On Büchi One-Counter Automata

Stanislav Böhm*[1], Stefan Göller†[2], Simon Halfon[3], and Piotr Hofman‡[4]

1 **Technical University of Ostrava, Ostrava, Czech Republic**
`stanislav.bohm@vsb.cz`
2 **LSV, CNRS & ENS Cachan, Université Paris-Saclay, Cachan, France**
`goeller@lsv.fr`
3 **LSV, CNRS & ENS Cachan, Université Paris-Saclay, Cachan, France**
`halfon@lsv.fr`
4 **LSV, CNRS & ENS Cachan, Université Paris-Saclay, Cachan, France**
`hofman@lsv.fr`

Abstract

Equivalence of deterministic pushdown automata is a famous problem in theoretical computer science whose decidability has been shown by Sénizergues. Our first result shows that decidability no longer holds when moving from finite words to infinite words. This solves an open problem that has recently been raised by Löding. In fact, we show that already the equivalence problem for deterministic Büchi one-counter automata is undecidable. Hence, the decidability border is rather tight when taking into account a recent result by Löding and Repke that equivalence of deterministic weak parity pushdown automata (a subclass of deterministic Büchi pushdown automata) is decidable.

Another known result on finite words is that the universality problem for vector addition systems is decidable. We show undecidability when moving to infinite words. In fact, we prove that already the universality problem for nondeterministic Büchi one-counter nets (or equivalently vector addition systems with one unbounded dimension) is undecidable.

1998 ACM Subject Classification F.1.1 Models of Computation, F.4.3 Formal Languages

Keywords and phrases infinite words, deterministic pushdown automata

Digital Object Identifier 10.4230/LIPIcs.STACS.2017.14

1 Introduction

One of the most prominent results in theoretical computer science is the decidability of the equivalence problem for deterministic pushdown automata, shown by Sénizergues [17]. Stirling proved the first complexity bound for this problem [18], namely a tower of exponentials of elementary height, see also [8] for a more recent proof by Jančar. Still there remains a remarkable complexity gap for this problem, to the best of the authors' knowledge the best-known lower bound is P-hardness, which trivially follows from the emptiness problem.

Although a doubly-exponential upper bound has been proved during the nineteen seventies by Valiant [19], too, the regularity problem for deterministic pushdown automata seems to be far from being understood: to the best of the authors' knowledge the best-known lower bound is P-hardness, again trivially following from the emptiness problem.

* Supported by grant GAČR 15-13784S.
† Supported by Labex Digicosme, Univ. Paris-Saclay, project VERICONISS.
‡ Supported by Labex Digicosme, Univ. Paris-Saclay, project VERICONISS.

Although it is unclear whether to lower the upper bound or to raise the lower bound for these central problems, it seems fair to say that we lack techniques to show lower bounds for them. Indeed, the presence of *determinism* seems to be too restrictive to encode computations of Turing machines into instances of the respective problems. To date, we do not even know if equivalence of deterministic pushdown automata is hard for NP – the upper bound of a tower of exponentials of elementary height leaves a huge complexity gap for this problem.

Therefore, subclasses have been studied in the literature for which tight complexity bounds have been obtained, such as deterministic 1-counter [2], 1-counter nets [7], visibly pushdown automata [1], normed context-free processes [3, 4, 6, 5], and height-deterministic pushdown automata [12], to mention a few.

One should bear in mind that for deterministic models universality is typically equally hard as emptiness. For nondeterministic models however universality becomes undecidabile very quickly, for instance already for 1-counter automata. Decidability of universality can be regained for such a nondeterministic model by disallowing the automata to test for zero: even universality of vector addition systems can be shown decidable by applying standard well-quasi order arguments. For its subclass 1-counter nets (i.e. 1-counter automata that cannot test for zero) universality has recently been proven to be Ackermann-complete by Hofman and Totzke [7].

The situation on infinite words. Unfortunately, equivalence and universality have not gained much attention on infinite words beyond ω-regular languages so far, in particular for pushdown automata. Recently Löding and Repke have studied the equivalence problem of deterministic pushdown automata on infinite words and proved that this problem is decidable for weak parity conditions [10, 9, 15] (this is a strict subclass of deterministic Büchi pushdown automata). In fact they showed that given two deterministic weak parity pushdown automata $\mathcal{A}$ and $\mathcal{B}$ one can effectively construct two deterministic pushdown automata $\mathcal{A}'$ and $\mathcal{B}'$ running on finite words such that $\mathcal{A}$ and $\mathcal{B}$ accept the same infinite words if, and only if, $\mathcal{A}'$ and $\mathcal{B}'$ accept the same finite words. Extending this decidability result to deterministic parity pushdown automata, or first to the class of deterministic Büchi pushdown automata did not seem to be achievable that easily. In a recent article [9] Löding explicitly raised the question if equivalence of deterministic parity pushdown automata is decidable.

Our contributions. In this paper we show that two central decision problems that are decidable on finite words, become undecidable on infinite words. First, we prove that the equivalence problem is already undecidable for deterministic 1-counter automata with a Büchi acceptance condition (even without ε-transitions). This solves the above-mentioned question raised by Löding in [9]. This undecidability result can be considered as rather tight since, as mentioned above, equivalence of deterministic weak parity pushdown automata is decidable [10].

Our proof heavily exploits the infinity of the input words and we are confident that our proof technique may be applied to related problems on infinite words. As mentioned before, significant lower bounds involving deterministic automata are typically difficult to prove. In fact, to the best of our knowledge, there are no handy lower bound techniques known for equivalence problem of *deterministic* automata.

Moreover, we consider this undecidability result as somewhat surprising since from a given deterministic 2-counter automaton we construct a deterministic 1-counter automaton such that for all two of its configurations all ω-words that distinguish the two configurations must end in the quite restrictive form $(w\#)^\omega$, where w is the unique halting computation of the initial deterministic 2-counter automaton, if it exists. We contrast our undecidability result on infinite words with a recent NL-completeness result for deterministic 1-counter automata on finite words [2].

As a second result we show that on infinite words universality becomes undecidable on Büchi 1-counter nets. This is shown by a sequence of reductions starting from the boundedness problem for decremental-error vector addition systems, which is undecidable due to Schnoebelen [16]. In the final step of these reductions we again heavily rely on the fact that we employ a Büchi condition.

To the best of the authors' knowledge comparable jumps from decidability on finite words to undecidability on infinite words appear for satisfiability of metric temporal logic [13, 14].

Organization of this paper. In Section 2 we introduce the necessary definitions and state our main results. The equivalence problem for deterministic Büchi 1-counter automata is shown to be undecidable in Section 3. Universality of Büchi 1-counter nets is shown to be undecidable in Section 4. We conclude in Section 5. Some of the proofs appear in the appendix due to space restrictions.

2 Preliminaries

By $\mathbb{N} = \{0, 1, \ldots, \}$ we denote the non-negative integers. For each non-negative integer k we denote by $[1, k]$ the set $\{1, \ldots, k\}$. If X is a non-empty set and $\boldsymbol{x} = (x_1, \ldots, x_k) \in X^k$ we write $\boldsymbol{x}(i)$ to denote x_i, i.e. the i-th component of $\boldsymbol{x}$. We use bold-face font only to denote such vectors. For every integer $z \in \mathbb{Z}$ we denote its *signum* by $\mathrm{sgn}(z) = -1$ if $z < 0$, $\mathrm{sgn}(z) = 0$ if $z = 0$ and $\mathrm{sgn}(z) = 1$ if $z > 0$. Given a set Σ and some finite word $w = a_1 \cdots a_n \in \Sigma^n$ with $a_i \in \Sigma$ we define the infix $w[i, j] = a_i a_{i+1} \cdots a_j$ and $w[i] = w[i, i] = a_i$. Similar remarks apply to infinite words $w = a_1 a_2 \cdots$; moreover we write $w[i, \infty]$ to denote $a_i a_{i+1} a_{i+2} \cdots$.

For each $k \geq 1$ a *k-counter automaton* is a tuple $\mathcal{A} = (Q_\mathcal{A}, \Sigma_\mathcal{A}, \delta_\mathcal{A}, q_0, F_\mathcal{A})$, where $Q_\mathcal{A}$ is a finite set of *states*, $\Sigma_\mathcal{A}$ is a non-empty finite *alphabet*, $\delta_\mathcal{A} \subseteq Q_\mathcal{A} \times \Sigma_\mathcal{A} \times \{0,1\}^k \times \{-1, 0, 1\}^k \times Q_\mathcal{A}$ is a set of *transitions*, $q_0 \in Q_\mathcal{A}$ is an *initial state*, and $F_\mathcal{A} \subseteq \delta_\mathcal{A}$ is a set of *final transitions*. It is worth mentioning that typically the accepting condition is given by a set of states rather than by a set of transitions. But it is not hard to see that the two formalisms are effectively equivalent: indeed one can translate one formalism to the other in polynomial time.

When it is not of importance we sometimes also drop the last component of $\mathcal{A}$. We say $\mathcal{A}$ is *deterministic* if for every $(q, a, \boldsymbol{\sigma}) \in Q_\mathcal{A} \times \Sigma_\mathcal{A} \times \{0, 1\}^k$ there is at most one transition $\tau = (p, b, \boldsymbol{\mu}, \boldsymbol{u}, p') \in \delta_\mathcal{A}$ with $p = q$, $b = a$ and $\boldsymbol{\mu} = \boldsymbol{\sigma}$. A *configuration* of $\mathcal{A}$ is an element $(q, \boldsymbol{n}) \in Q_\mathcal{A} \times \mathbb{N}^k$ that we also write as $q(\boldsymbol{n})$ in the following. The bit vector $\boldsymbol{\sigma} \in \{0, 1\}^k$ that appears in a transition determines which sign each counter is expected to have for the transition to be applicable. More formally, for two configurations $p(\boldsymbol{m})$ and $q(\boldsymbol{n})$ and a transition of the form $\tau = (p, a, \boldsymbol{\sigma}, \boldsymbol{u}, q)$ we write $p(\boldsymbol{m}) \xrightarrow{\tau} q(\boldsymbol{n})$ if $\boldsymbol{\sigma}(i) = \mathrm{sgn}(\boldsymbol{m}(i))$ and $\boldsymbol{n}(i) = \boldsymbol{m}(i) + \boldsymbol{u}(i)$ for all $i \in [1, k]$. The relation $\xrightarrow{\tau}$ is extended to finite words over $\delta_\mathcal{A}$ inductively as follows, where $\rho \in \delta_\mathcal{A}^*$ and $\tau \in \delta_\mathcal{A}$: $p(\boldsymbol{m}) \xrightarrow{\varepsilon} p(\boldsymbol{m})$ for all configurations $p(\boldsymbol{m})$ and $p(\boldsymbol{m}) \xrightarrow{\rho\tau} q(\boldsymbol{n})$ if $p(\boldsymbol{m}) \xrightarrow{\rho} r(\boldsymbol{\ell})$ and $r(\boldsymbol{\ell}) \xrightarrow{\tau} q(\boldsymbol{n})$ for some configuration $r(\boldsymbol{\ell})$. We write $p(\boldsymbol{m}) \xrightarrow{*} q(\boldsymbol{n})$ if $p(\boldsymbol{m}) \xrightarrow{\rho} q(\boldsymbol{n})$ for some $\rho \in \delta_\mathcal{A}^*$. The following decision problem is well-known to be undecidable in its full generality by [11].

REACHABILITY

Input: A k-counter automaton $\mathcal{A} = (Q_\mathcal{A}, \Sigma_\mathcal{A}, \delta_\mathcal{A}, q_0)$ and a control state $q_f \in Q_\mathcal{A}$.
Question: $q_0(\boldsymbol{0}) \xrightarrow{*} q_f(\boldsymbol{0})$?

For every transition $\tau = (p, a, \boldsymbol{\sigma}, \boldsymbol{u}, q)$ let $\textsc{Read}(\tau) = a$ denote the letter of the transition. We extend $\textsc{Read}$ to a (letter-to-letter) morphism from $\delta_{\mathcal{A}}^*$ to $\Sigma_{\mathcal{A}}^*$ in the usual way.

For words $w \in \Sigma_{\mathcal{A}}^*$ we write $p(\boldsymbol{m}) \xrightarrow{w} q(\boldsymbol{u})$ if $p(\boldsymbol{m}) \xrightarrow{\rho} q(\boldsymbol{n})$ for some $\rho \in \delta_{\mathcal{A}}^*$ with $\textsc{Read}(\rho) = w$. In this case we also say that ρ is a *run* of $\mathcal{A}$ for the word w from $p(\boldsymbol{m})$ to $q(\boldsymbol{n})$. We say that a finite run ρ is *accepting* if $\rho \in \delta_{\mathcal{A}}^* F_{\mathcal{A}}$, i.e. the last transition of ρ is accepting, otherwise we say ρ is *rejecting*.

The only acceptance condition on infinite words that we study in this paper is the Büchi acceptance condition. Given an infinite word $w \in \Sigma_{\mathcal{A}}^\omega$ an *ω-run* of $\mathcal{A}$ for w from configuration $p(\boldsymbol{m})$ is an infinite sequence $\rho \in \delta_{\mathcal{A}}^\omega$ such that every finite prefix π of ρ is a run from $p(\boldsymbol{m})$ to some configuration, and $\textsc{Read}(\pi)$ is a prefix of w. We say that the ω-run ρ is *accepting* if $\rho[i] \in F_{\mathcal{A}}$ for infinitely many $i \geq 1$, otherwise we say ρ is *rejecting*. The *language of a configuration $p(\boldsymbol{m})$ of $\mathcal{A}$* is defined as

$$L(\mathcal{A}, p(\boldsymbol{m})) = \{w \in \Sigma_{\mathcal{A}}^* \mid \text{there is an accepting run from } p(\boldsymbol{m}) \text{ for } w\}.$$

Similarly the *ω-language of a configuration $p(\boldsymbol{m})$ of $\mathcal{A}$* is defined as

$$L_\omega(\mathcal{A}, p(\boldsymbol{m})) = \{w \in \Sigma_{\mathcal{A}}^\omega \mid \text{there is an accepting } \omega\text{-run from } p(\boldsymbol{m}) \text{ for } w\}.$$

We sometimes just write $L(p(\boldsymbol{m}))$ or $L_\omega(p(\boldsymbol{m}))$ if $\mathcal{A}$ is clear from the context.

Given a finite alphabet $\Sigma_{\mathcal{A}}$ and a language $L \subseteq \Sigma_{\mathcal{A}}^*$ (resp. ω-language $R \subseteq \Sigma_{\mathcal{A}}^\omega$) we say L (resp. R) is *universal* if $L = \Sigma_{\mathcal{A}}^*$ (resp. $R = \Sigma_{\mathcal{A}}^\omega$).

We say k-counter automaton $\mathcal{A} = (Q_{\mathcal{A}}, \Sigma_{\mathcal{A}}, \delta_{\mathcal{A}}, q_0, F_{\mathcal{A}})$ is a *k-counter net* if the test for zero is not allowed, formally for any transition $(p, a, \boldsymbol{\sigma}, \boldsymbol{u}, q) \in \delta_{\mathcal{A}}$ and for any $\boldsymbol{\mu} \in \{0,1\}^k$ the transition $(p, a, \boldsymbol{\mu}, \boldsymbol{u}, q)$ is in $\delta_{\mathcal{A}}$ as well. As the third component of any transition of a k-counter net no longer plays a role we omit it, i.e. we stipulate $\delta_{\mathcal{A}} \subseteq Q_{\mathcal{A}} \times \Sigma_{\mathcal{A}} \times \{-1, 0, 1\}^k \times Q_{\mathcal{A}}$. The following monotonicity property holds immediately by definition of k-counter nets (see for instance [7]).

▶ **Lemma 1.** *$L(\mathcal{B}, p(\boldsymbol{m})) \subseteq L(\mathcal{B}, p(\boldsymbol{n}))$ if $\boldsymbol{m} \leq \boldsymbol{n}$ (and where $\leq$ is meant component-wise) and $\mathcal{B}$ is a one-counter net.*

The following problems will be of central interest in this paper, where the latter is a special case of the former.

ω-Language Equivalence

Input: A k-counter automaton $\mathcal{A}$ and two configurations $p(\boldsymbol{m})$ and $q(\boldsymbol{n})$.
Question: $L_\omega(p(\boldsymbol{m})) = L_\omega(q(\boldsymbol{n}))$?

Our first main result is the following.

▶ **Theorem 2.** ω-Language Equivalence *is undecidable for deterministic* 1*-counter automata.*

ω-Universality

Input: A k-counter automaton $\mathcal{A}$ and a configuration $p(\boldsymbol{m})$.
Question: $L_\omega(p(\boldsymbol{m})) = \Sigma_{\mathcal{A}}^\omega$?

Our second main result is the following.

▶ **Theorem 3.** ω-Language Universality *is undecidable for* 1*-counter nets.*

3 The equivalence problem for deterministic Büchi 1-counter automata is undecidable

We reduce from Reachability for deterministic 2-counter automata, which is undecidable [11]. Our construction does not rely on the fact that our 2-counter automaton is deterministic but it makes the constructions easier to state as distuingishing words of two configurations of the constructed deterministic Büchi 1-counter automaton must repeatedly encounter the unique sequence of transitions from the initial configuration $q_0(0,0)$ to the configuration $q_f(0,0)$.

Let us therefore fix some deterministic 2-counter automaton $\mathcal{A} = (Q_\mathcal{A}, \Sigma_\mathcal{A}, \delta_\mathcal{A}, q_0)$ and some state $q_f \in Q_\mathcal{A}$. We show how to construct a deterministic 1-counter automaton $\mathcal{B} = (Q, \Sigma, \delta, q_0, F)$, where Q will contain state $q_0^{(1)}$ and $q_0^{(2)}$ such that

$$q_0(0,0) \xrightarrow{*} q_f(0,0) \quad \text{if, and only if,} \quad L_\omega(q_0^{(1)}(0)) \neq L_\omega(q_0^{(2)}(0)) \ .$$

We can assume without loss of generality that $\Sigma_\mathcal{A} = \{a\}$ is a singleton and that there is no $(q, a, \boldsymbol{\sigma}, \boldsymbol{u}, q') \in \delta_\mathcal{A}$ with $q = q_f$. Under this assumption and the fact that $\mathcal{A}$ is deterministic we have that if $q_0(0,0) \xrightarrow{*} q_f(0,0)$, then there exists a unique run $q_0(0,0) \xrightarrow{\rho} q_f(0,0)$. Let $Q^{(1)} = \{q^{(1)} \mid q \in Q_\mathcal{A}\}$ and $Q^{(2)} = \{q^{(2)} \mid q \in Q_\mathcal{A}\}$ be fresh copies of $Q_\mathcal{A}$. We set $Q = Q^{(1)} \uplus Q^{(2)} \uplus \{p^{(1)}, p^{(2)}\}$, where $p^{(1)}$ and $p^{(2)}$ are two fresh control states. A *mode* is an element from $(Q^{(1)} \cup Q^{(2)}) \times \{0,1\}$.

The idea is that $\mathcal{B}$'s configuration in states $Q^{(1)}$ are there to mimic the first counter of $\mathcal{A}$, whereas the configurations in states $Q^{(2)}$ are there to mimic the second counter of $\mathcal{A}$. To this end, we define that a mode $(q^{(i)}, \sigma) \in (Q^{(1)} \cup Q^{(2)}) \times \{0,1\}$ is *compatible* with a transition $\tau = (r, a, \boldsymbol{\sigma}, \boldsymbol{u}, r')$ if $q = r$ and σ is the same signum as τ's signum of the counter that is to be mimicked, formally $\sigma = \boldsymbol{\sigma}(i)$. Moreover, those configurations with control state in $p^{(1)}$ or in $p^{(2)}$ are denoted to be *erroneous*. Let us define the remaining components of $\mathcal{B}$.

We define the alphabet to be $\Sigma = \delta_\mathcal{A} \uplus \{\#\}$, where $\#$ is a fresh symbol. The transitions are defined as $\delta = \delta_{sim} \cup \delta_{restart} \cup \delta_{reset} \cup \delta_{err}$, whose components we successively list and comment on below; in addition a general structure of the construction is presented in Figure 1.

The transitions in δ_{sim} correspond to faithful simulations of transitions in $\mathcal{A}$ in which, as already mentioned above, a state $q^{(i)}$ is supposed to mimic the behavior on counter i:

$$\delta_{sim} = \{(q^{(i)}, \tau, \boldsymbol{\sigma}(i), \boldsymbol{u}(i), r^{(i)}) \mid (q^{(i)}, \boldsymbol{\sigma}(i)) \text{ is comp. with } \tau = (q, a, \boldsymbol{\sigma}, \boldsymbol{u}, r), i \in \{1,2\}\}$$

The letter $\#$ is to be understood as a letter to restart the simulation of the machine $\mathcal{A}$ and is only treated as non-erroneous when executed from the configurations $\{p^{(1)}(0), p^{(2)}(0)\}$ or from $\{q_f^{(1)}(0), q_f^{(2)}(0)\}$.

$$\delta_{restart} \quad = \quad \{(p^{(i)}, \#, 0, 0, q_0^{(3-i)}) \mid i \in \{1,2\}\} \cup \{(q_f^{(i)}, \#, 0, 0, q_0^{(i)}) \mid i \in \{1,2\}\} \tag{1}$$

The next type of transitions are transitions that are executed in order to reset the simulation, that is to lead back to a simulation of $\mathcal{A}$ from configuration $q_0(0,0)$, possibly by decrementing the counter for it to eventually equal zero.

$$\begin{aligned} \delta_{reset} \quad = \quad & \{(p^{(i)}, \tau, 1, -1, p^{(i)}), (p^{(i)}, \tau, 0, 0, p^{(i)}) \mid \tau \in \delta_\mathcal{A}, i \in \{1,2\}\} & (2) \\ & \cup \{(p^{(i)}, \#, 1, -1, p^{(i)}) \mid i \in \{1,2\}\} & (3) \end{aligned}$$

The last type of transitions that remain are erroneous transitions – they all lead to erroneous configurations. The first kind of erroneous transitions can be applied at configurations

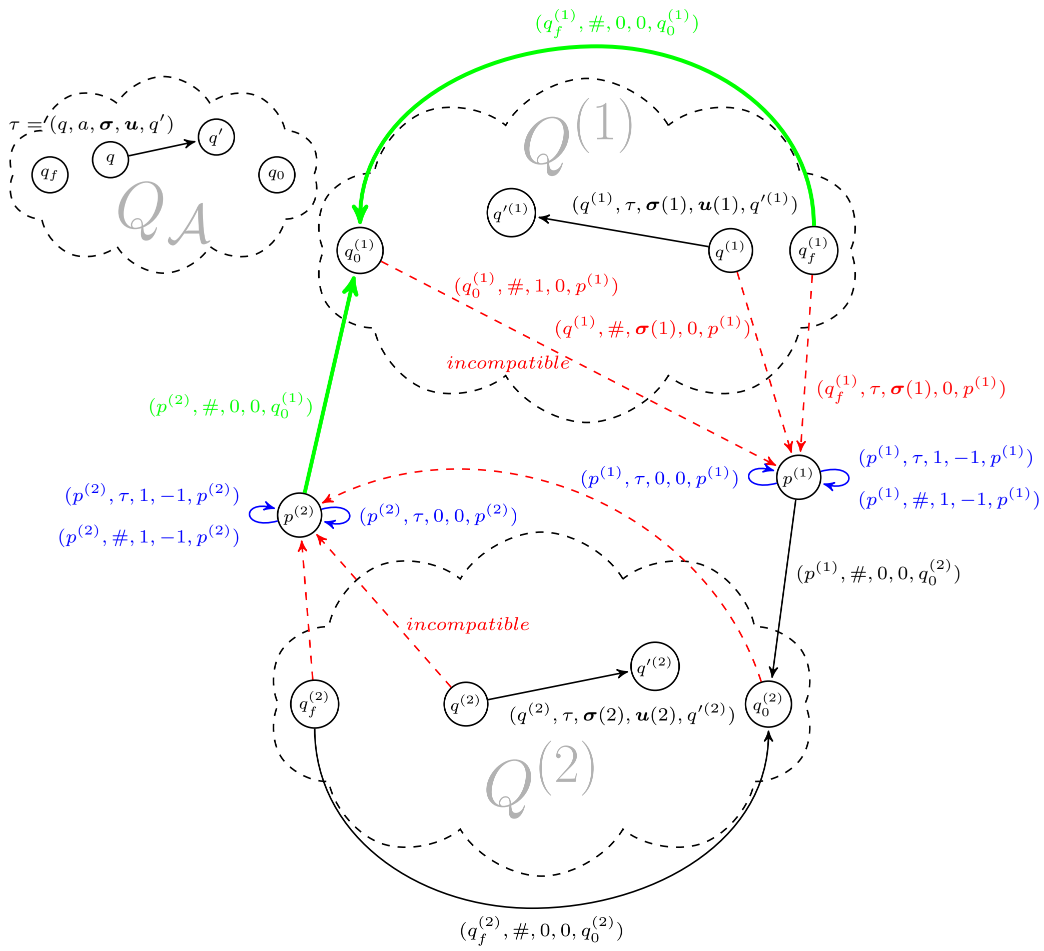

Figure 1 On the left hand side there is a cloud that is an emblematic description of the deterministic 2-counter automaton $\mathcal{A}$ with the set of states $Q_{\mathcal{A}}$. On the right hand side the structure of the automaton $\mathcal{B}$ is presented. Two clouds stand for two copies of $Q_{\mathcal{A}}$, where the bottom copy that represents $Q^{(2)}$ is graphically flipped. Red dashed arrows are elements of δ_{err}, blue arrows are elements of δ_{reset}, thick green arrows are accepting.

of the form $q^{(i)}(n) \in (Q^{(1)} \cup Q^{(2)}) \times \mathbb{N}$ whose mode $(q^{(i)}, \mathrm{sgn}(n))$ is incompatible with the letter $\tau \in \delta_{\mathcal{A}}$ to be read. The second kind of erroneous transitions are transitions that read the symbol $\#$ in a moment when a restart was not expected.

$$\begin{aligned} \delta_{err} \quad = \quad & \{(q^{(i)}, \tau, \sigma, 0, p^{(i)}) \mid (q^{(i)}, \sigma) \text{ is incompatible with } \tau \in \delta_{\mathcal{A}}, i \in \{1, 2\}\} && (4)\\ & \cup \{(q^{(i)}, \#, \sigma, 0, p^{(i)}) \mid q \in Q_{\mathcal{A}} \setminus \{q_f\}, \sigma \in \{0, 1\}, i \in \{1, 2\}\} && (5)\\ & \cup \{(q_f^{(i)}, \#, 1, 0, p^{(i)}) \mid i \in \{1, 2\}\} && (6) \end{aligned}$$

Since $\mathcal{A}$ is deterministic it is readily checked that $\mathcal{B}$ is deterministic as well. The set of final transitions is defined to be $F = \{(q_f^{(1)}, \#, 0, 0, q_0^{(1)}), (p^{(2)}, \#, 0, 0, q_0^{(1)})\}$.

For any two sets X and Y let $\Delta(X, Y) = X \setminus Y \cup Y \setminus X$ denote their symmetric difference.

▶ **Lemma 4.** *If $q_0(0, 0) \xrightarrow{*} q_f(0, 0)$, then $L_\omega(q_0^{(1)}(0)) \neq L_\omega(q_0^{(2)}(0))$.*

Proof. Assume there exists a run $q_0(0, 0) \xrightarrow{\rho} q_f(0, 0)$ in $\mathcal{A}$. Since $\mathcal{B}$ is deterministic there exists a unique run $q_0^{(1)}(0) \xrightarrow{\pi^{(1)}} q_f^{(1)}(0)$ and a unique run $q_0^{(2)}(0) \xrightarrow{\pi^{(2)}} q_f^{(2)}(0)$ with $\textsc{Read}(\pi^{(1)}) =$

$\textsc{Read}(\pi^{(2)}) = \rho$ in $\mathcal{B}$. Since $\rho \in \delta_{\mathcal{A}}^*$ it follows $\pi^{(1)}, \pi^{(2)} \in (\delta \setminus F)^*$, i.e. neither $\pi^{(1)}$ nor $\pi^{(2)}$ contains any final transition. Moreover we have $q_f^{(1)}(0) \xrightarrow{(q_f^{(1)},\#,0,0,q_0^{(1)})} q_0^{(1)}(0)$ and $q_f^{(2)}(0) \xrightarrow{(q_f^{(2)},\#,0,0,q_0^{(2)})} q_0^{(2)}(0)$, by construction of $\mathcal{B}$, where the former used transition is final and the latter used transition is not final.

Hence, on the infinite word $(\rho\#)^\omega$ the unique ω-run from $q_0^{(1)}(0)$ encounters infinitely many final transitions, whereas the unique ω-run from $q_0^{(2)}(0)$ does not encounter any final transition. Thus $(\rho\#)^\omega \in \Delta(L_\omega(q_0^{(1)}(0)), L_\omega(q_0^{(2)}(0)))$ and hence $L_\omega(q_0^{(1)}(0)) \neq L_\omega(q_0^{(2)}(0))$. ◀

Let us fix a sequence of transitions $\pi \in \delta^\omega$. For $j \in \{1,2\}$ we say π is *j-pure* if $\pi[i] \in Q^{(j)} \times \Sigma \times \{0,1\} \times \{-1,0,1\} \times Q^{(j)}$ for all $i \geq 1$.

▶ **Lemma 5.** *Let π be the ω-run from some configuration in $\mathcal{B}$ such that $\#$ appears infinitely often in $\textsc{Read}(\pi)$. Then π is rejecting if, and only if, $\pi[\ell,\infty]$ is 2-pure for some $\ell \geq 1$.*

Proof.

"If": If π is 2-eventually, then clearly is π rejecting since every accepting transition has $q_0^{(1)}$ as last component.

"Only-if": Assume π is a rejecting ω-run from the configuration $q^{(i)}(n)$ in $\mathcal{B}$. Since $\#$ appears infinitely often in $\textsc{Read}(\pi)$ a simple inspection of the rules in $\mathcal{B}$ (in particular the resetting behavior at states $p^{(1)}$ and $p^{(2)}$, see also Figure 1) shows that π must contain infinitely many occurrences of at least one of the following transitions:

- $\tau_1 = (p^{(1)}, \#, 0, 0, q_0^{(2)})$
- $\tau_2 = (p^{(2)}, \#, 0, 0, q_0^{(1)})$
- $\tau_3 = (q_f^{(1)}, \#, 0, 0, q_0^{(1)})$
- $\tau_4 = (q_f^{(2)}, \#, 0, 0, q_0^{(2)})$

Surely, both τ_2 and τ_3 can only appear finitely often in π since $F = \{\tau_2, \tau_3\}$ and π is assumed to be rejecting. Bearing in mind that $\textsc{Read}(\pi)$ contains infinitely many $\#$'s one further observes that by construction of $\mathcal{B}$ the transition τ_1 can only appear infinitely often if the transition τ_2 appears infinitely often. Therefore all the transitions τ_1, τ_2 and τ_3 can only appear finitely often in π. Hence there exists some $\ell \in \mathbb{N}$ such that $\pi[\ell,\infty]$ contains neither of the transitions τ_1, τ_2 nor τ_3. Since $\textsc{Read}(\pi[\ell,\infty])$ still contains infinitely many $\#$'s it follows that $\pi[\ell,\infty]$ is 2-pure by construction of $\mathcal{B}$. ◀

Let π be a run or an ω-run of $\mathcal{B}$. For any state $r \in Q$, we say that π is *r-free* if none of the transitions that appears in π contains r as first or as last component. The following lemma shows that in case an infinite word lies in the symmetric difference of the language of *any* two configurations of $\mathcal{B}$, then it eventually repeatedly simulates the unique run from $q_0(0,0)$ to $q_f(0,0)$ with a separating symbol $\#$ in between, if such a run exists.

▶ **Lemma 6.** *Let $s(m)$ and $t(n)$ be two arbitrary configurations of $\mathcal{B}$. Then the following holds:*

1. *Every $w \in \Delta(L_\omega(s(m)), L_\omega(t(n)))$ contains infinitely many $\#$'s.*
2. *Let $w \in \Delta(L_\omega(s(m)), L_\omega(t(n)))$, let α be the unique run from $s(m)$ and let β be the unique run from $t(n)$ such that $\textsc{Read}(\alpha) = \textsc{Read}(\beta) = w$. Then there is some $j \in \{1,2\}$ such that*
   (a) *$\alpha[h_1,\infty]$ is j-pure and $\beta[h_1,\infty]$ is $(3-j)$-pure for some $h_1 \geq 1$ and moreover*
   (b) *$q_0(0,0) \xrightarrow{*} q_f(0,0)$ and $w \in \Sigma^*(\rho\#)^\omega$, where $q_0(0,0) \xrightarrow{\rho} q_f(0,0)$ is the unique run from $q_0(0,0)$ to $q_f(0,0)$.*

Proof. Point (1) immediately follows from the fact that $\textsc{Read}(\tau) = \#$ for all $\tau \in F$.

Let us show (2). Let us fix an arbitrary $w \in \Delta(L_\omega(s(m)), L_\omega(t(n)))$. Let $s(m) \xrightarrow{\alpha}$ be the unique run from $s(m)$ and let $t(m) \xrightarrow{\beta}$ be the unique run from $t(n)$ such that $\textsc{Read}(\alpha) = \textsc{Read}(\beta) = w$. Without loss of generality let us assume that α is accepting and that β is rejecting. Then there exists a position $h_0 \in \mathbb{N}$ such that

$$\beta[h_0, \infty] \text{ is 2-pure, due to Lemma 5.} \tag{7}$$

This in turn, by construction of $\mathcal{B}$ and recalling that any transition having $p^{(2)}$ as source or target state is *not* 2-pure, implies that every transition in β that reads a # must lead to the configuration $q_0^{(2)}(0)$, i.e.:

$$\forall i \geq h_0 : \quad (\textsc{Read}(\beta[i])) = \# \quad \Longrightarrow \quad \beta[i] = (q_f^{(2)}, \#, 0, 0, q_0^{(2)}) \tag{8}$$

For establishing (2.a), it suffices to show that $\alpha[h_1, \infty]$ is 1-pure for some $h_1 \geq h_0$. For this, we prove the following claim, whose proof we postpone to the end.

▶ **Claim 7.** *$\alpha[h_0, \infty]$ is $p^{(1)}$-free.*

We will show that $\alpha[h_1, \infty]$ is 1-pure for some $h_1 \geq h_0$. As α is accepting and $\alpha[h_0, \infty]$ is $p^{(1)}$-free the only possible accepting transition that appears infinitely often in α must be the transition $(q_f^{(1)}, \#, 0, 0, q_0^{(1)})$. Hence there exists some $h_1 \geq h_0$ such that $\alpha[h_1, \infty]$ does not contain the other accepting transition, namely $(p^{(2)}, \#, 0, 0, q_0^{(1)})$. Altogether we have that the only transition labeled with # in $\alpha[h_1, \infty]$ is $(q_f^{(1)}, \#, 0, 0, q_0^{(1)})$. So $\alpha[h_1, \infty]$ is 1-pure which immediately follows from construction of $\mathcal{B}$ and from the fact that $\textsc{Read}(\alpha[h_1, \infty])$ contains infinitely many #'s. This shows (2a.).

Let us finally prove (2b.). Still, the word $w[h_1, \infty]$ contains infinitely many #'s. That is, we can uniquely factorize $w[h_1, \infty]$ as

$$w[h_1, \infty] = w_0 \# w_1 \# \cdots$$

where $w_\ell \in \delta_{\mathcal{A}}^*$ for each $\ell \geq 0$. Since $\alpha[h_1, \infty]$ is 1-pure and therefore $p^{(1)}$-free and $\beta[h_1, \infty]$ is 2-pure and therefore $p^{(2)}$-free it follows from construction of $\mathcal{B}$ that for all $\ell \geq 0$:

- $s(m) \xrightarrow{w[1,h_1-1]w_1\#w_2\cdots\#w_\ell} q_f^{(1)}(0) \xrightarrow{\#} q_0^{(1)}(0)$ and
- $t(n) \xrightarrow{w[1,h_1-1]w_1\#w_2\cdots\#w_\ell} q_f^{(2)}(0) \xrightarrow{\#} q_0^{(2)}(0)$.

Therefore

$$\forall \ell \geq 1 : \quad q_0^{(1)}(0) \xrightarrow{w_\ell} q_f^{(1)}(0) \quad \text{and} \quad q_0^{(2)}(0) \xrightarrow{w_\ell} q_f^{(2)}(0) \quad . \tag{9}$$

Recall that $\textsc{Read}(\alpha[h_1, \infty]) = \textsc{Read}(\beta[h_1, \infty]) = w_1 \# w_2 \# \cdots$ Again since $\alpha[h_1, \infty]$ is 1-pure and $\beta[h_1, \infty]$ is 2-pure we can conclude from (9) and the construction of $\mathcal{B}$ that

$$\forall \ell \geq 1 : \quad q_0(0,0) \xrightarrow{w_\ell} q_f(0,0) \quad .$$

But since $\mathcal{A}$ is assumed to be deterministic and the latter is clearly only possible when $q_0(0,0) \xrightarrow{*} q_f(0,0)$ we must therefore have $w_\ell = \rho$ for all $\ell \geq 1$, where ρ denotes the unique run from $q_0(0,0)$ to $q_f(0,0)$. Hence Point (2b.) follows.

Proof of the Claim. Assume by contradiction that $\alpha[h_0, \infty]$ is not $p^{(1)}$-free. Then there exists some $k \geq h_0$ such that $\alpha[k] = (q, x, \sigma, u, p^{(1)}) \in \delta$ $\alpha[k] = (p^{(1)}, x, \sigma, u, q) \in \delta$ for some $q \in Q$, some $x \in \Sigma$, some $\sigma \in \{0,1\}$, and some $u \in \{-1, 0, 1\}$. We only treat the case $\alpha[k] = (q, x, \sigma, u, p^{(1)}) \in \delta$ here, the other case is analogous. Hence, $s(m) \xrightarrow{\alpha[1,k]} p^{(1)}(m_1)$ for some $m_1 \in \mathbb{N}$. In the same moment (recalling that $\beta[h_0, \infty]$ is 2-pure) we have $t(n) \xrightarrow{\beta[1,k]} q^{(2)}(n_1)$

for some $n_1 \in \mathbb{N}$ and some $q^{(2)} \in Q^{(2)}$. Of course $w[k+1,\infty]$ still contains infinitely many #'s. Let z be the shortest finite prefix of $w[k+1,\infty]$ of length at least $m_1 + 1$ that ends with a #. Note that $w[1,k]z$ is a finite prefix of our infinite word w.

We have $p^{(1)}(m_1) \xrightarrow{z} q_0^{(2)}(0)$ by definition of z and the construction of $\mathcal{B}$ and therefore $s(m) \xrightarrow{w[1,k]z} q_0^{(2)}(0)$. Moreover, since z ends with a # we have $t(n) \xrightarrow{w[1,k]z} q^{(2)}(0)$ by (8). Thus, both $s(m) \xrightarrow{w[1,k]z} q_0^{(2)}(0)$ and $t(n) \xrightarrow{w[1,k]z} q^{(2)}(0)$. Since $\mathcal{B}$ is deterministic this clearly contradicts $w \in \Delta(L_\omega(s(m)), L_\omega(t(n)))$ and ends the proof of the claim. ◀

◀

Lemma 4 and Lemma 6 together imply that $q_0(0,0) \xrightarrow{*} q_f(0,0)$ if, and only if, $L_\omega(q_0^{(1)}(0)) \neq L_\omega(q_0^{(2)}(0))$. Hence Theorem 2 follows.

4 The universality problem for 1-counter nets

In this section we will work with k-counter automata that have incremental errors. For this it is convenient to denote for each $i \in [1,k]$ the unit vector $\boldsymbol{e}_i \in \mathbb{N}^k$, i.e. $\boldsymbol{e}_i(i) = 1$ and $\boldsymbol{e}_i(j) = 0$ for all $j \in [1,k] \setminus \{i\}$. We define $E = \{\boldsymbol{e}_i \mid i \in [1,k]\}$. Given a k-counter automaton $\mathcal{A} = (Q_\mathcal{A}, \Sigma_\mathcal{A}, \delta_\mathcal{A}, q_0, F_\mathcal{A})$ we define an *incrementing-error transition relation* $\xrightarrow{\tau}_+$ as follows, where τ ranges over $\delta_\mathcal{A} \cup E$ and $p(\boldsymbol{m})$ and $q(\boldsymbol{n})$ are configurations:

$$p(\boldsymbol{m}) \xrightarrow{\tau}_+ q(\boldsymbol{n}) \text{ if } \begin{cases} \tau \in \delta_\mathcal{A} \text{ and } p(\boldsymbol{m}) \xrightarrow{a} q(\boldsymbol{n}), \text{or} \\ \tau = \boldsymbol{e}_i \text{ and } p = q \text{ and } \boldsymbol{n} = \boldsymbol{m} + \boldsymbol{e}_i \end{cases}$$

The relation $\xrightarrow{\tau}_+$ is naturally extended to words over $\delta_\mathcal{A} \cup E$. An *inc-run* is a run of the form $p(\boldsymbol{m}) \xrightarrow{\rho}_+ q(\boldsymbol{n})$ for some $\rho \in (\delta_\mathcal{A} \cup E)^*$. We say that $q(\boldsymbol{n})$ is *inc-reachable* from $p(\boldsymbol{m})$ (and also write $p(\boldsymbol{m}) \xrightarrow{*}_+ q(\boldsymbol{n})$ for this) if $p(\boldsymbol{m}) \xrightarrow{\rho}_+ q(\boldsymbol{n})$ for some $\rho \in (\delta_\mathcal{A} \cup E)^*$. Finally, for a word $w \in (\Sigma_\mathcal{A} \cup E)^*$ we write $p(\boldsymbol{m}) \xrightarrow{w}_+ q(\boldsymbol{n})$ if there is ρ an inc-run $p(\boldsymbol{m}) \xrightarrow{\rho}_+ q(\boldsymbol{n})$ such that $\textsc{Read}(\rho) = w$ where $\textsc{Read}$ is defined like in Section 2 and stipulating $\textsc{Read}(\boldsymbol{e_i}) = \boldsymbol{e_i}$ for every $1 \leq i \leq k$.

Let $q_f \in Q$. We write $q_0(\mathbb{N}, 0, \ldots, 0) \xrightarrow{*}_+ q_f(\boldsymbol{0})$ if for all $n \in \mathbb{N}$ we have $q_0(n, 0, \ldots, 0) \xrightarrow{*}_+ q_f(\boldsymbol{0})$. We also say that q_f is *forall inc-reachable from* q_0.

The following lemma can be proven by reduction from the boundedness problem for decremental-error k-counter automata (a.k.a. lossy counter machines) [16].

▶ **Lemma 8.** *Given a k-counter automaton and two of its states q_0 and q_f, the question if q_f is forall inc-reachable from q_0 is undecidable.*

The following lemma states that one can simulate inc-runs of k-counter automata in terms of a universality problem for 1-counter nets. The construction was essentially already present in Theorem 3 in [7], however we need a more general simulation lemma since we are interested in simulating runs that start in particular configurations (and not in configurations that have all its counters zero).

▶ **Lemma 9.** *For a given k-counter automaton $\mathcal{A} = (Q_\mathcal{A}, \Sigma_\mathcal{A}, \delta_\mathcal{A}, q_0)$ and a state $q_f \in Q_\mathcal{A}$ one can construct a 1-counter net $\mathcal{B} = (Q_\mathcal{B}, \Sigma_\mathcal{B}, \delta_\mathcal{B}, q_\mathcal{B}, \delta_\mathcal{B})$ with a state $q_z \in Q_\mathcal{B}$ such that for every $n \in \mathbb{N}$ the two following statements are equivalent.*

- *$q_0(n, 0 \ldots 0) \xrightarrow{*}_+ q_f(\boldsymbol{0})$ in $\mathcal{A}$.*
- *$L(\mathcal{B}, q_\mathcal{B}(n)) \cup L(\mathcal{B}, q_z(0))$ is not universal.*

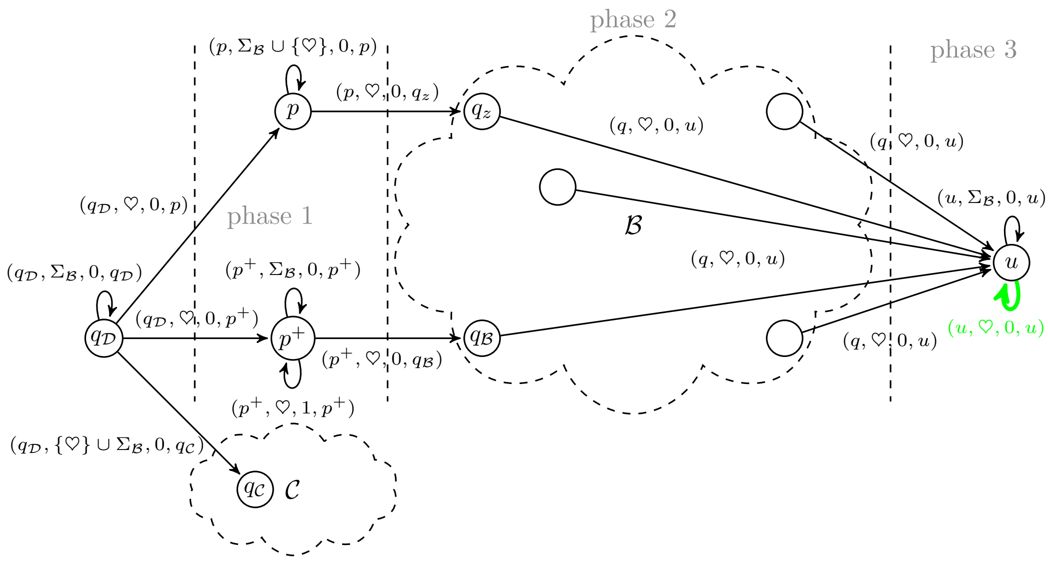

Figure 2 Picture of the one counter net $\mathcal{D}$. The green thick transitions are accepting (the accepting transitions in $\mathcal{C}$ are hidden). Transitions that have a set of letters instead of a single letter, like $(q_{\mathcal{D}}, \Sigma_{\mathcal{B}}, 0, q_{\mathcal{D}})$, denotes a set of transitions, one for every element of the set of letters.

▶ Remark. Note that all transitions in the constructed net $\mathcal{B}$ are accepting. Thus, if a word w does not belong to the language $L(\mathcal{B}, q_z(0)) \cup L(\mathcal{B}, q_{\mathcal{B}}(n))$, then there is no run for the word w in $\mathcal{B}$ that starts in $q_z(0)$ or in $q_{\mathcal{B}}(n)$. This will be used in the proof of Claim 12.

Lemma 8 and Lemma 9 immediately imply the following undecidability result.

▶ **Corollary 10.** *Given a 1-counter net $\mathcal{B} = (Q_{\mathcal{B}}, \Sigma_{\mathcal{B}}, \delta_{\mathcal{B}}, q_{\mathcal{B}}, \delta_{\mathcal{B}})$ and a state $q_z \in Q_{\mathcal{B}}$ it is undecidable if for all $n \in \mathbb{N}$ we have that $L(\mathcal{B}, q_{\mathcal{B}}(n)) \cup L(\mathcal{B}, q_z(0))$ is not universal.*

4.1 Proof of Theorem 3

We reduce the problem proven to be undecidable in Corollary 10 to ω-universality of 1-counter nets.

Formally, for a given 1-counter net $\mathcal{B} = (Q_{\mathcal{B}}, \Sigma_{\mathcal{B}}, \delta_{\mathcal{B}}, q_{\mathcal{B}}, \delta_{\mathcal{B}})$ we construct a 1-counter net $\mathcal{D} = (Q_{\mathcal{D}}, \Sigma_{\mathcal{D}}, \delta_{\mathcal{D}}, q_{\mathcal{D}}, F_{\mathcal{D}})$ such that the ω-language $L_\omega(\mathcal{D}, q_{\mathcal{D}}(0))$ is universal if, and only if, $L(\mathcal{B}, q_{\mathcal{B}}(n)) \cup L(\mathcal{B}, q_z(0))$ is universal for some $n \in \mathbb{N}$.

We first define the following gadget: given the finite alphabet $\Sigma_{\mathcal{B}}$ and a symbol $\heartsuit \notin \Sigma_{\mathcal{B}}$ we construct the Büchi 1-counter net $\mathcal{C} = (Q_{\mathcal{C}}, \Sigma_{\mathcal{B}} \cup \{\heartsuit\}, \delta_{\mathcal{C}}, q_{\mathcal{C}}, F_{\mathcal{C}})$ such that $L_\omega(\mathcal{C}, q_{\mathcal{C}}(0)) = (\Sigma_{\mathcal{B}} \cup \{\heartsuit\})^* \Sigma_{\mathcal{B}}^\omega$, which is in fact even an ω-regular language.

Define $\mathcal{C}$ as follows:

$$\begin{aligned}
Q_{\mathcal{C}} &= \{q_{\mathcal{C}}, q_1\} \\
\delta_{\mathcal{C}} &= \{(q_{\mathcal{C}}, a, 0, q_{\mathcal{C}}),\ (q_{\mathcal{C}}, a, 0, q_1),\ (q_1, a, 0, q_1) \mid a \in \Sigma_{\mathcal{B}}\} \cup \{(q_{\mathcal{C}}, \heartsuit, 0, q_{\mathcal{C}})\} \\
F_{\mathcal{C}} &= \{(q_1, a, 0, q_1) \mid a \in \Sigma_{\mathcal{B}}\}.
\end{aligned}$$

It is easy to check that the language accepted by $\mathcal{C}$ is indeed $(\Sigma_{\mathcal{B}}^* \heartsuit)^* \Sigma_{\mathcal{B}}^\omega$.

Figure 2 depicts the construction of $\mathcal{D}$. The idea is that we introduce a special symbol $\heartsuit$ such that the only relevant words are those that contain infinitely many $\heartsuit$'s. We filter out all words with finite number of $\heartsuit$ using the gadget $\mathcal{C}$. Next, the net $\mathcal{D}$ has three phases between which we switch reading the symbol $\heartsuit$. The first phase is responsible for setting up the inputs to the net $\mathcal{B}$. It reads $\heartsuit$'s and nondeterministically decides to increment the counter or go to the second phase. The second phase tests if $L(\mathcal{B}, q_{\mathcal{B}}(n)) \cup L(\mathcal{B}, q_z(0))$ is not universal for a given n. The third phase is to accept if it happens that for a given n the language $L(\mathcal{B}, q_{\mathcal{B}}(n)) \cup L(\mathcal{B}, q_z(0))$ is universal. Now observe that if there is some $n \in \mathbb{N}$ such that $L(\mathcal{B}, q_{\mathcal{B}}(n)) \cup L(\mathcal{B}, q_z(0))$ is universal, then for every run with infinitely many occurrences of $\heartsuit$ we can increment the counter to the value n then go to the second phase in which we cannot get stuck due to the universality of $L(\mathcal{B}, q_{\mathcal{B}}(n)) \cup L(\mathcal{B}, q_z(0))$. Finally, seeing the next $\heartsuit$ we jump to the third phase where we accept. Conversely, if for all $n \in \mathbb{N}$ we have $L(\mathcal{B}, q_{\mathcal{B}}(n)) \cup L(\mathcal{B}, q_z(0))$ is not universal, then one can build an ω-word such that no matter how long we wait in the first phase incrementing the counter, when we decide to jump to the second phase we will get stuck in it before having the possibility to jump to phase 3; so there is no possibility of accepting this specially designed word.

Formally, the 1-counter net $\mathcal{D}$ is defined as follows:

$$
\begin{aligned}
Q_{\mathcal{D}} &= \{q_{\mathcal{D}}, p, p^+, u\} \cup Q_{\mathcal{C}} \cup Q_{\mathcal{B}} \\
\Sigma_{\mathcal{D}} &= \Sigma_{\mathcal{B}} \cup \{\heartsuit\} \\
\delta_{\mathcal{D}} &= \delta_{\mathcal{C}} \cup \delta_{\mathcal{B}} \\
&\quad \cup \{(u, a, 0, u), (p, a, 0, p), (p^+, a, 0, p^+), (q_{\mathcal{D}}, a, 0, q_{\mathcal{D}}) \mid a \in \Sigma_{\mathcal{B}}\} \\
&\quad \cup \{(u, \heartsuit, 0, u), (p, \heartsuit, 0, p), (p^+, \heartsuit, 1, p^+)\} \\
&\quad \cup \{(q_{\mathcal{D}}, \heartsuit, 0, p), (q_{\mathcal{D}}, \heartsuit, 0, p^+), (q_{\mathcal{D}}, \heartsuit, 0, q_{\mathcal{C}})\} \\
&\quad \cup \{(p, \heartsuit, 0, q_z), (p^+, \heartsuit, 0, q_{\mathcal{B}})\} \\
&\quad \cup \{(q, \heartsuit, 0, u) \mid q \in Q_{\mathcal{B}}\} \\
F_{\mathcal{D}} &= F_{\mathcal{C}} \cup \{(u, \heartsuit, 0, u)\}
\end{aligned}
$$

We start with a general observation on $\mathcal{D}$. The language $L_\omega(\mathcal{D}, q_{\mathcal{D}}(0))$ can be partitioned into two sets, each of them being accepted by a different subset of the set of accepting transitions.

1. The set of words in which the letter $\heartsuit$ appears finitely often. They are indeed all accepted by $\mathcal{D}$, namely by runs in which one of the transitions $(q_{\mathcal{D}}, \heartsuit, 0, q_{\mathcal{C}})$ or $(q_{\mathcal{D}}, a, 0, q_{\mathcal{C}})$ for $a \in \Sigma_{\mathcal{B}}$ is used, finally allowing transitions in $F_{\mathcal{C}}$ to be used infinitely often.
2. The set of words in which the letter $\heartsuit$ appears infinitely often. In this case acceptance has to be due to an infinite number of occurrences of the transition $(u, \heartsuit, 0, u)$.

Due to case 1 universality of $L_\omega(\mathcal{D}, q_{\mathcal{D}}(0))$ holds if, and only if, $(\Sigma_{\mathcal{B}}^* \heartsuit)^\omega \subseteq L_\omega(\mathcal{D}, q_{\mathcal{D}}(0))$.

▶ **Claim 11.** *If $L(\mathcal{B}, q_{\mathcal{B}}(n)) \cup L(\mathcal{B}, q_z(0))$ is universal for some $n \in \mathbb{N}$, then $(\Sigma_{\mathcal{B}}^* \heartsuit)^\omega \subseteq L_\omega(\mathcal{D}, q_{\mathcal{D}}(0))$.*

Let $w \in (\Sigma_{\mathcal{B}}^* \heartsuit)^\omega$, i.e. w contains infinitely many $\heartsuit$'s. Let w' be the infix of w that is strictly in between the $(n+2)$-th and $(n+3)$-th occurrence of the letter $\heartsuit$. Due to our assumption w' belongs to $L(\mathcal{B}, q_z(0))$ or to $L(\mathcal{B}, q_{\mathcal{B}}(n))$.

If $w' \in L(\mathcal{B}, q_z(0))$, then w is accepted by the following run. We wait for the first $\heartsuit$ in state $q_{\mathcal{D}}$ and then we use transition $(q_{\mathcal{D}}, \heartsuit, 0, p)$. Next, we loop in the state p until the $(n+2)$-th occurrence of the letter $\heartsuit$ on which we move to the configuration $q_z(0)$ using the transition $(p, \heartsuit, 0, q_z)$. At this point, we follow an accepting run in $\mathcal{B}$ starting from $q_z(0)$ for the word w' and then on the next occurrence of $\heartsuit$ we move to state u via the transition $(q, \heartsuit, 0, u)$ for some $q \in Q_{\mathcal{B}}$. Finally, in u we accept by performing the transition $(u, \heartsuit, 0, u)$ infinitely often.

If $w' \in L(\mathcal{B}, q_{\mathcal{B}}(n))$, then w is accepted by the following run. We wait for the first occurrence of $\heartsuit$ in state $q_{\mathcal{D}}$. Next we use transition $(q_{\mathcal{D}}, \heartsuit, 0, p^+)$,then we stay in state p^+ just before the $(n+2)$-th occurrence of $\heartsuit$, increasing the counter value to n. Now on upon reading $\heartsuit$ we move to the configuration $q_{\mathcal{B}}(n)$ using transition $(p^+, \heartsuit, 0, q_{\mathcal{B}})$. At this moment we follow an accepting run in $\mathcal{B}$ starting from $q_{\mathcal{B}}(n)$ for the word w' and then on the next occurrence of $\heartsuit$ we move to state u via the transition $(q, \heartsuit, 0, u)$ for some $q \in Q_{\mathcal{B}}$. Finally, in u we accept performing transition $(u, \heartsuit, 0, u)$ infinitely often. This finishes the proof of the claim.

▶ **Claim 12.** *If $L(\mathcal{B}, q_{\mathcal{B}}(n)) \cup L(\mathcal{B}, q_z(0))$ is not universal for all $n \in \mathbb{N}$, then $(\Sigma_{\mathcal{B}}^* \heartsuit)^\omega \not\subseteq L_\omega(\mathcal{D}, q_{\mathcal{D}}(0))$.*

First, observe for every $n \in \mathbb{N}$ there is a finite word $w_n \in \Sigma_{\mathcal{B}}^*$ such that $w_n \notin L(\mathcal{B}, q_z(0)) \cup L(\mathcal{B}, q_{\mathcal{B}}(n))$.

Let $v_\omega = \heartsuit\heartsuit w_0 \heartsuit w_1 \heartsuit w_2 \heartsuit \cdots$. Since v_ω contains infinitely many $\heartsuit$'s, it can only be accepted by infinitely many transitions of the form $(u, \heartsuit, 0, u)$, due to construction of $\mathcal{D}$. Moreover, a run using the transition $(u, \heartsuit, 0, u)$ infinitely often must exactly once use either $(p, \heartsuit, 0, q_{\mathcal{B}})$ or $(p^+, \heartsuit, 0, q_z)$. Each of these two transitions read the letter $\heartsuit$. For the sake of contradiction, suppose $v_\omega \in L_\omega(\mathcal{D}, q_{\mathcal{D}}(0))$ is accepted by the run ρ. We split ρ into three parts $\rho = \rho_0\rho_1\rho_2$, where ρ_0 is the longest prefix of ρ that does not contain $(p, \heartsuit, 0, q_{\mathcal{B}})$ or $(p^+, \heartsuit, 0, q_z)$, ρ_1 is either of the latter two transitions (length one), and ρ_2 is a remaining infinite suffix. Suppose, the symbol $\heartsuit$ appears $(i+2)$ times in $\textsc{Read}(\rho_0)$, then the configuration of the net $\mathcal{B}$ after reading ρ_0 can be $p(0)$ or $p^+(i)$. So after reading $\rho_0\rho_1$ the run has reached either $q_z(0)$ or $q_{\mathcal{B}}(i)$. Moreover we have $\textsc{Read}(\rho_2) = w_i \heartsuit w_{i+1} \ldots$. However, we know that there is no run for word w_i in the net $\mathcal{B}$, neither one starting from $q_z(0)$ nor one starting from $q_{\mathcal{B}}(i)$, thus ρ_2 cannot be a run in $\mathcal{D}$ starting from $q_z(0)$ or $q_{\mathcal{B}}(i)$. Concluding, we have established a contradiction to the fact that ρ was an accepting run. Thus, $v_\omega (\Sigma_{\mathcal{B}}^* \heartsuit)^\omega \setminus L_\omega(\mathcal{D}, q_{\mathcal{D}}(0))$.

5 Conclusion and outlook

In this paper we have shown that the following two problems are undecidable on infinite words although being decidable finite words. First, we showed that the equivalence problem of deterministic Büchi 1-counter automata is undecidable (Theorem 2). Second, we have shown that the universality problem for Büchi 1-counter nets is undecidable (Theorem 3) by a sequence of reductions from the boundedness problem for incremental-error k-counter machines.

The exact recursive complexity of both problems is yet unclear to us and is planned to be investigated in the full version of this paper.

References

1 Rajeev Alur and P. Madhusudan. Visibly pushdown languages. In *Proc. of STOC*, pages 202–211. ACM, 2004.

2 Stanislav Böhm, Stefan Göller, and Petr Jančar. Equivalence of deterministic one-counter automata is NL-complete. In *Proc. of STOC*, pages 131–140, Palo Alto, California, USA, June 2013. ACM Press. URL: `http://www.lsv.ens-cachan.fr/Publis/PAPERS/PDF/BGJ-stoc13.pdf`, `doi:10.1145/2488608.2488626`.

3 Didier Caucal, Dung T. Huynh, and Lu Tian. Deciding Branching Bimiliarity of Normed Context-Free Processes Is in Σ_2^p. *Inf. Comput.*, 118(2):306–315, 1995.

4 Yuxi Fu. Checking Equality and Regularity for Normed BPA with Silent Moves. In *Proc. of ICALP*, Lecture Notes in Computer Science. Springer, 2013. to appear.

5 Yoram Hirshfeld and Mark Jerrum. Bisimulation equivanlence is decidable for normed process algebra. In *Proc. of ICALP*, volume 1644 of *Lecture Notes in Computer Science*, pages 412–421. Springer, 1999.

6 Yoram Hirshfeld, Mark Jerrum, and Faron Moller. A polynomial algorithm for deciding bisimilarity of normed context-free processes. *Theoretical Computer Science*, 158(1&2):143–159, 1996.

7 Piotr Hofman and Patrick Totzke. Trace inclusion for one-counter nets revisited. In *Proc. of Reachability Problems*, pages 151–162, 2014.

8 Petr Jancar. Equivalences of pushdown systems are hard. In *Foundations of Software Science and Computation Structures – 17th International Conference, FOSSACS 2014, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2014, Grenoble, France, April 5-13, 2014, Proceedings*, pages 1–28, 2014. doi:10.1007/978-3-642-54830-7_1.

9 Christof Löding. Simplification Problems for Deterministic Pushdown Automata on Infinite Words. *Int. J. Found. Comput. Sci.*, 26(8):1041–1068, 2015.

10 Christof Löding and Stefan Repke. Regularity Problems for Weak Pushdown ω-Automata and Games. In *In Proc. of MFCS*, pages 764–776. Springer, 2012.

11 M. L. Minsky. *Computation: Finite and Infinite Machines.* Prentice-Hall International, Englewood Cliffs, 1967.

12 Dirk Nowotka and Jirí Srba. Height-Deterministic Pushdown Automata. In *Proc. of MFCS*, volume 4708 of *Lecture Notes in Computer Science*, pages 125–134. Springer, 2007.

13 Joël Ouaknine and James Worrell. On Metric Temporal Logic and Faulty Turing Machines. In *In Proc. of FoSSaCSs*, volume 3921 of *Lecture Notes in Computer Science*, pages 217–230. Springer, 2006.

14 Joël Ouaknine and James Worrell. On the decidability and complexity of Metric Temporal Logic over finite words. *Logical Methods in Computer Science*, 3(1), 2007.

15 S. Repke. *Simplification problems for automata and games.* PhD thesis, RWTH Aachen, 2014.

16 Philippe Schnoebelen. Lossy Counter Machines Decidability Cheat Sheet. In Antonín Kučera and Igor Potapov, editors, *Proc. of Reachability Problems*, pages 51–75. Springer, 2010.

17 Géraud Sénizergues. L(A)=L(B)? decidability results from complete formal systems. *Theor. Comput. Sci.*, 251(1-2):1–166, 2001.

18 Colin Stirling. Deciding DPDA Equivalence Is Primitive Recursive. In *Proc. of ICALP*, volume 2380 of *Lecture Notes in Computer Science*, pages 821–832. Springer, 2002.

19 Leslie G. Valiant. Regularity and related problems for deterministic pushdown automata. *J. ACM*, 22(1):1–10, January 1975.