt{g}'_h)$ = COUNTER-STATE$(\mathtt{d})$, COUNTER-VALUE$(\mathtt{g}_{i_k})$ = COUNTER-VALUE$(\mathtt{d}) = m + i_k < m + h$, and the minimum and maximum counter values encountered in the run is $m + i_k$ and $m + R - (h - i_k)$ respectively (see Figure 3). Furthermore, $\mathbf{x}_d = \mathbf{x}_{\mathtt{g}_{i_k}} \mathbb{A}_h$. Since COUNTER-STATE$(\mathtt{g}'_{i_k})$ = COUNTER-STATE$(\mathtt{g}'_h)$, we get that COUNTER-STATE$(\mathtt{g}'_{i_k})$ = COUNTER-STATE$(\mathtt{d})$. Moreover, since COUNTER-VALUE$(\mathtt{g}'_{i_k})$ = COUNTER-VALUE$(\mathtt{g}_{i_k})$, we have COUNTER-VALUE$(\mathtt{g}'_{i_k})$ = COUNTER-VALUE$(\mathtt{d})$. Therefore, $\pi(w_{i_k}, \mathtt{d})$ is the run $\mathtt{d} \xrightarrow{w_{i_k}} \mathtt{h}$ where $\mathbf{x}_\mathtt{h} = \mathbf{x}_\mathtt{d} \mathbb{B}_{i_k}$ and hence $\mathbf{x}_\mathtt{h} = \mathbf{x}_{\mathtt{g}_{i_k}} \mathbb{A}_h \mathbb{B}_{i_k} \in \overline{\mathcal{V}}$. This concludes that z_{sub} is a witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, \{m\})$ and satisfies the properties mentioned in the lemma.

2. Consider the sequence of matrices: $\mathbb{A}_{i_1}, \mathbb{A}_{i_2}, \ldots, \mathbb{A}_{i_r}$. Note that the matrices are ordered in reverse compared to the ordering in the previous case. By following a similar argument, we get a word z_{sup} satisfying the required properties. ◀

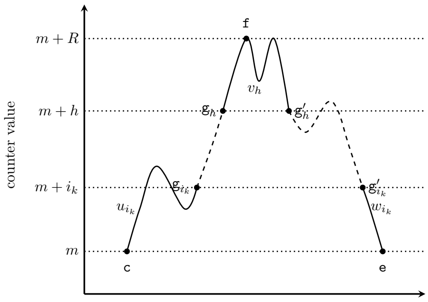


Figure 3 The figure shows the floating run from a configuration c with COUNTER-VALUE(c) $= m$ to a configuration $\mathsf{e} = (\mathbf{x}, p, m)$ such that $\mathbf{x} \in \overline{\mathcal{U}}$. Configurations g_{i_k} and g_h (resp. g'_{i_k} and g'_h) are where the counter values $m + i_k$ and $m + h$ are encountered for the last (resp. first) time before (resp. after) reaching $m + R$. Also, COUNTER-STATE(g_{i_k}) = COUNTER-STATE(g_h) and COUNTER-STATE(g'_h) = COUNTER-STATE(g'_{i_k}). The dashed line denotes the part of the run that can be removed to get a shorter witness for $(\mathsf{c}, \overline{\mathcal{U}}, \{p\}, \{n\})$.

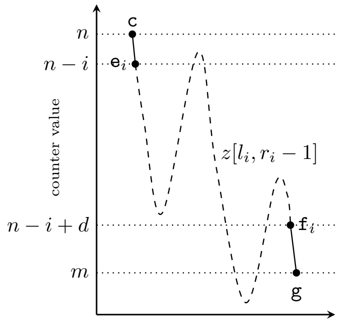


Figure 4 The figure shows the floating run from a configuration c with COUNTER-VALUE(c) $= n$ to a configuration $\mathsf{g} = (\mathbf{x}, p, m)$ such that $\mathbf{x} \in \overline{\mathcal{V}}$. The points e_i and $\mathtt{f}_i$ denotes the configurations where the counter values $n - i$ and $n - i + d$ are encountered for the first (resp. last) time during this run. The dashed line represents the part of the run due to factor $z[l_i, r_i - 1]$ and has a counter effect d.

It is important to note that we do not end up in the same configuration while pumping up/down, but we ensure that we reach a configuration with the same counter state, counter value, and whose weight vector is in the complement of the given vector space.

Now, we prove that for any run (it need not necessarily be a floating run) of a minimal reachability witness z for $(\mathsf{c}, \overline{\mathcal{V}}, S, \{m\})$, the maximum counter value encountered during the run $\mathsf{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times \{m\}$ is bounded by a polynomial in the number of states of the machine, and the initial and final counter values. This can be achieved by iteratively applying Lemma 10 on the run of the minimal witness (refer Figure 6) and using Observation 8 and Lemma 9.

▶ **Corollary 11.** *If $z \in \Sigma^*$ is a minimal witness for $(\mathsf{c}, \overline{\mathcal{V}}, S, \{m\})$, then*

1. *the maximum counter value encountered during the run $\mathsf{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times \{m\}$ is less than max(COUNTER-VALUE(c)$, m) + K^2$, and*
2. *$|z| \leq K^3 + max$(COUNTER-VALUE(c)$, m) \cdot K$.*

The following lemma (depicted in Figure 5) helps us show that the length of a minimal witness for co-VS coverability is polynomially bounded in the number of states.

▶ **Lemma 12** (cut lemma)**.** *Let $z \in \Sigma^*$ be a witness for $(\mathsf{c}, \overline{\mathcal{V}}, S, \mathbb{N})$, where c is a configuration with* COUNTER-VALUE(c) $= n$ *for some $n \in \mathbb{N}$, and $\mathsf{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times \{m\}$ is a floating run for some $m \in \mathbb{N}$. If $m - n > K$, then there exists $z_{sub} \in \Sigma^*$ such that z_{sub} is a subword of z, $\mathsf{c} \xrightarrow{z_{sub}} \overline{\mathcal{V}} \times S \times \{m'\}$ is a floating run and $m' - n < m - n$.*

Now, we prove that the co-VS reachability and co-VS coverability problems of weighted ODCA are in P by demonstrating a small model property. We have already established using Lemma 10, Corollary 11, and Lemma 12 that the maximum and minimum counter values encountered during the run of the minimal witness do not exceed some polynomial bound. This, in turn, implies a polynomial bound on the length of the witness by Lemma 9. As a result, we get the following theorem.

▶ **Theorem 13.** *The co-VS reachability and co-VS coverability problems for weighted* ODCA *can be decided in polynomial time when the counter values are given in unary notation.*

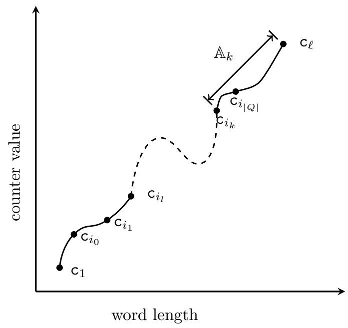


Figure 5 The figure shows a run from configuration c_1 to $\mathsf{c}_\ell = (\mathbf{x}_{\mathsf{c}_\ell}, p_{\mathsf{c}_\ell}, n_{\mathsf{c}_\ell})$ such that $\mathbf{x}_{\mathsf{c}_\ell} \in \overline{\mathcal{V}}$. The configurations c_{i_l} and c_{i_k} are where the counter values $n_{\mathsf{c}_{i_l}}$ and $n_{\mathsf{c}_{i_k}}$ are encountered for the last time. Also the configurations c_{i_l} and c_{i_k} have the same counter state. The dashed line is the part that can be removed to get a shorter witness for $(\mathsf{c}, \overline{\mathcal{V}}, \{p_{\mathsf{c}_\ell}\}, \mathbb{N})$.

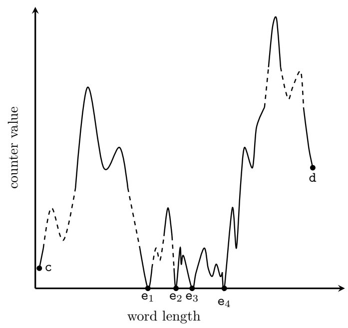


Figure 6 The figure shows a run from configuration c to $\mathsf{d} = (\mathbf{x}_\mathsf{d}, p_\mathsf{d}, n_\mathsf{d})$ such that $\mathbf{x}_\mathsf{d} \in \overline{\mathcal{V}}$. Configurations $\mathsf{e}_1, \mathsf{e}_2, \mathsf{e}_3, \mathsf{e}_4$ are where the counter value zero is encountered during the run. The dashed lines denote the parts that can be removed to obtain a shorter witness for $(\mathsf{c}, \overline{\mathcal{V}}, \{p_\mathsf{d}\}, \{n_\mathsf{d}\})$.

3.3 Lexicographically minimal witness

This section will show that the lexicographically minimal witness has a distinct structure.

We assume a total order on the symbols in Σ. Given two words $u, v \in \Sigma^*$, we say that u precedes v in the *(length) lexicographical ordering* if $|u| < |v|$ or if $|u| = |v|$ and there exists an $i \in [0, |u|-1]$ such that $u[0, i-1] = v[0, i-1]$ and $u[i]$ precedes $v[i]$ in the total ordering assumed on Σ. A word $z \in \Sigma^*$ is called the lexicographically minimal witness for $(\mathsf{c}, \overline{\mathcal{V}}, S, \{m\})$, if $\mathsf{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times \{m\}$ and for all $u \in \Sigma^* \setminus \{z\}$ with $\mathsf{c} \xrightarrow{u} \overline{\mathcal{V}} \times S \times \{m\}$, z precedes u in the lexicographical ordering. We show that the lexicographically minimal witness z for $(\mathsf{c}, \overline{\mathcal{V}}, S, \{m\})$ has a canonical form. First, we prove this for floating runs.

▶ **Lemma 14.** *There exist polynomials $p_1 : \mathbb{N} \to \mathbb{N}$, and $p_2 : \mathbb{N}^2 \to \mathbb{N}$ such that, if $z \in \Sigma^*$ is the lexicographically minimal witness for $(\mathsf{c}, \overline{\mathcal{V}}, S, \{m\})$ and $\mathsf{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times \{m\}$ is a floating run, then there exist $u, y, w \in \Sigma^*$ and $r \in \mathbb{N}$ such that $z = uy^r w$ and the following are true:*
1. *$|uyw| \le p_1(K)$, and*
2. *$r \le p_2(K, |\text{COUNTER-VALUE}(\mathsf{c}) - m|)$.*

Proof. Let z be the lexicographically minimal witness for $(\mathsf{c}, \overline{\mathcal{V}}, S, \{m\})$, and $\mathsf{g} \in \overline{\mathcal{V}} \times S \times \{m\}$ such that $\mathsf{c} \xrightarrow{z} \mathsf{g}$ is a floating run. Let n be such that $\text{COUNTER-VALUE}(c) = n$. We consider the case $n > m$. The case where $m \ge n$ is analogous. Let $t = n - m$.

▷ Claim 1. $|z| \le 2K^3 + t \cdot K$.

Proof. From Point 1 of Lemma 10, it follows that the maximum counter value during the run $\mathsf{c} \xrightarrow{z} \mathsf{g}$ is less than $n + K^2$. By a symmetric argument, it follows that the minimum counter value during the run is greater than $m - K^2$. Hence, there are at most $t + 2K^2$ distinct counter values during the run. From Lemma 9 it follows that $|z| \le 2K^3 + t \cdot K$. ◁

If $t \le K^2$, then from Claim 1, we get that $|z| \le 3K^3$, and the lemma is trivially true. Let us assume $t > K^2$ and let $d = K^2 - t$. Note that d is a negative number. Let $\mathsf{c}_1 = \mathsf{c}$ and $\pi(z, \mathsf{c}_1) = \mathsf{c}_1 \tau_1 \mathsf{c}_2 \cdots \tau_{\ell-1} \mathsf{c}_\ell$ denote the run on word z from c. For any $i \in [0, K^2]$, we denote by l_i the index such that the counter value $n - i$ is encountered for the first time, and r_i the index such that the counter value $n - i + d$ is encountered for the last time in

$\pi(z, \mathtt{c}_1)$ (see Figure 4). Let $I = \{(l_i, r_i)\}_{i \in [0, K^2]}$ be the set of these pairs of indices, and let $W = \{z[l, r-1] \mid (l, r) \in I\}$ be the set of corresponding factors. Note that $|I| > K^2$. We argue that these factors $z[l_i, r_i - 1]$ for $i \in [0, K^2]$ need not all be distinct.

▷ Claim 2. $|W| \leq K^2$.

Proof. Assume for contradiction that $|W| > (|Q| \cdot |C|)^2$. Since the number of counter states is $|C|$, by Pigeonhole principle there exists $Y \subseteq I$ with $|Y| = |Q|^2 + 1$ such that for all $(l, r), (l', r') \in Y$, configurations $\mathtt{c}_l$ and $\mathtt{c}_{l'}$ have the same counter state, configurations $\mathtt{c}_r$ and $\mathtt{c}_{r'}$ have the same counter state, and $z[l, r-1] \neq z[l', r'-1]$. We say $(l, r) < (l', r')$ if $z[l, r-1]$ precedes $z[l', r'-1]$ in the lexicographical order. Therefore, the elements in Y have an ordering as follows: $(l_0, r_0) < (l_1, r_1) < \cdots < (l_{|Q|^2}, r_{|Q|^2})$. For any configuration $\mathtt{h}$, let $\mathbf{x}_{\mathtt{h}} = \textsc{weight-vector}(\mathtt{h})$. For all $i \in [0, |Q|^2]$, let $u_i = z[1, l_i - 1], x_i = z[l_i, r_i - 1], w_i = z[r_i, \ell - 1]$, configurations $\mathtt{e}_i, \mathtt{f}_i$ be such that $\mathtt{c} \xrightarrow{u_i} \mathtt{e}_i \xrightarrow{x_i} \mathtt{f}_i \xrightarrow{w_i} \mathtt{g}$ and matrices $\mathbb{A}_i, \mathbb{M}_i, \mathbb{B}_i$ be such that $\mathbf{x}_{\mathtt{e}_i} = \mathbf{x}_{\mathtt{c}} \mathbb{A}_i$, $\mathbf{x}_{\mathtt{f}_i} = \mathbf{x}_{\mathtt{e}_i} \mathbb{M}_i$, $\mathbf{x}_{\mathtt{g}} = \mathbf{x}_{\mathtt{f}_i} \mathbb{B}_i$.

We know that for all $k \in [0, |Q|^2]$, $\mathbf{x}_{\mathtt{c}} \mathbb{A}_k \mathbb{M}_k \mathbb{B}_k \in \overline{\mathcal{V}}$. Consider the sequence of matrices $\mathbb{M}_0, \mathbb{M}_1, \cdots, \mathbb{M}_{|Q|^2}$. Since there can be at most $|Q|^2$ independent matrices, we get that there exists $i \in [0, |Q|^2]$ such that $\mathbb{M}_i$ is a linear combination of $\mathbb{M}_0, \ldots, \mathbb{M}_{i-1}$. Hence, we get that there exists a j where $j < i$ such that $\mathbf{x}_{\mathtt{c}} \mathbb{A}_i \mathbb{M}_j \mathbb{B}_i \in \overline{\mathcal{V}}$. Since $x_j = z[l_j, r_j - 1]$ precedes $x_i = z[l_i, r_i - 1]$, the word $u_i x_j w_i$ precedes z in the lexicographical ordering. Therefore the run $\pi(u_i x_j w_i, \mathtt{c})$ contradicts the lexicographical minimality of z. ◁

Since $|W| \leq K^2$ and $|I| > K^2$, there exists $i, j \in [0, K^2]$, with $i < j$ and $x \in \Sigma^*$ such that $(l_i, r_i) \in I, (l_j, r_j) \in I$ and $x = z[l_i, r_i - 1] = z[l_j, r_j - 1]$ (see Figure 7). Let $u_1, w_1, u_2, w_2 \in \Sigma^*$ such that $z = u_1 x w_1 = u_2 x w_2$. Since $u_1 \neq u_2$, either u_1 is a prefix of u_2 or u_2 a prefix of u_1. Without loss of generality, let us assume u_1 is a prefix of u_2. Therefore, there exists $v \in \Sigma^*$ such that $u_2 = u_1 v$. Let $\mathtt{e}$ be a configuration such that $\mathtt{c} \xrightarrow{u_1} \mathtt{e}$.

▷ Claim 3. $|u_1|, |v|, |w_1| \leq 3K^3$.

Proof. Consider the set I. For any $i, j \in [0, K^2]$, the difference between the counter values of configurations $\mathtt{c}_{l_i}$ and $\mathtt{c}_{l_j}$ and the difference between the counter values of the configurations $\mathtt{c}_{r_j}$ and $\mathtt{c}_{r_i}$ is at most $K^2 + 1$. Therefore the counter-effect of u_2, w_2, and v can be at most K^2. Since $\pi(v, \mathtt{e})$ is a floating run from Claim 1, we get that $|v| \leq 3K^3$. By similar arguments, the counter-effect of u_1 and w_1 can be at most K^2, and again by Claim 1, we get that their lengths are at most $3K^3$. ◁

▷ Claim 4. There exist $v' \in \Sigma^*$ and $r \in [0, K^3 + t \cdot K]$ such that $x = v^r v'$ with $|v'| \leq |v|$.

Proof. Let $r \in \mathbb{N}$ be the largest number such that x is of the form $v^r v'$ for some $v' \in \Sigma^*$ (see Figure 7). We know that $z = u_2 x w_2$ and $u_2 = u_1 v$. Therefore, $z = u_1 v x w_2 = u_1 v v^r v' w_2 = u_1 v^r v v' w_2$. Furthermore, $z = u_1 x w_1 = u_1 v^r v' w_1$. Now since $u_1 v^r v v' w_2 = u_1 v^r v' w_1$, we get that $v v' w_2 = v' w_1$. Hence, if $|v'| \geq |v|$, then v is a prefix of v'. This is a contradiction since r was chosen to be the largest number such that x is of the form $v^r v'$.

To show the bound on the value r, we observe the following. We know that the counter effect of the run $\pi(x, \mathtt{e})$ is d. Therefore from Claim 1, we get that $|x| \leq 2K^3 + |d| \cdot K$. Hence, $r \leq 2K^3 + |d| \cdot K$. ◁

From Claim 4 and Claim 3, we get that $|u_1 v v' w_1| \leq 12K^3$ and $z = u_1 v^r v' w_1$ for some $r \in [0, 2K^3 + |d| \cdot K)]$. ◀

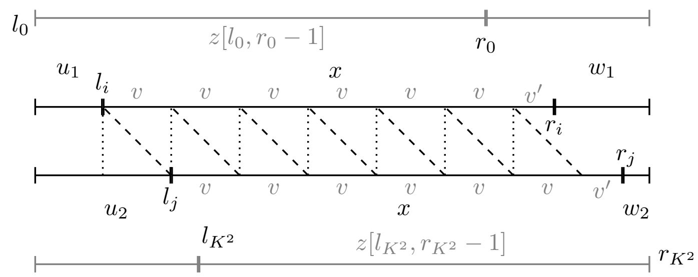


Figure 7 The figure shows the factorisation of a word $z = u_1xw_1 = u_2xw_2$, where $x = z[l_i, r_i - 1] = z[l_j, r_j - 1]$, and $u_1 \neq u_2$. The factor v is a prefix of x such that $u_2 = u_1v$. The word z can be written as $u_1v^iv'w_2$ for some $i \in \mathbb{N}$ and v' prefix of v. For $k \in [0, K^2]$, l_k is the index such that the counter value $n - k$ is encountered for the first time and r_k the index such that the counter value $n - k + d$ is encountered for the last time during the run $\mathtt{c} \xrightarrow{z} \mathtt{g}$.

We now establish that the lexicographically minimal witness z (whose run need not be floating) for a co-VS reachability problem has the form $uy_1^{r_1}vy_2^{r_2}w$. Here, lengths of the words u, y_1, y_2, v, and w are polynomially bounded in the number of states, and r_1 and r_2 are polynomial values dependent on the number of states and the input counter values.

▶ **Lemma 15** (special-word lemma). *If $z \in \Sigma^*$ is the lexicographically minimal witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, \{m\})$, then there exists $u, y_1, v_1, v_2, v_3, y_2, w \in \Sigma^*$ and $r_1, r_2 \in \mathbb{N}$ such that $z = uy_1^{r_1}vy_2^{r_2}w$ and the following are true:*

1. *$|uy_1vy_2w|$ is polynomially bounded in the number of states of the machine.*
2. *r_1 and r_2 are polynomially bounded in the number of states of the machine, m, and* COUNTER-VALUE$(\mathtt{c})$.

4 Equivalence of weighted ODCA

In this section, we present a polynomial time algorithm to decide the equivalence of two weighted ODCAs whose weights come from a fixed field. The techniques developed in the previous section in conjunction with those presented in Valiant and Paterson [24], and Böhm et al. [3] for deterministic real-time OCA give us the algorithm. Here, we give a proof sketch of Theorem 3.

In the remainder of this section, we fix two non-equivalent weighted ODCAs $\mathcal{A}_1$ and $\mathcal{A}_2$ over an alphabet Σ and a field $\mathcal{F}$. We fix a minimal word z (also called *witness*) that distinguishes them. We denote a configuration pair as $\mathtt{h}_i = \langle \mathtt{c}_i, \mathtt{d}_i \rangle$ where $\mathtt{c}_i$ is a configuration of $\mathcal{A}_1$ and $\mathtt{d}_i$ is a configuration of $\mathcal{A}_2$. We denote by $\Pi = \mathtt{h}_0\tau_0\mathtt{h}_1 \cdots \tau_{\ell-1}\mathtt{h}_{|z|}$ the synchronisation of runs over z in $\mathcal{A}_1$ and $\mathcal{A}_2$ from their initial configurations, where $\mathtt{h}_i$ are pairs of configurations and τ_i are pairs of transitions. The main idea to prove Theorem 3 is to show that the length of z is polynomially bounded in the size of the two weighted ODCAs.

▶ **Lemma 16.** *There is a polynomial* $\mathrm{poly}_0 : \mathbb{N} \to \mathbb{N}$ *such that if two weighted* ODCAs $\mathcal{A}_1$ *and* $\mathcal{A}_2$ *are not equivalent, then there exists a witness z such that the counter values encountered during Π are less than* $\mathrm{poly}_0(max(|\mathcal{A}_1|, |\mathcal{A}_2|))$.

Lemma 16 allows to show that the length of the witness z is bounded by a polynomial $\mathrm{poly}_1(max(|\mathcal{A}_1|, |\mathcal{A}_2|))$. Thus we can reduce the equivalence problem of weighted ODCA over fields to that of weighted automata over fields (which is in P [23]) by "simulating" the runs of weighted ODCAs $\mathcal{A}_1$ and $\mathcal{A}_2$ up to length $\mathrm{poly}_1(max(|\mathcal{A}_1|, |\mathcal{A}_2|))$ using two weighted automata that are unfolding of the weighted ODCAs upto counter value $\mathrm{poly}_1(max(|\mathcal{A}_1|, |\mathcal{A}_2|))$.

The rest of this section is dedicated to proving Lemma 16. Following Böhm et al. [3], we define a partition of the set of configuration pairs to facilitate this. We partition the set of configuration pairs into three: initial space, belt space, and background space (see Figure 8). Given a configuration $\mathtt{c}$, we use $n_{\mathtt{c}}$ to denote COUNTER-VALUE($\mathtt{c}$).

- *initial space*: All configuration pairs $\langle \mathtt{c}, \mathtt{d} \rangle$ such that $n_{\mathtt{c}}, n_{\mathtt{d}} < \mathrm{poly}_2(max(|\mathcal{A}_1|, |\mathcal{A}_2|))$.
- *belt space*: Let α, β be co-prime numbers whose values are bounded by a polynomial in $|\mathcal{A}_1|$ and $|\mathcal{A}_2|$. A belt of slope $\frac{\alpha}{\beta}$ consists of those configuration pairs $\langle \mathtt{c}, \mathtt{d} \rangle$ outside the initial space that satisfies $|\alpha.n_{\mathtt{c}} - \beta.n_{\mathtt{d}}| \leq \mathrm{poly}_3(max(|\mathcal{A}_1|, |\mathcal{A}_2|))$. The belt space contains all configuration pairs $\langle \mathtt{c}, \mathtt{d} \rangle$ that are inside belts with slope $\frac{\alpha}{\beta}$.
- *background space*: All remaining configuration pairs.

These partitions are indexed on two carefully chosen polynomials poly_2 and poly_3, so that all belts are disjoint outside the initial space.

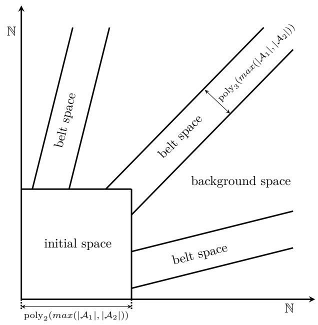


Figure 8 Projection of configuration space.

To prove Lemma 16, there are two cases to consider: either there is no background space point in Π, or there is a background space point in Π.

Case 1: When there is no background space point in Π

Since there is no background space point in Π, all the points in Π are either in the initial or belt space. By definition, the counter values of configuration pairs inside the initial space are bounded by $\mathrm{poly}_2(max(|\mathcal{A}_1|, |\mathcal{A}_2|))$. Now, we look at the sub-run of Π inside the belt space. If a sub-run of Π enters and exits a belt from the initial space or if Π ends inside a belt, then we show that the counter values encountered during that belt visit are polynomially bounded. For this proof, it is crucial that the belts are disjoint.

Consider a pair of sub-run inside a belt. The counter values of $\mathcal{A}_1$ and $\mathcal{A}_2$ are related by a linear expression. In particular, given the counter value of $\mathcal{A}_1$, the counter value of $\mathcal{A}_2$ can only have polynomial many choices. Hence an ODCA $\mathcal{A}_3$ can simulate this run of the pair inside a belt by tracking the counter value of $\mathcal{A}_1$ in its counter and the other counter value using finitely many states. We use co-Vs reachability/coverability to show that if a sub-run of this ODCA $\mathcal{A}_3$ reaches counter values higher than some polynomial in $|\mathcal{A}_1|$ and $|\mathcal{A}_2|$, then there exists a shorter witness (contradicting minimality). We achieve this by applying pseudo-pumping lemma (Lemma 10) and cut lemma (Lemma 12) on this sub-run. Hence, we get that the pair of runs of the minimal witness cannot reach counter values higher than some polynomial bound if it does not enter the background space.

Case 2: When there is a background space point in Π

We now consider the case where the witness ultimately enters the background space. Using co-VS reachability, we prove that the counter values encountered during Π till the first background space point are polynomially bounded. We also show that the length of the remaining run is polynomially bounded in the number of states of the machines.

To that end, we need the notion of *underlying uninitialised weighted automaton.* Roughly speaking, an underlying uninitialised weighted automaton of an ODCA $\mathcal{A}$ is the uninitialised weighted automaton $\mathrm{U}(\mathcal{A})$ that is syntactically equivalent to $\mathcal{A}$ without zero tests. In other words, the transition function of $\mathrm{U}(\mathcal{A})$ will be determined by the transition functions of $\mathcal{A}$ for positive counter values. Floating runs of $\mathcal{A}$ are isomorphic to runs of this weighted automaton $\mathrm{U}(\mathcal{A})$. Given $k \in \mathbb{N}$, a configuration $\mathtt{c}$ of a weighted ODCA $\mathcal{A}$ is said to be k-equivalent to a configuration $\bar{\mathtt{c}}$ of an uninitialised weighted automaton $\mathcal{B}$, if for all $w \in \Sigma^{\leq k}$, $f_{\mathcal{A}}(w, \mathtt{c}) = f_{\mathcal{B}}(w, \bar{\mathtt{c}})$. We say that $\mathtt{c}$ is not k-equivalent to $\bar{\mathtt{c}}$ otherwise.

We consider the uninitialised weighted automaton $\mathcal{B}$, which is a disjoint union of $\mathrm{U}(\mathcal{A}_1)$ and $\mathrm{U}(\mathcal{A}_2)$. This gives us a single automaton with which we can compare the configurations of $\mathcal{A}_1$ and $\mathcal{A}_2$. For $i \in \{1,2\}$, let C_i be the set of counter states of $\mathcal{A}_i$. For all $p \in C_i$ and $m < |\mathcal{B}|$, we define the sets $\mathcal{W}_i^{p,m}$. The set $\mathcal{W}_i^{p,m}$ contains vectors $\mathbf{x}$ such that the configuration $(\mathbf{x}, p, m)$ of $\mathcal{A}_i$ is $|\mathcal{B}|$-equivalent to some configuration of $\mathcal{B}$. The set $\overline{\mathcal{W}}_i^{p,m}$ is the complement of $\mathcal{W}_i^{p,m}$. For any $i \in \{1,2\}$, $p \in C_i$ and $m < |\mathcal{B}|$, the set $\mathcal{W}_i^{p,m}$ is a vector space. The distance of a configuration $\mathtt{c}$ of $\mathcal{A}_i$ (denoted as $\mathrm{dist}_{\mathcal{A}_i}(\mathtt{c})$) is the length of a minimal word that takes you from $\mathtt{c}$ to a configuration $(\mathbf{x}, p, m)$ for some $m < |\mathcal{B}|$ and $p \in C_i$ such that $\mathbf{x} \in \overline{\mathcal{W}}_i^{p,m}$. Given two configurations $\mathtt{c}$, $\mathtt{d}$ of $\mathcal{A}_1$ and $\mathcal{A}_2$ respectively, if $\mathrm{dist}_{\mathcal{A}_1}(\mathtt{c}) \neq \mathrm{dist}_{\mathcal{A}_2}(\mathtt{d})$, then $\mathtt{c} \not\equiv \mathtt{d}$.

By special word lemma (Lemma 14), the lexicographically minimal reachability witness has a special form. This is used to show that if a configuration $\mathtt{c}$ of an ODCA $\mathcal{A}$ has finite distance, then $\mathrm{dist}_{\mathcal{A}}(\mathtt{c}) = \frac{a}{b}\textsc{counter-value}(\mathtt{c}) + t$, where $a, b, t \in \mathbb{N}$ and are polynomially bounded in $|\mathcal{A}|$. This helps us in proving that configuration pairs outside the initial space having equal distance lie inside a belt. Therefore, the background space points either have unequal or infinite distances. Similar to that in [3], we can show that the length of the run Π in the background space is polynomially bounded in $|\mathcal{A}_1|, |\mathcal{A}_2|$, and the counter values of the first background point in Π. The following lemma bounds the counter values of the first configuration pair in the background space, if it exists, during the run Π.

▶ **Lemma 17.** *If $\mathtt{h}_j = \langle \mathtt{c}_j, \mathtt{d}_j \rangle$ is the first background point in Π then, $\textsc{counter-value}(\mathtt{c}_j)$ and $\textsc{counter-value}(\mathtt{d}_j)$ are polynomially bounded in $|\mathcal{A}_1|$ and $|\mathcal{A}_2|$.*

Proof sketch. Since $\mathtt{h}_j = \langle \mathtt{c}_j, \mathtt{d}_j \rangle$ is a background point in Π, either $\mathrm{dist}_{\mathcal{A}_1}(\mathtt{c}_j) \neq \mathrm{dist}_{\mathcal{A}_2}(\mathtt{d}_j)$ or $\mathrm{dist}_{\mathcal{A}_1}(\mathtt{c}_j) = \mathrm{dist}_{\mathcal{A}_2}(\mathtt{d}_j) = \infty$. Assume for contradiction that the counter values of $\mathtt{h}_j$ are not polynomially bounded.

Consider the case where $\mathrm{dist}_{\mathcal{A}_1}(\mathtt{c}_j) \neq \mathrm{dist}_{\mathcal{A}_2}(\mathtt{d}_j)$. Without loss of generality, we assume $\mathrm{dist}_{\mathcal{A}_1}(\mathtt{c}_j) < \mathrm{dist}_{\mathcal{A}_2}(\mathtt{d}_j)$. Therefore there exists a configuration pair $\langle (\mathbf{x}, p, m), (\mathbf{y}, q, n) \rangle$ in Π such that $m < |\mathcal{B}|$ and $\mathbf{x} \in \overline{\mathcal{W}}_1^{p,m}$. Using an argument similar to the one used to prove the pseudo-pumping lemma (Lemma 10), we show that we can pump out some portion from the sub-run inside the belt to reach a configuration pair in the background space with unequal distance and smaller counter values.

In the case where $\mathrm{dist}_{\mathcal{A}_1}(\mathtt{c}_j) = \mathrm{dist}_{\mathcal{A}_2}(\mathtt{d}_j) = \infty$, we can show that $\mathtt{c}_j \not\equiv_{|\mathcal{B}|} \mathtt{d}_j$. We can then apply Lemma 12 to show the existence of a shorter run, which enters the background space at a point $\mathtt{h}_{j'} = \langle \mathtt{c}_{j'}, \mathtt{d}_{j'} \rangle$ with smaller counter values such that $\mathtt{c}_{j'} \not\equiv_{|\mathcal{B}|} \mathtt{d}_{j'}$. ◀

Proof of Lemma 16. Consider the run Π. We have shown that the counter values of configuration pairs inside the belt space during this run and that of the first background point, if it exists, are polynomially bounded in $|\mathcal{A}_1|$ and $|\mathcal{A}_2|$. We also proved that the length of Π after the first background point is polynomially bounded. Since the counter values of configuration pairs inside the initial space are also polynomially bounded, we get that the maximum counter value in Π is polynomially bounded in $|\mathcal{A}_1|$ and $|\mathcal{A}_2|$. ◀

5 Non-deterministic ODCA

In this section, we consider the counter-determinacy restriction over weightless OCAs (equivalently, with weights from the boolean semiring). These results do not follow from previous sections, as booleans are not a field.

▶ **Example 18.** *The following languages are defined over the alphabet $\Sigma = \{a, b\}$ and are recognised by non-deterministic OCA with counter-determinacy.*

(a) *The language $\mathcal{L}_1 = \{a^n ba^n \mid n > 0\}$.*
(b) *The language $\mathcal{L}_2 = \{(a+b)^* \mid$ number of a's is greater than number of b's$\}$.*
(c) *The language $\mathcal{L}_3 = \{a^n(b+c)^m b(b+c)^k \mid m, n \in \mathbb{N}$ and $m > n\}$.*

Note that none of the above languages are definable by visibly pushdown automata.

We observe that the relationship between non-deterministic and deterministic ODCAs is similar to that between non-deterministic and deterministic finite automata. By definition, deterministic ODCAs have at most one unique path for any fixed word. Therefore, they are deterministic OCAs with counter-determinacy. It is also easy to observe that deterministic OCAs are deterministic ODCAs. It follows that deterministic ODCAs and deterministic OCAs are expressively equivalent. Similar to non-deterministic finite automata, we observe that non-deterministic ODCAs can be determinised by a subset construction of the states of the finite state machine. However, this results in an exponential blow-up. In Example 18, the deterministic ODCA that recognises the language $\mathcal{L}_3$ has to check whether every b encountered after reading the word $a^n(b+c)^{n+1}$ is at the k^{th} position from the end. This will require at least $\mathcal{O}(2^k)$ states. On the other hand, there is a non-deterministic ODCA with $\mathcal{O}(k)$ states recognising the same language. Similar to finite automata, non-deterministic ODCAs are a "succinct" way to represent deterministic OCAs.

▶ **Theorem 19** (Determinisation). *Given a non-deterministic ODCA, a polynomial space machine can output an equivalent deterministic ODCA of exponential size.*

The idea in proving the above theorem is a simple subset construction. The above result and the fact that equivalence of deterministic ODCA is in NL gives us the upper bound in the following theorem. The lower bound follows from that of NFAs [16].

▶ **Theorem 20.** *The equivalence problem for non-deterministic ODCA is* PSPACE*-complete.*

The equivalence of non-deterministic OCA is undecidable [24]. Our theorem shows that undecidability is due to non-determinism in the component that modifies the counter.

6 Conclusion

We introduced a new model called one-deterministic-counter automata. The model "separates" the machine into two components, (1) counter structure - that can modify the counter, and (2) finite state machine - that can access the counter. This separation of the "writing" and "reading" part gives some natural advantages to the model. We show that the equivalence problem for weighted ODCA is in P if the weights are from a field while that of non-deterministic ODCA is in PSPACE. Note that the equivalence problems on weighted automata (where weights are from a field) and non-deterministic finite automata are in P and PSPACE respectively. On the other hand, the equivalence problem for non-deterministic OCA is undecidable and that of weighted OCA (weights from a field) is not-known. It will be interesting to look at other models where we can separate the "writing" and the "reading" parts. For example, a natural extension is to consider *stack-deterministic* pushdown automata - where a deterministic machine updates the stack. We also leave open the question of learning of weighted ODCAs.

References

1 Steven P. Abney, David A. McAllester, and Fernando Pereira. Relating probabilistic grammars and automata. In Robert Dale and Kenneth Ward Church, editors, *27th Annual Meeting of the Association for Computational Linguistics, University of Maryland, College Park, Maryland, USA, 20-26 June 1999*, pages 542–549. ACL, 1999. doi:10.3115/1034678.1034759.

2 Rajeev Alur and P. Madhusudan. Visibly pushdown languages. In László Babai, editor, *Proceedings of the 36th Annual ACM Symposium on Theory of Computing, Chicago, IL, USA, June 13-16, 2004*, pages 202–211. ACM, 2004. doi:10.1145/1007352.1007390.

3 Stanislav Böhm and Stefan Göller. Language equivalence of deterministic real-time one-counter automata is NL-Complete. In Filip Murlak and Piotr Sankowski, editors, *MFCS*, volume 6907 of *Lecture Notes in Computer Science*, pages 194–205. Springer, 2011.

4 Stanislav Böhm, Stefan Göller, and Petr Jancar. Bisimilarity of one-counter processes is PSPACE-Complete. In Paul Gastin and François Laroussinie, editors, *CONCUR*, volume 6269 of *Lecture Notes in Computer Science*, pages 177–191. Springer, 2010.

5 Stanislav Böhm, Stefan Göller, and Petr Jancar. Equivalence of deterministic one-counter automata is NL-Complete. In Dan Boneh, Tim Roughgarden, and Joan Feigenbaum, editors, *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*, pages 131–140. ACM, 2013. doi:10.1145/2488608.2488626.

6 Stanislav Böhm, Stefan Göller, and Petr Jancar. Bisimulation equivalence and regularity for real-time one-counter automata. *J. Comput. Syst. Sci*, 80(4):720–743, 2014.

7 Tomás Brázdil, Javier Esparza, Stefan Kiefer, and Antonín Kucera. Analyzing probabilistic pushdown automata. *Formal Methods Syst. Des*, 43(2):124–163, 2013.

8 Tomáš Brázdil, Antonín Kučera, and Oldřich Stražovský. On the decidability of temporal properties of probabilistic pushdown automata. In Volker Diekert and Bruno Durand, editors, *STACS 2005*, pages 145–157, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

9 Vojtech Forejt, Petr Jancar, Stefan Kiefer, and James Worrell. Bisimilarity of probabilistic pushdown automata. In Deepak D'Souza, Telikepalli Kavitha, and Jaikumar Radhakrishnan, editors, *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2012, December 15-17, 2012, Hyderabad, India*, volume 18 of *LIPIcs*, pages 448–460. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2012. doi:10.4230/LIPIcs.FSTTCS.2012.448.

10 Vojtech Forejt, Petr Jancar, Stefan Kiefer, and James Worrell. Language equivalence of probabilistic pushdown automata. *Inf. Comput*, 237:1–11, 2014.

11 Juraj Hromkovič and Georg Schnitger. On probabilistic pushdown automata. *Information and Computation*, 208(8):982–995, 2010. doi:10.1016/j.ic.2009.11.001.

12 Stefan Kiefer, Andrzej S. Murawski, Joël Ouaknine, Björn Wachter, and James Worrell. On the complexity of equivalence and minimisation for q-weighted automata. *Log. Methods Comput. Sci.*, 9(1), 2013. doi:10.2168/LMCS-9(1:8)2013.

13 Daniel Krob. The equality problem for rational series with multiplicities in the tropical semiring is undecidable. In *Proceedings of the 19th International Colloquium on Automata, Languages and Programming*, ICALP '92, pages 101–112, Berlin, Heidelberg, 1992. Springer-Verlag.

14 Antonín Kucera, Javier Esparza, and Richard Mayr. Model checking probabilistic pushdown automata. *Log. Methods Comput. Sci.*, 2(1), 2006. `doi:10.2168/LMCS-2(1:2)2006`.

15 Antonín Kučera. Methods for quantitative analysis of probabilistic pushdown automata. *Electronic Notes in Theoretical Computer Science*, 149(1):3–15, 2006. Proceedings of the 7th International Workshop on Verification of Infinite-State Systems (INFINITY 2005). `doi:10.1016/j.entcs.2005.11.013`.

16 Meyer and Stockmeyer. Word problems requiring exponential time. In *STOC: ACM Symposium on Theory of Computing (STOC)*, 1973.

17 Dirk Nowotka and Jirí Srba. Height-deterministic pushdown automata. In Ludek Kucera and Antonín Kucera, editors, *MFCS*, volume 4708 of *Lecture Notes in Computer Science*, pages 125–134. Springer, 2007.

18 William Ogden. A helpful result for proving inherent ambiguity. *Mathematical Systems Theory*, 2(3):191–194, 1968.

19 Federico Olmedo, Benjamin Lucien Kaminski, Joost-Pieter Katoen, and Christoph Matheja. Reasoning about recursive probabilistic programs. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science*, LICS '16, pages 672–681, New York, NY, USA, 2016. Association for Computing Machinery. `doi:10.1145/2933575.2935317`.

20 M. P. Schützenberger. On the definition of a family of automata. *Information and Control*, 4(2–3):245–270, September 1961.

21 Géraud Sénizergues. The equivalence problem for deterministic pushdown automata is decidable. In Pierpaolo Degano, Roberto Gorrieri, and Alberto Marchetti-Spaccamela, editors, *Automata, Languages and Programming*, pages 671–681, Berlin, Heidelberg, 1997. Springer Berlin Heidelberg.

22 Stirling. Deciding DPDA equivalence is primitive recursive. In *ICALP: Annual International Colloquium on Automata, Languages and Programming*, 2002.

23 Wen-Guey Tzeng. A polynomial-time algorithm for the equivalence of probabilistic automata. *SIAM J. Comput*, 21(2):216–227, 1992.

24 Valiant and Paterson. Deterministic one-counter automata. *JCSS: Journal of Computer and System Sciences*, 10, 1975.

A Section 3. Reachability problems of weighted ODCA

Given a weighted automaton $\mathcal{B}$ over a field $\mathcal{F}$, with k states, an initial configuration $\bar{\mathsf{c}}$, a vector space $\mathcal{U} \subseteq \mathcal{F}^k$ and a set of counter states S, the co-VS reachability problem asks whether there exists a run $\bar{\mathsf{c}} \xrightarrow{*} \overline{\mathcal{U}} \times S$.

▶ **Theorem 21.** *There is a polynomial time algorithm that decides the co-VS reachability problem for weighted automata and outputs a minimal reachability witness if it exists.*

Proof. The idea of equivalence checking of weighted automata goes back to the seminal paper by Schützenberger [20]. Tzeng [23] provided a polynomial time algorithm for the equivalence of two probabilistic automata by reducing the problem to the co-VS reachability problem where $\mathcal{V} = \{\mathbf{0}\}$. The same algorithm can be modified to solve the general co-VS reachability problem of weighted automata. ◀

▶ **Lemma 9.** *Let $z \in \Sigma^*$ be a minimal witness for $(\mathsf{c}, \overline{\mathcal{V}}, S, X)$. If the number of distinct counter values encountered during the run $\mathsf{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times X$ is t, then $|z| \leq t \cdot K$.*

Proof. Let $\mathsf{c} = \mathsf{c}_1$ and $T(\mathsf{c}_1) = \mathsf{c}_1\tau_1\mathsf{c}_2 \cdots \tau_{h-1}\mathsf{c}_h$ be the run on word z from c_1 and T the corresponding sequence of transitions. Let t be the number of distinct counter values encountered during this run. Now assume for contradiction that $h > |Q| \cdot |C| \cdot t$, then by Pigeonhole principle, there are $|Q| + 1$ many configurations $\mathsf{c}_{i_0}, \mathsf{c}_{i_1}, \ldots, \mathsf{c}_{i_{|Q|}}$ with the same counter state and counter value during this run. Given a configuration $\mathtt{c}$, let $\mathbf{x}_{\mathtt{c}}$ denote WEIGHT-VECTOR($\mathtt{c}$). Let $\mathbb{A}_j$ denote the matrix such that $\mathbf{x}_{\mathtt{c}_{i_j}}\mathbb{A}_j = \mathbf{x}_{\mathtt{c}_h}$ for all $j \in [0, |Q|]$. Using linear algebra, we get that there exists $r \leq |Q|$, and $t \in [0, r-1]$ such that $\mathbf{x}_{\mathtt{c}_{i_t}}\mathbb{A}_r \in \overline{\mathcal{V}}$. Consider the sequence of transitions $T' = \tau_{1 \cdots i_t}\tau_{r \cdots \ell-1}$ and $v = \mathtt{word}(T')$. The run $\pi(v, \mathtt{c}_1) = T'(\mathtt{c}_1)$ is a run since configurations $\mathtt{c}_t$ and $\mathtt{c}_r$ have the same counter state and counter value. This is a shorter run than $\pi(z, \mathtt{c}_1)$ and $\mathtt{c}_1 \xrightarrow{v} \overline{\mathcal{V}} \times S \times X$. This contradicts the minimality of z. ◀

▶ **Corollary 11.** *If $z \in \Sigma^*$ is a minimal witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, \{m\})$, then*
1. *the maximum counter value encountered during the run $\mathtt{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times \{m\}$ is less than $max(\text{COUNTER-VALUE}(\mathtt{c}), m) + K^2$, and*
2. *$|z| \leq K^3 + max(\text{COUNTER-VALUE}(\mathtt{c}), m) \cdot K$.*

Proof. Let $z \in \Sigma^*$ be a minimal reachability witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, \{m\})$, where $\mathtt{c}$ is a configuration with counter value n.

1. Consider the run of word z from $\mathtt{c}$. Let $\mathtt{d} \in \overline{\mathcal{V}} \times S \times \{m\}$ such that $\mathtt{c} \xrightarrow{z} \mathtt{d}$. Assume for contradiction that the maximum counter value encountered during the run $\mathtt{c} \xrightarrow{z} \mathtt{d}$ is greater than $max(n, m) + (|Q| \cdot |C|)^2$. Let $\mathtt{e}_1, \mathtt{e}_2, \cdots, \mathtt{e}_t$ be all the configurations in this run such that their counter values are zero. There exists words $u_1, u_2, \cdots, u_{t+1} \in \Sigma^*$ such that $z = u_1 u_2 \cdots u_{t+1}$ and $\mathtt{c} \xrightarrow{u_1} \mathtt{e}_1 \xrightarrow{u_2} \mathtt{e}_2 \xrightarrow{u_3} \cdots \xrightarrow{u_t} \mathtt{e}_t \xrightarrow{u_{t+1}} \mathtt{d}$. Note that $\mathtt{c} \xrightarrow{u_1} \mathtt{e}_1$, $\mathtt{e}_t \xrightarrow{u_{t+1}} \mathtt{d}$ and $\mathtt{e}_i \xrightarrow{u_{i+1}} \mathtt{e}_{i+1}$ for all $i \in [1, t-1]$ are floating runs (refer Figure 6).

We show that the counter values are bounded during these floating runs. First, we consider the floating run $\mathtt{c} \xrightarrow{u_1} \mathtt{e}_1$. Given a configuration $\mathtt{c}$, we use $\mathbf{x}_{\mathtt{c}}$ to denote WEIGHT-VECTOR($\mathtt{c}$). Let $\mathbb{A} \in \mathcal{F}^{|Q| \times |Q|}$ be such that $\mathbf{x}_{\mathtt{d}} = \mathbf{x}_{\mathtt{e}_1}\mathbb{A}$. The set $\mathcal{U} = \{\mathbf{y} \in \mathcal{F}^{|Q|} \mid \mathbf{y}\mathbb{A} \in \mathcal{V}\}$ is a vector space and hence the vector $\mathbf{x}_{\mathtt{e}_1} \in \overline{\mathcal{U}}$. From Observation 8, we know that u_1 is a minimal reachability witness for $(\mathtt{c}, \overline{\mathcal{U}}, \{p_{\mathtt{e}_1}\}, \{0\})$ and therefore by Lemma 10 we know that the maximum counter value encountered during the run $\pi(u_1, \mathtt{c})$ is less than $n + (|Q| \cdot |C|)^2$.

Similarly for the floating run $\mathtt{e}_t \xrightarrow{u_{t+1}} \mathtt{d}$, the maximum counter value is bounded by $m + (|Q| \cdot |C|)^2$. Now consider the floating runs $\mathtt{e}_i \xrightarrow{u_{i+1}} \mathtt{e}_{i+1}$ for all $i \in [1, t-1]$. Again by applying Lemma 10 we get that the maximum counter value encountered during these sub-runs is less than $(|Q| \cdot |C|)^2$. Therefore, the maximum counter value encountered during the run $\mathtt{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times \{m\}$ is less than $max(n, m) + (|Q| \cdot |C|)^2$.

2. From the previous point, we know that the maximum counter value encountered during the run $\mathtt{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times \{m\}$ is less than $max(n, m) + (|Q| \cdot |C|)^2$. Therefore, there are at most $max(n, m) + (|Q| \cdot |C|)^2$ many distinct counter values encountered during this run. Now from Lemma 9 we get that $|z| \leq (|Q| \cdot |C|) \cdot (max(n, m) + (|Q| \cdot |C|)^2)$. ◀

▶ **Lemma 12** (cut lemma)**.** *Let $z \in \Sigma^*$ be a witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, \mathbb{N})$, where $\mathtt{c}$ is a configuration with COUNTER-VALUE($\mathtt{c}$) $= n$ for some $n \in \mathbb{N}$, and $\mathtt{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times \{m\}$ is a floating run for some $m \in \mathbb{N}$. If $m - n > K$, then there exists $z_{sub} \in \Sigma^*$ such that z_{sub} is a subword of z, $\mathtt{c} \xrightarrow{z_{sub}} \overline{\mathcal{V}} \times S \times \{m'\}$ is a floating run and $m' - n < m - n$.*

Proof. Let $z \in \Sigma^*$ be a witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, \mathbb{N})$ and $\mathtt{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times \{m\}$ is a floating run. Let n be the counter value of configuration $\mathtt{c}$ and $m > n + |Q| \cdot |C|$. Let $\mathtt{c}_1 = \mathtt{c}$ and $\pi(z, \mathtt{c}_1) = \mathtt{c}_1\tau_1\mathtt{c}_2 \cdots \tau_{\ell-1}\mathtt{c}_\ell$ be such that configuration $\mathtt{c}_\ell$ has counter value m. Consider the sequence of transitions $T = \tau_0\tau_1 \cdots \tau_{\ell-1}$ in $\pi(z, \mathtt{c}_1)$.

Since there are only $|C|$ counter states, by the Pigeonhole principle, there exists a strictly increasing sequence $I = 0 < i_0 < i_1 < \cdots < i_{|Q|} \leq \ell$ such that for all $j, j' \in I$ (Condition 1) COUNTER-STATE($\mathtt{c}_j$) $=$ COUNTER-STATE($\mathtt{c}_{j'}$) and (Condition 2) if $j < j'$ then for all $d \in [j+1, j'-1]$, COUNTER-VALUE($\mathtt{c}_j$) $<$ COUNTER-VALUE($\mathtt{c}_d$) $<$ COUNTER-VALUE($\mathtt{c}_{j'}$)

and COUNTER-VALUE($\mathtt{c}_j$) < COUNTER-VALUE($\mathtt{c}_{j'}$). Given a configuration $\mathtt{c}$, let $\mathbf{x}_{\mathtt{c}}$ denote WEIGHT-VECTOR($\mathtt{c}$). Consider the set of configurations $\mathtt{c}_{i_0}, \mathtt{c}_{i_1}, \ldots, \mathtt{c}_{i_{|Q|}}$. For any $j \in [0, |Q|]$, let $\mathbb{A}_j$ denote the matrix such that $\mathbf{x}_{\mathtt{c}_{i_j}} \mathbb{A}_j = \mathbf{x}_{\mathtt{c}_\ell}$. Since $\mathbf{x}_{\mathtt{c}_{i_d}} \mathbb{A}_d \in \overline{\mathcal{V}}$ for all $d \in [0, |Q|]$, using linear algebra, we get that there exists $l, k \in [0, |Q|]$ with $l < k$ such that $\mathbf{x}_{\mathtt{c}_{i_l}} \mathbb{A}_k \in \overline{\mathcal{V}}$. Consider a configuration $\mathtt{e} = (\mathbf{x}, p, n)$. If $\pi(u, \mathtt{e})$ is a floating run with $min_{\mathtt{ce}}(\pi(u, \mathtt{e})) > 0$, then for all $m \in \mathbb{N}$ and $\mathbf{y} \in \mathcal{F}^{|Q|}$, $\pi(u, (\mathbf{y}, p, m))$ is a run. Consider the sequence of transitions $T' = \tau_{i_k \cdots \ell-1}$ and let $u = \mathtt{word}(T')$. Because of Condition 2, $min_{\mathtt{ce}}(\pi(u, \mathtt{c}_{i_k})) > 0$. Therefore the run $T''(\mathtt{c}_1)$ where $T'' = \tau_{1 \cdots i_l - 1} \tau_{i_k \cdots \ell-1}$ is a run shorter than $\pi(z, \mathtt{c}_1)$ with smaller counter effect. ◀

Now we show that for any run (need not be floating) of a minimal coverability witness z for $(\mathtt{c}, \overline{\mathcal{V}}, S, \mathbb{N})$, the maximum counter value encountered during the run $\mathtt{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times \mathbb{N}$ is polynomially bounded in the number of states of the machine and the initial counter value.

▶ **Corollary 22.** *If $z \in \Sigma^*$ is a minimal witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, \mathbb{N})$, where $\mathtt{c}$ is a configuration with counter value n, then the maximum counter value encountered during the run $\mathtt{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times \mathbb{N}$ is less than $max(n, |Q| \cdot |C|) + (|Q| \cdot |C|)^2$.*

Proof. Let $z \in \Sigma^*$ be a minimal witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, \mathbb{N})$, where $\mathtt{c}$ is a configuration with counter value n. Consider the run of word z from $\mathtt{c}$. Let $\mathtt{d} \in \overline{\mathcal{V}} \times S \times \mathbb{N}$ such that $\mathtt{c} \xrightarrow{z} \mathtt{d}$. If $\mathtt{c} \xrightarrow{z} \mathtt{d}$ is a floating run, then by Lemma 12 the maximum counter value encountered during this run will be less than $n + |Q| \cdot |C|$. Now if $\mathtt{c} \xrightarrow{z} \mathtt{d}$ is not a floating run, then there exists $u_1, u_2 \in \Sigma^*$ such that $z = u_1 u_2$ and $\mathtt{c} \xrightarrow{u_1} \mathtt{e} \xrightarrow{u_2} \mathtt{d}$ where, counter value of configuration $\mathtt{e}$ is zero and $\mathtt{e} \xrightarrow{u_2} \mathtt{d}$ is a floating run.

Given a configuration $\mathtt{c}$, let $\mathbf{x}_{\mathtt{c}}$ denote WEIGHT-VECTOR($\mathtt{c}$). Let $\mathbb{A} \in \mathcal{F}^{|Q| \times |Q|}$ be such that $\mathbf{x}_{\mathtt{d}} = \mathbf{x}_{\mathtt{e}} \mathbb{A}$. The set $\mathcal{U} = \{\mathbf{y} \in \mathcal{F}^{|Q|} \mid \mathbf{y}\mathbb{A} \in \mathcal{V}\}$ is a vector space and hence the vector $\mathbf{x}_{\mathtt{e}} \in \overline{\mathcal{U}}$. Note that for all $\mathbf{y} \in \overline{\mathcal{U}}$, the vector $\mathbf{y}\mathbb{A} \in \overline{\mathcal{V}}$. From Observation 8, we know that u_1 is a minimal reachability witness for $(\mathtt{c}, \overline{\mathcal{U}}, \{p_{\mathtt{e}}\}, \{0\})$, where $p_{\mathtt{e}}$ is the counter state of configuration $\mathtt{e}$, and therefore by Corollary 11, we know that the maximum counter value encountered during the run $\pi(u_1, \mathtt{c})$ is less than $n + (|Q| \cdot |C|)^2$. Now since $\mathtt{e} \xrightarrow{u_2} \mathtt{d}$ is a floating run and u_2 is the minimal such word, from Lemma 12, we get that the counter value of configuration $\mathtt{d}$ is less than or equal to $|Q| \cdot |C|$, and by Lemma 10, we know that the maximum counter value encountered during this run is less than $|Q| \cdot |C| + (|Q| \cdot |C|)^2$. Therefore, we get that the maximum counter value encountered during the run $\mathtt{c} \xrightarrow{z} \mathtt{d}$ is less than $max(n, |Q| \cdot |C|) + (|Q| \cdot |C|)^2$. ◀

Our next objective is to show that the counter values are polynomially bounded during the run of a minimal coverability witness.

▶ **Corollary 23.** *Let $\mathtt{c}$ be a configuration with counter value n. If z is a minimal witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, \mathbb{N})$ then $|z| \leq (|Q| \cdot |C|) \cdot (max(n, (|Q| \cdot |C|)) + (|Q| \cdot |C|)^2)$.*

Proof. Let $z \in \Sigma^*$ be a minimal reachability witness for $(\mathtt{c}, \overline{\mathcal{V}}, S, \mathbb{N})$. From Corollary 22, we know that the maximum counter value encountered during the run $\mathtt{c} \xrightarrow{z} \overline{\mathcal{V}} \times S \times \mathbb{N}$ is less than $max(n, (|Q| \cdot |C|)) + (|Q| \cdot |C|)^2$. Therefore, there are at most $max(n, (|Q| \cdot |C|)) + (|Q| \cdot |C|)^2$ many distinct counter values encountered during this run. Now from Lemma 9 we get that $|z| \leq (|Q| \cdot |C|) \cdot (max(n, (|Q| \cdot |C|)) + (|Q| \cdot |C|)^2)$. ◀

▶ **Theorem 13.** *The co-VS reachability and co-VS coverability problems for weighted ODCA can be decided in polynomial time when the counter values are given in unary notation.*

Proof. Assume we are given a weighted ODCA $\mathcal{A} = ((C, \delta_0, \delta_1, p_0), (Q, \boldsymbol{\lambda}, \Delta, \boldsymbol{\eta}))$, initial configuration $\mathsf{c} = (\mathbf{x}, p, n)$, vector space $\mathcal{V}$, set of counter states S and counter value m as inputs for the co-VS reachability problem. For solving this reachability problem, we first consider the $max(n,m) + (|Q| \cdot |C|)^2$-unfolding weighted automaton $\mathcal{A}^{max(n,m)+(|Q|\cdot|C|)^2} = (C', \delta', p'_0;\ Q', \boldsymbol{\lambda}', \Delta', \boldsymbol{\eta}'_F)$ of $\mathcal{A}$. From Corollary 11, we know that the maximum counter value encountered during the run of the minimal reachability witness z for $(\mathsf{c}, \overline{\mathcal{V}}, S, \{m\})$ is less than $max(n,m) + (|Q| \cdot |C|)^2$. We define a vector space $\mathcal{U} \subseteq \mathcal{F}^{|Q'|}$ as follows: A vector $\mathbf{z} \in \mathcal{F}^{|Q'|}$ is in $\mathcal{U}$ if there exists $\mathbf{y} \in \mathcal{V}$ such that for all $i \in [0, |Q|-1]$, $\mathbf{z}[|Q| \cdot m + i] = \mathbf{y}[i]$ and for all $m' \neq m$ and $i \in [0, |Q|-1]$, $\mathbf{z}[|Q| \cdot m' + i] = 0$.

Given a configuration $\mathsf{c} = (\mathbf{x}, p, n)$ of a weighted ODCA, we define the vector $\mathbf{z}_\mathsf{c} \in \mathcal{F}^{|Q'|}$.

$$\mathbf{z}_\mathsf{c}[i] = \begin{cases} \mathbf{x}[i \bmod |Q|], & \text{if } \frac{i}{|Q|} = n \\ 0, & \text{otherwise} \end{cases}$$

Now, consider the configuration $\bar{\mathsf{c}} = (\mathbf{z}_\mathsf{c}, (p, n))$ of $\mathcal{A}^{max(n,m)+(|Q|\cdot|C|)^2}$ and check whether $\bar{\mathsf{c}} \xrightarrow{*} \overline{\mathcal{U}} \times S \times \{0\}$. This is a co-VS reachability problem of weighted automata. Using Theorem 21, this can be solved in polynomial time.

For solving co-VS coverability problem when a weighted ODCA $\mathcal{A}$ with an initial configuration $\mathsf{c} = (\mathbf{z}, p, n)$, a vector space $\mathcal{V}$ and a set of counter states S are given as inputs, we consider the $max(n, (|Q| \cdot |C|)) + (|Q| \cdot |C|)^2$-unfolding weighted automaton $\mathcal{A}^{max(n,(|Q|\cdot|C|))+(|Q|\cdot|C|)^2} = (C', \delta', p'_0;\ Q', \boldsymbol{\lambda}', \Delta', \boldsymbol{\eta}'_F)$ of $\mathcal{A}$. From Corollary 22, we know that the maximum counter value encountered during the run of a minimal reachability witness z for $(\mathsf{c}, \overline{\mathcal{V}}, S, \mathbb{N})$ is less than $max(n, (|Q| \cdot |C|)) + (|Q| \cdot |C|)^2$. We define a vector space $\mathcal{U} \subseteq \mathcal{F}^{|Q'|}$ as follows: A vector $\mathbf{x} \in \mathcal{F}^{|Q'|}$ is in $\mathcal{U}$ if there exists $\mathbf{y} \in \mathcal{V}$ and $m \in \mathbb{N}$ such that for all $i \in [0, |Q|-1]$, $\mathbf{x}[|Q| \cdot m + i] = \mathbf{y}[i]$ and for all $m' \neq m$ and $i \in [0, |Q|-1]$, $\mathbf{x}[|Q| \cdot m' + i] = 0$. Given a configuration $\mathsf{c} = (\mathbf{x}, p, n)$ of a weighted ODCA, we define the vector $\mathbf{z}_\mathsf{c} \in \mathcal{F}^{|Q'|}$.

$$\mathbf{z}_\mathsf{c}[i] = \begin{cases} \mathbf{x}[i \bmod |Q|], & \text{if } \frac{i}{|Q|} = n \\ 0, & \text{otherwise} \end{cases}$$

Now, consider the configuration $\bar{\mathsf{c}} = (\mathbf{z}_\mathsf{c}, (p, n))$ of $\mathcal{A}^{max(n,(|Q|\cdot|C|))+(|Q|\cdot|C|)^2}$ and check whether $\bar{\mathsf{c}} \xrightarrow{*} \overline{\mathcal{U}} \times S \times \{0\}$. This is a co-VS reachability problem of a weighted automaton. From Theorem 21, we know that this can be solved in polynomial time. ◀

▶ **Lemma 15** (special-word lemma). *If $z \in \Sigma^*$ is the lexicographically minimal witness for $(\mathsf{c}, \overline{\mathcal{V}}, S, \{m\})$, then there exists $u, y_1, v_1, v_2, v_3, y_2, w \in \Sigma^*$ and $r_1, r_2 \in \mathbb{N}$ such that $z = uy_1^{r_1} v y_2^{r_2} w$ and the following are true:*

1. *$|uy_1vy_2w|$ is polynomially bounded in the number of states of the machine.*
2. *r_1 and r_2 are polynomially bounded in the number of states of the machine, m, and COUNTER-VALUE(c).*

Proof. Let $z \in \Sigma^*$ be the lexicographically minimal reachability witness for $(\mathsf{c}, \overline{\mathcal{V}}, S, \{m\})$, where c is a configuration with counter value n. Consider the run of word z from c. Let $\mathsf{d} \in \overline{\mathcal{V}} \times S \times \{m\}$ such that $\mathsf{c} \xrightarrow{z} \mathsf{d}$. Let $\mathsf{c} = \mathsf{c}_1$ and $T(\mathsf{c}_1) = \mathsf{c}_1\tau_1\mathsf{c}_2 \cdots \tau_{\ell-1}\mathsf{c}_\ell$ denote the run on word z from the configuration c_1 and T the corresponding sequence of transitions. Let e_1 be the first configuration with counter value zero and e_2 be the last configuration with counter value zero during this run. Let $z_1, z_2, z_3 \in \Sigma^*$ be such that $\mathsf{c} \xrightarrow{z_1} \mathsf{e}_1 \xrightarrow{z_2} \mathsf{e}_2 \xrightarrow{z_3} \mathsf{c}_\ell$ and $z = z_1z_2z_3$. Observe that $\mathsf{c} \xrightarrow{z_1} \mathsf{e}_1$ and $\mathsf{e}_2 \xrightarrow{z_3} \mathsf{c}_\ell$ are floating runs.

From Lemma 14, we know that there exists $u_1, u_3, v_1, v_3, y_1, y_3 \in \Sigma^*$ and $r_1, r_3 \in \mathbb{N}$ such that $z_1 = u_1 y_1^{r_1} v_1$, $z_3 = u_3 y_3^{r_3} v_3$, $|u_1 y_1 v_1 u_3 y_3 v_2| \leq 2 \cdot p_1(|Q| \cdot |C|)$, $r_1 \leq p_2(|Q| \cdot |C|, n)$, and $r_3 \leq p_2(|Q| \cdot |C|, m)$. Also, from Corollary 11 we get that $|z_2| \leq (|Q| \cdot |C|)^3$. ◀

B Regularity of ODCA is in P

We say that a weighted ODCA $\mathcal{A}$ is regular if there is a weighted automaton $\mathcal{B}$ that is equivalent to it. We look at the regularity problem - the problem of deciding whether a weighted ODCA is regular. We fix a weighted ODCA $\mathcal{A} = ((C, \delta_0, \delta_1, p_0), (Q, \boldsymbol{\lambda}, \Delta, \boldsymbol{\eta}))$ and use N to denote $|C| \cdot |Q|$. The proof technique is adapted from the ideas developed by Böhm et al. [6] in the context of real-time OCA. The crucial idea in proving regularity is to check for the existence of infinitely many equivalence classes. The proof relies on the notion of distance of configurations. Distance of a configuration is the length of a minimal word to be read to reach a configuration that does not have an N equivalent configuration in the underlying weighted automaton. The challenge is to find infinitely many configurations reachable from the initial configuration, so that no two of them have same distance.

▶ **Theorem 4.** *The regularity problem of weighted ODCA (weights from a field) is in* P.

Proof. Let $\mathcal{A}$ be a weighted ODCA and c_0 be its initial configuration. Lemma 24 shows that if $\mathcal{A}$ is not regular, then there are words $u, v \in \Sigma^*$ and configurations d, e such that there is a run of the form $\mathsf{c}_0 \xrightarrow{u} \mathsf{d} \xrightarrow{v} \mathsf{e}$ such that $\mathsf{N}^2 + \mathsf{N} \leq \text{COUNTER-VALUE}(\mathsf{d}) \leq 2\mathsf{N}^2 + \mathsf{N}$, $\text{WEIGHT-VECTOR}(\mathsf{e}) \in \overline{\mathcal{W}}^{\text{COUNTER-STATE}(\mathsf{e}), \text{COUNTER-VALUE}(\mathsf{e})}$ with $\text{COUNTER-VALUE}(\mathsf{e}) < \mathsf{N}$. The existence of such words u and v can be decided in polynomial time since the minimal length of such a path if it exists, is polynomially bounded in the number of states of the weighted ODCA by Corollary 11. This concludes the proof. ◀

We use c to denote some configuration of $\mathcal{A}$ and $\bar{\mathsf{c}}$ to denote some configuration of $\mathrm{U}(\mathcal{A})$. For a $p \in C$ and $m \in \mathbb{N}$, we define $\mathcal{W}^{p,m} = \{\mathbf{x} \in \mathcal{F}^{|Q|} \mid \exists \bar{\mathsf{c}} \in \mathcal{F}^{\mathsf{N}}, \mathsf{c} = (\mathbf{x}, p, m) \sim_{\mathsf{N}} \bar{\mathsf{c}}\}$. The set $\overline{\mathcal{W}}^{p,m}$ is $\mathcal{F}^{|Q|} \setminus \mathcal{W}^{p,m}$. The distance of a configuration c (denoted by $\mathrm{dist}(\mathsf{c})$) is

$$\mathrm{dist}(\mathsf{c}) = \min\{|w| \mid \mathsf{c} \xrightarrow{w} (\mathbf{x}, p, m)\ \exists p \in C, m < \mathsf{N}, \text{ and } \mathbf{x} \in \overline{\mathcal{W}}^{p,m}\} .$$

The following lemma shows when $\mathcal{A}$ is not regular. Given any configuration c, we use $\mathbf{x}_{\mathsf{c}}$ to denote $\text{WEIGHT-VECTOR}(\mathsf{c})$, p_{c} to denote $\text{COUNTER-STATE}(\mathsf{c})$ and n_{c} to denote $\text{COUNTER-VALUE}(\mathsf{c})$.

▶ **Lemma 24.** *Let* c *be the initial configuration of an ODCA* $\mathcal{A}$. *The following are equivalent.*
1. $\mathcal{A}$ *is not regular.*
2. *for all* $t \in \mathbb{N}$, *there exists configurations* d, e *s.t.* $n_{\mathsf{e}} < \mathsf{N}, \mathsf{c} \xrightarrow{*} \mathsf{d} \xrightarrow{*} \mathsf{e}$, $\mathbf{x}_{\mathsf{e}} \in \overline{\mathcal{W}}^{p_{\mathsf{e}}, n_{\mathsf{e}}}$ *and* $t < \mathrm{dist}(\mathsf{d}) < \infty$.
3. *there exists configurations* d, e *and a run* $\mathsf{c} \xrightarrow{*} \mathsf{d} \xrightarrow{*} \mathsf{e}$ *s.t.* $\mathsf{N}^2 + \mathsf{N} \leq n_{\mathsf{d}} \leq 2\mathsf{N}^2 + \mathsf{N}$, $\mathbf{x}_{\mathsf{e}} \in \overline{\mathcal{W}}^{p_{\mathsf{e}}, n_{\mathsf{e}}}$ *with* $n_{\mathsf{e}} < \mathsf{N}$.

Proof. $3 \to 2$: Consider an arbitrary $q \in C$, $m < \mathsf{N}$ and vector space $\mathcal{V} = \mathcal{W}^{q,m}$. Let us assume for contradiction the complement of Point 2. That is, there exists a $t \in \mathbb{N}$ such that for all configurations d' where $\mathsf{c} \xrightarrow{*} \mathsf{d}' \xrightarrow{*} \overline{\mathcal{V}} \times \{q\} \times \{m\}$, $\mathrm{dist}(\mathsf{d}') \leq t$. Note that for all d' where $n_{\mathsf{d}'} > \mathsf{N}$, $\mathrm{dist}(\mathsf{d}') \geq n_{\mathsf{d}'} - \mathsf{N}$. Hence there exists an $M \in \mathbb{N}$ such that for all d' where $\mathsf{c} \xrightarrow{*} \mathsf{d}' \xrightarrow{*} \overline{\mathcal{V}} \times \{q\} \times \{m\}$, $n_{\mathsf{d}'} \leq M$.

Consider a configuration $\mathtt{d}$ where $n_{\mathtt{d}} > \mathsf{N}^2 + \mathsf{N}$ and a run $\mathtt{c} \xrightarrow{*} \mathtt{d} \xrightarrow{*} \overline{\mathcal{V}} \times \{q\} \times \{m\}$. Point 3 shows the existence of such a run. For contradiction, it suffices to show there exists a $\mathtt{d}'$ such that $\mathtt{c} \xrightarrow{*} \mathtt{d}' \xrightarrow{*} \overline{\mathcal{V}} \times \{q\} \times \{m\}$ and $n_{\mathtt{d}'} > n_{\mathtt{d}}$. We get this from Lemma 10 Point 2, since $n_{\mathtt{c}} = 0$ and $n_{\mathtt{d}} > \mathsf{N}^2 + \mathsf{N}$.

$2 \rightarrow 1$: Assume for contradiction that for all $t \in \mathbb{N}$, there exists configurations $\mathtt{d}, \mathtt{e}$ such that $\mathtt{c} \xrightarrow{*} \mathtt{d} \xrightarrow{*} \mathtt{e}$, $\mathbf{x}_{\mathtt{e}} \in \overline{\mathcal{W}}^{p_{\mathtt{e}}, n_{\mathtt{e}}}$, $n_{\mathtt{e}} < \mathsf{N}$ and $t < \mathrm{dist}(\mathtt{d}) < \infty$ but $\mathcal{A}$ is regular. Let $\mathcal{B}$ be the weighted automaton equivalent to $\mathcal{A}$. We use $|\mathcal{B}|$ to represent the number of states of $\mathcal{B}$.

Let $t_1, t_2, \ldots t_{|\mathcal{B}|+1} \in \mathbb{N}$ such that for $i \in [1, |\mathcal{B}|]$, $t_i < t_{i+1}$, and $\mathtt{d}_{t_i}$ be such that $\mathtt{c} \xrightarrow{*} \mathtt{d}_{t_i} \xrightarrow{*} (\mathbf{x}_i, p_{\mathtt{e}}, n_{\mathtt{e}})$, $\mathbf{x}_i \in \overline{\mathcal{W}}^{p_{\mathtt{e}}, n_{\mathtt{e}}}$ and $t_i < \mathrm{dist}(\mathtt{d}_{t_i}) < t_{i+1} < \infty$. Clearly $\mathtt{d}_{t_i} \not\equiv \mathtt{d}_{t_j}$ for $i \neq j$ and corresponds to two different states of $\mathcal{B}$. Since we can find more than $|\mathcal{B}|$ pairwise non-equivalent configurations, it contradicts the assumption that $\mathcal{B}$ is equivalent to $\mathcal{A}$.

$1 \rightarrow 3$: We prove the contrapositive of the statement. Let us assume that there is no configurations $\mathtt{d}, \mathtt{e}$ and a run $\mathtt{c} \xrightarrow{*} \mathtt{d} \xrightarrow{*} \mathtt{e}$ such that $\mathsf{N}^2 + \mathsf{N} \leq n_{\mathtt{d}} \leq 2\mathsf{N}^2 + \mathsf{N}$, $\mathbf{x}_{\mathtt{e}} \in \overline{\mathcal{W}}^{p_{\mathtt{e}}, n_{\mathtt{e}}}$ with $n_{\mathtt{e}} < \mathsf{N}$. This implies that there does not exists a configuration $\mathtt{d}'$ such that $n_{\mathtt{d}'} > 2\mathsf{N}^2$, $\mathtt{c} \xrightarrow{*} \mathtt{d}' \xrightarrow{*} (\mathbf{y}, p_{\mathtt{e}}, n_{\mathtt{e}})$ for some $\mathbf{y} \in \overline{\mathcal{W}}^{p_{\mathtt{e}}, n_{\mathtt{e}}}$. Assume for contradiction that there is such a run, then there exists a portion in this run that can be "pumped down" to get a run $\mathtt{c} \xrightarrow{*} \mathtt{d}'' \xrightarrow{*} (\mathbf{y}', p_{\mathtt{e}}, n_{\mathtt{e}})$ for some configuration $\mathtt{d}''$ such that $\mathsf{N}^2 + \mathsf{N} \leq n_{\mathtt{d}''} \leq 2\mathsf{N}^2 + \mathsf{N}$ and $\mathbf{y}' \in \overline{\mathcal{W}}^{p_{\mathtt{e}}, n_{\mathtt{e}}}$. This is a contradiction. Therefore all runs starting from configuration with counter value greater than or equal to $\mathsf{N}^2 + \mathsf{N}$ "looks" similar to a run on a weighted automaton. This allows us to simulate the runs of $\mathcal{A}$ using a weighted automaton. ◀