On the generating functions of languages accepted by deterministic one-reversal counter machines

Paolo Massazza[1]

Università degli Studi dell'Insubria, Dipartimento di Scienze Teoriche e Applicate - Sezione Informatica, Via Mazzini 5, 21100 Varese, Italy
paolo.massazza@uninsubria.it

Abstract. We prove that the generating function of a language accepted by a one-way deterministic one-reversal counter machine without negative cycles is holonomic. The result is achieved by solving a particular case of the conjecture $\mathcal{L}_{\mathsf{DFCM}} \subsetneq \mathsf{RCM}$. Here, RCM is a class of languages that has been recently introduced and that admits some interesting properties, namely it contains only some particular languages with holonomic generating function.

1 Introduction

A well-known result of Chomsky-Schützenberger [4] states that the generating functions of regular languages are rational whereas the generating functions of unambiguous context-free languages are algebraic. This fact allows us to use analytic methods to determine properties of languages. For example, a method to show that a context-free language L is inherently ambiguous, employed by Flajolet in [5] and [6], consists of proving that the generating function of L is transcendental. It is then interesting to look for classes of languages with generating functions that belong to classes of functions whose properties can be exploited for solving classical problems in language theory.

In this context, holonomic functions have been widely investigated since the end of 1980s. The class of holonomic functions in one variable is an extension of the class of algebraic functions, and it contains all functions satisfying a homogeneous linear differential equation with polynomial coefficients (see [12, 13]). Holonomic functions were first used in the context of formal languages in [1], where the authors proved that the problem of deciding the holonomicity of the generating function of a context-free language is equivalent to the problem of deciding whether a context-free language is inherently ambiguous. Furthermore, a class of languages with holonomic generating functions, called LCL, was introduced in [10] by means of linear constraints on the number of occurrences of symbols of the alphabet. A particular subclass $\mathsf{LCL}_R \subsetneq \mathsf{LCL}$ was also studied in [1]. The idea of using constraints and finite state automata in order to define languages is also at the basis of a family of automata called Parikh Automata and defined in [8, 9]. In particular, the subclass LPA of Parikh Automata on letters has been defined in [2] (actually, as noted in [3], $\mathcal{L}_{\mathsf{LPA}} = \mathsf{LCL}_R$). Recently, in [3] a wider class of languages with holonomic generating functions, called RCM, has been defined. This class of languages is contained in $\mathcal{L}_{\mathsf{NFCM}}$, *i.e.* the class of languages recognized by nondeterministic one-way reversal bounded counter machines, whereas it is not contained in $\mathcal{L}_{\mathsf{DFCM}}$, *i.e.* the class of languages recognized by deterministic one-way reversal bounded counter machines [7]. Lastly, in [3] the conjecture $\mathcal{L}_{\mathsf{DFCM}} \subsetneq \mathsf{RCM}$ has been stated.

In this paper we prove that the conjecture is true for the subclass $\mathcal{L}_{\mathsf{DFCM}_{\emptyset}}$ consisting of the languages accepted by deterministic one-reversal counter machines without negative cycles (informally, on reading a symbol the automaton can decrement a counter by a value bounded by a constant). The result is obtained by generalizing the technique used in [11], where it is proved that the class $\mathcal{L}_{\mathsf{DFCM}(1,0,1)}$ of languages accepted by deterministic counter machines with one-way input tape and one counter that is one-reversal bounded is contained in RCM. We recall that for any class of languages $\mathcal{L}$, the relation $\mathcal{L} \subseteq \mathsf{RCM}$ implies that the generating function of a language in $\mathcal{L}$ is holonomic. This provides a method for proving that a language L is not in $\mathcal{L}$, which resembles in some sense the Flajolet methodology, used when $\mathcal{L}$ is the class of unambiguous context free languages.

2 Preliminaries

In this section we give some basics about languages, classes of languages and automata of our interest in the paper. Let $\Sigma = \{\sigma_1, \sigma_2, \ldots, \sigma_h\}$ be a finite alphabet and $w \in \Sigma^\star$. For all $\sigma \in \Sigma$ we indicate by $|w|_\sigma$ the number of occurrences of σ in w. The *length* of w is $|w| = \sum_{\sigma \in \Sigma} |w|_\sigma$. The *prefix* of w consisting of the first h symbols is $w_{\leq h}$. Similarly, $w_{>h}$ is the *suffix* of w starting at $h+1$. Given two finite alphabets Γ and Σ, a morphism $\mu : \Gamma^\star \mapsto \Sigma^\star$ is said to be *length preserving* if for all $w \in \Gamma^\star$ one has $|\mu(w)| = |w|$. In particular, we are interested in length preserving morphisms that are injective on a fixed language $L \subseteq \Gamma^\star$, that is, morphisms μ such that $v \neq w$ implies $\mu(v) \neq \mu(w)$. For any $k > 0$, we also consider a morphism $\kappa : \mathbb{N}^k \mapsto \{0,1\}^k$ defined by $\kappa(i) = 1$ if $i \neq 0$ and $\kappa(0) = 0$. From here on, boldface symbols indicate tuples of integer values, and $\boldsymbol{c}[i]$ denotes the i-th element of $\boldsymbol{c}$. Furthermore, if $\boldsymbol{c}, \boldsymbol{d} \in \mathbb{N}^k$, then $\boldsymbol{c} + \boldsymbol{d}$ is the their sum (componentwise).

Linear constraints on the number of occurrences of symbols in an alphabet have been used in [10, 3] to define two classes of languages with holonomic generating functions, called LCL and RCM, respectively.

Definition 1 (linear constraint). *A linear constraint on the occurrences of symbols of $\Gamma = \{\gamma_1, \gamma_2, \ldots, \gamma_h\}$ in $w \in \Gamma^\star$ is an expression of the form*

$$\sum_{i=1}^{h} c_i |w|_{\gamma_i} \bigtriangleup c_{h+1}, \qquad \textit{with } c_i \in \mathbb{Z}, \bigtriangleup \in \{<, \leq, =, \neq, \geq, >\}.$$

Definition 2 (system of linear constraints). *A system of linear constraints C is either a linear constraint, or $C_1 \vee C_2$ or $C_1 \wedge C_2$ or $\neg C_1$, where C_1 and C_2 are systems of linear constraints.*

We denote by $[C]$ the language consisting of the words in $\Gamma^\star$ that satisfy the system of linear constraints C. Let L be a language on Γ, C a system of linear constraints on the number of occurrences of symbols in Γ and $\mu : \Gamma^\star \mapsto \Sigma^\star$ a morphism. We indicate by $\langle L, C, \mu\rangle$ the language $\mu(L \cap [C]) \subseteq \Sigma^\star$. In [3] the class of languages RCM has been defined as follows.

Definition 3 (RCM). RCM *is the class of languages $\langle R, C, \mu\rangle$ where R is a regular language on an alphabet Γ, C a system of linear constraints on Γ and $\mu : \Gamma^\star \mapsto \Sigma^\star$ a length preserving morphism that is injective on $R \cap [C]$.*

The class RCM admits several interesting properties. Indeed, it is closed under union and intersection, and it contains languages with holonomic generating function. Moreover, most of the classical decision problems (*i.e.* equivalence, inclusion, disjointness, emptiness, universe) are decidable for RCM, see [3].

2.1 Counter machines

In Section 4, the class RCM will be compared to the class of languages accepted by a particular family of counter machines. We recall that a two-way k-counter machine is a finite automaton equipped with k counters. The operations admitted on a counter are the increment or the decrement by 1, as well as the comparison with 0. The machine is called l-reversal bounded (l-reversal for short) if the count in each counter alternately increases and decreases at most l times. We refer to [7] for all definitions and for main results concerning the class $\mathsf{DFCM(k,m,n)}$ of *deterministic (m,n)-reversal bounded k-counter machines*, that is, n-reversal k-counter machines with a two-way input tape, where the input head reverses direction at most m times. In particular, we are interested in the class $\mathsf{DFCM(k,0,1)}$ where the input tape is one-way and the counters can change from increasing to decreasing mode at most once. Formally, a machine $M \in \mathsf{DFCM(k,0,1)}$ is a 7-tuple $M = (k, Q, \Sigma, \$, \delta, \dot{q}, F)$, where k indicates the number of counters, Q is a finite set of states, Σ is the input alphabet, \$ is the right end-marker, δ is the transition function, $\dot{q} \in Q$ is the initial state and $F \subseteq Q$ is the set of final states. The transition function is a mapping from $Q \times (\Sigma \cup \{\$\}) \times \{0,1\}^k$ into $Q \times \{S, R\} \times \{-1, 0, +1\}^k$ such that if $\delta(q, a, c_1, \ldots, c_k) = (p, d, d_1, \ldots, d_k)$ and $c_i = 0$ for some i, then d_i has to be nonnegative to prevent negative values in a counter. The symbols S and R are used to indicate the movement of the input tape head ($S =$ stay, $R =$ right).

A configuration of M is a triple $(q, x\$, \boldsymbol{c})$ where $q \in Q$, $x \in \Sigma^*$ is the unread suffix of the input word, and the content of the k counters is $\boldsymbol{c} \in \mathbb{N}^k$. The transition relation on the set of configurations is denoted by $\to$, and its reflexive and transitive closure by $\xrightarrow{*}$. Hence, we write $(p, v, \boldsymbol{c}) \to (q, z, \boldsymbol{c}')$ if and only if $\delta(p, \sigma, \kappa(\boldsymbol{c})) = (q, \boldsymbol{b})$, $\boldsymbol{c}' = \boldsymbol{c} + \boldsymbol{b}$ and either $v = \sigma z$ (if $d = R$) or $z = v$ (if $d = S$). We are also interested in the relation $\Rightarrow$, called *one-symbol transition.*

Definition 4 ($\Rightarrow$). *Let $M = (k, Q, \Sigma, \$, \delta, \dot{q}, F) \in \mathsf{DFCM(k, 0, 1)}$. For any $x \in \Sigma^\star$ and $\sigma \in \Sigma$ we write $(p, \sigma x\$, \boldsymbol{c}) \Rightarrow (q, x\$, \boldsymbol{c'})$ if and only if $p, q \in Q$, and either $\delta(p, \sigma, \kappa(\boldsymbol{c})) = (q, R, \boldsymbol{d})$ with $\boldsymbol{c'} = \boldsymbol{c} + \boldsymbol{d}$, or $\delta(p, \sigma, \kappa(\boldsymbol{c})) = (q_1, S, \boldsymbol{d_1})$, $\delta(q_1, \sigma, \kappa(\boldsymbol{c} + \boldsymbol{d_1})) = (q_2, S, \boldsymbol{d_2}), \ldots, \delta(q_h, S, \kappa(\boldsymbol{c} + \sum_{i=1..h} \boldsymbol{d_i})) = (q, R, \boldsymbol{d_{h+1}})$, with $\boldsymbol{c'} = \boldsymbol{c} + \sum_{i=1..h+1} \boldsymbol{d}_i$.*

Notice that the transition $(p, \sigma x\$, \boldsymbol{c}) \Rightarrow (q, x\$, \boldsymbol{c'})$ uniquely identifies a sequence $\{\boldsymbol{d}_i\}$ of tuples in $\{-1, 0, 1\}^k$ and a sequence $\{q_i\}$ of states in Q. A sequence of $|w|$ one-symbol transitions reading a word w is shortened as $(p, wx\$, \boldsymbol{c}) \overset{|w|}{\Rightarrow} (q, x\$, \boldsymbol{c'})$. A word $w \in \Sigma^\star$ is accepted by M if and only if $(\dot{q}, w\$, 0^k) \overset{|w|}{\Rightarrow} (p, \$, \boldsymbol{c}) \xrightarrow{*} (q, \$, \boldsymbol{c'})$, for some $q \in F$ and suitable $\boldsymbol{c}, \boldsymbol{c'} \in \mathbb{N}^k$.The language accepted by M, denoted by $L(M)$, is the set of all words accepted by M. Without loss of generality, we suppose that M always terminates and has only one final state, denoted by $\ddot{q}$, and that a word is accepted with all the counters equal to 0. The only accepting configuration is then $(\ddot{q}, \$, 0^k)$. In the following, we consider only deterministic 1-reversal counter machines that do not admit negative cycles. This class of machines is denoted by $\mathsf{DFCM}_{\not\circlearrowleft}\mathsf{(k, 0, 1)}$.

Definition 5 (negative cycle). *Let $M = (k, Q, \Sigma, \$, \delta, \dot{q}, \{\ddot{q}\})$ be a counter machine in $\mathsf{DFCM(k, 0, 1)}$. Then, M has a* negative cycle *if there exists a sequence of states $q_1, \ldots, q_h \in Q$, a symbol $\sigma \in \Sigma$ and a suitable $\boldsymbol{b} \in \{0, 1\}^k$ such that:*

- $\delta(q_1, \sigma, \boldsymbol{b}) = (q_2, S, \boldsymbol{d}_1), \delta(q_2, \sigma, \boldsymbol{b}) = (q_3, S, \boldsymbol{d}_2), \ldots, \delta(q_h, \sigma, \boldsymbol{b}) = (q_1, S, \boldsymbol{d}_h)$
- *$\boldsymbol{d}[l] < 0$ for at least one l, with $1 \leq l \leq k$, where $\boldsymbol{d} = \sum_{i=1..h} \boldsymbol{d}_i$;*
- *if $\boldsymbol{d}[l] < 0$ then $\boldsymbol{b}[l] = 1$ and $\boldsymbol{d}_i[l] \leq 0$ for all i;*
- *if $\boldsymbol{d}[l] > 0$ then $\boldsymbol{b}[l] = 1$ and $\boldsymbol{d}_i[l] \geq 0$ for all i;*
- *if $\boldsymbol{d}[l] = 0$ then $\boldsymbol{d}_i[l] = 0$ for all i;*

The k-tuple $\boldsymbol{d}$ is called the weight *of the cycle.*

In a machine $M \in \mathsf{DFCM}_{\not\circlearrowleft}\mathsf{(k, 0, 1)}$, the effect on the counters of any one-symbol transition is bounded. More precisely, the following lemma holds.

Lemma 1. *Let $(k, Q, \Sigma, \$, \delta, \dot{q}, \{\ddot{q}\}) \in \mathsf{DFCM}_{\not\circlearrowleft}\mathsf{(k, 0, 1)}$. Then, for any one-symbol transition $(p, \sigma x\$, \boldsymbol{c}) \Rightarrow (q, x\$, \boldsymbol{c} + \boldsymbol{d})$ one has $0 \leq |\boldsymbol{d}[l]| \leq (3k + 1)|Q|$ for all l, with $1 \leq l \leq k$.*

In particular, a counter machine that always terminates can not have a positive cycle (defined as in Def. 5, with the only difference that for all i and l one has $\boldsymbol{d}_i[l] \geq 0$, and $\boldsymbol{d}[l] > 0$ for at least one l).

At each step of the computation of a deterministic 1-reversal k-counter machine, each counter is exactly in one of four different states, denoted by a value in the set $U = \{0, 1, 2, 3\}$ and called the *global state* of the counter. More precisely, 0 is associated with a zero counter that has not been increased yet, 1 is associated with a counter that has been increased but not decreased, 2 is associated with a counter that has been increased and decreased and it is still greater than zero and, finally, 3 is associated with a counter that has been increased and decreased and it is equal to zero. Obviously, the global state of a counter may change from i to j, with $i \leq j$, but not vice versa, hence the ordering $0 < 1 < 2 < 3$ naturally arises. During a computation of a counter machine with k counters, a sequence of strings in U^k is used to represent the evolution of the global states of the counters. The set U^k is equipped with a partial order $\prec$, which is defined as follows: given $\boldsymbol{\alpha}, \boldsymbol{\beta} \in U^k$, define $\boldsymbol{\alpha} \prec \boldsymbol{\beta}$ if and only if $\boldsymbol{\alpha}[i] \leq \boldsymbol{\beta}[i]$ for all i with $1 \leq i \leq k$. So, if $\boldsymbol{\alpha}_0, \boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_r$ is the sequence of global states of the counters of a machine that reads an input word $w \in \Sigma^\star$, then one has $\boldsymbol{\alpha}_0 \prec \boldsymbol{\alpha}_1 \cdots \prec \boldsymbol{\alpha}_r$, with $\boldsymbol{\alpha}_0 = 0^k$ (and $\boldsymbol{\alpha}_r \in \{3, 0\}^k$ if w is accepted). Since the machine is reversal, there are at most $3k+1$ different global states in the sequence $\boldsymbol{\alpha}_0, \boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_r$. In other words, in the poset $(U^k, \prec)$ any chain has a length which is smaller than or equal to $3k$.

Let $\nu : U^k \mapsto \{0,1\}^k$ be the morphism defined by $\nu(0) = \nu(3) = 0$ and $\nu(1) = \nu(2) = 1$. If $\boldsymbol{\alpha}$ is the global state of the counters in a given configuration $(q, \sigma x\$, \boldsymbol{c})$, then one has $\nu(\boldsymbol{\alpha}) = \kappa(\boldsymbol{c})$.

A sequence $\{\boldsymbol{d}_i\}$ of tuples in $\{-1, 0, 1\}^k$ is called *1-reversal acceptable* if and only if for all l, with $1 \leq l \leq k$, one has that $\boldsymbol{d}_i[l] = -1$ implies $\boldsymbol{d}_j[l] \leq 0$ for all $j > i$. Moreover, $\{\boldsymbol{d}_i\}$ is *compatible* with $\boldsymbol{\alpha} \in U^k$ if it is 1-reversal acceptable and for all l, with $1 \leq l \leq k$, one has:

- if $\boldsymbol{\alpha}[l] = 3$ then $\forall i\ \boldsymbol{d}_i[l] = 0$;
- if $\boldsymbol{\alpha}[l] = 2$ then $\forall i\ \boldsymbol{d}_i[l] \leq 0$;
- if $\boldsymbol{\alpha}[l] = 0$ then $|\{i|\boldsymbol{d}_i[l] = -1\}| \leq |\{i|\boldsymbol{d}_i[l] = 1\}|$.

Furthermore, given $\boldsymbol{\alpha}, \boldsymbol{\beta} \in U^k$ with $\boldsymbol{\alpha} \prec \mathbf{b}$, we say that a sequence $\{\boldsymbol{d}_i\}$, *changes* $\boldsymbol{\alpha}$ into $\boldsymbol{\beta}$ if it is compatible with $\boldsymbol{\alpha}$ and for all l, with $1 \leq l \leq k$, the conditions in the following table hold (a dash indicates a case that can not occur, $r_l = |\{i|\boldsymbol{d}_i[l] = -1\}|$, $s_l = |\{i|\boldsymbol{d}_i[l] = 1\}|$). A sequence $\{\boldsymbol{d}_i\}$ that changes $\boldsymbol{\alpha}$ into $\boldsymbol{\alpha}$ is called *stable* w.r.t. $\boldsymbol{\alpha}$.

	$\boldsymbol{\beta}[l] = 0$	$\boldsymbol{\beta}[l] = 1$	$\boldsymbol{\beta}[l] = 2$	$\boldsymbol{\beta}[l] = 3$
$\boldsymbol{\alpha}[l] = 0$	$r_l = s_l = 0$	$r_l = 0 \wedge s_l > 0$	$s_l > r_l > 0$	$r_l > 0 \wedge r_l = s_l$
$\boldsymbol{\alpha}[l] = 1$	–	$r_l = 0$	$r_l > 0$	$r_l - s_l > 0$
$\boldsymbol{\alpha}[l] = 2$	–	–	$s_l = 0$	$s_l = 0$
$\boldsymbol{\alpha}[l] = 3$	–	–	–	$r_l = s_l = 0$

Let $\boldsymbol{\alpha} \in U^k$ be the global state of the counters in a configuration $(p, \sigma x\$, \boldsymbol{c})$. Consider a transition $T = (p, \sigma x\$, \boldsymbol{c}) \Rightarrow (q, x\$, \boldsymbol{c'})$ and its associated finite sequence of increments/decrements $\{\boldsymbol{d}_i\}$, $\boldsymbol{d}_i \in \{-1, 0, 1\}^k$. We say that T is *stable* w.r.t. $\boldsymbol{\alpha}$ if $\{\boldsymbol{d}_i\}$ is stable w.r.t. $\boldsymbol{\alpha}$ (*i.e.* the global state of the counters in $(q, x\$, \boldsymbol{c'})$ is still $\boldsymbol{\alpha}$), whereas T *changes* $\boldsymbol{\alpha}$ into $\boldsymbol{\beta}$, with $\boldsymbol{\alpha} \prec \boldsymbol{\beta}$, if $\{\boldsymbol{d}_i\}$ changes $\boldsymbol{\alpha}$ into $\boldsymbol{\beta}$ (*i.e.* the global state of the counters in $(q, x\$, \boldsymbol{c'})$ is $\boldsymbol{\beta}$). We write $\overset{\boldsymbol{\alpha}}{\underset{\boldsymbol{\beta}}{\Rightarrow}}$ (resp., $\overset{\boldsymbol{\alpha}}{\underset{\boldsymbol{\alpha}}{\Rightarrow}}$) for a transition that changes $\boldsymbol{\alpha}$ into $\boldsymbol{\beta}$ (resp., that is stable). The transitive closure of $\overset{\boldsymbol{\alpha}}{\underset{\boldsymbol{\beta}}{\Rightarrow}}$ is $\overset{\boldsymbol{\alpha}}{\underset{\boldsymbol{\beta}}{\Rightarrow}}{}^\star$.

The global state of the counters is used to define suitable subsets of the set of states Q of a 1-reversal k-counter machine. Indeed, for any $\boldsymbol{\beta} \in U^k$ we define the set of states $Q_{\boldsymbol{\beta}}$ as follows.

Definition 6 **($Q_{\boldsymbol{\beta}}$).**

Let $(k, Q, \Sigma, \$, \delta, q_0, F) \in \mathsf{DFCM}(\mathsf{k}, 0, 1)$*. Then,* $Q_{\boldsymbol{\beta}} \subseteq Q$ *is inductively determined as follows:*

($\boldsymbol{\beta} = 0^k$) Q_{0^k} *is the set of states in* Q *that are reachable from* q_0 *by a sequence of transitions that are stable w.r.t.* 0^k*,*

$$Q_{0^k} = \{q \in Q | \exists w \in \Sigma^\star : (q_0, wx\$, \mathbf{0}) \overset{0^k}{\underset{0^k}{\Rightarrow}}{}^\star (q, x\$, \mathbf{0})\};$$

($\boldsymbol{\beta} \neq 0^k$) *Let* $Q'_{\boldsymbol{\beta}} = \{q \in Q | \exists p \in Q_{\boldsymbol{\alpha}}, \sigma \in \Sigma : \boldsymbol{\alpha} \prec \boldsymbol{\beta}, \wedge (p, \sigma x\$, \mathbf{c}) \overset{\boldsymbol{\alpha}}{\underset{\boldsymbol{\beta}}{\Rightarrow}} (q, x\$, \mathbf{c}')\}$*. Then,* $Q_{\boldsymbol{\beta}} = Q'_{\boldsymbol{\beta}} \cup Q''_{\boldsymbol{\beta}}$ *where*

$$Q''_{\boldsymbol{\beta}} = \{q \in Q | \exists w \in \Sigma^\star, p \in Q'_{\boldsymbol{\beta}} : (p, wx\$, \mathbf{c}) \overset{\boldsymbol{\beta}}{\underset{\boldsymbol{\beta}}{\Rightarrow}}{}^\star (q, x\$, \mathbf{c}').$$

Example 1. Figure 1 shows a machine M in $\mathsf{DFCM}_{\emptyset}(2, 0, 1)$. A label of type $\sigma_1, \sigma_2, b_1 b_2/d_1 d_2, D$ indicates a transition on an input symbol in $\{\sigma_1, \sigma_2\}$ and two counters c_1, c_2 satisfying $\kappa(c_1) = b_1$ and $\kappa(c_2) = b_2$ (in Fig. 1 the symbol d stands for any symbol in $\{0, 1\}$). D represents the movement of the input head, and d_1, d_2 the increments/decrements. The only sets of states $Q_{\boldsymbol{\alpha}}$ that are not empty are $Q_{00} = Q_{01} = \{\dot{q}\}$, $Q_{11} = \{\dot{q}, t, u\}$, $Q_{22} = \{u\}$, $Q_{23} = Q_{32} = \{u, v\}$ and $Q_{33} = \{\ddot{q}, u, v, z\}$.

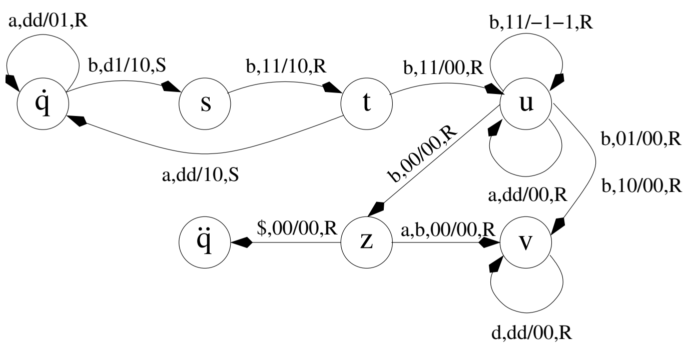

Fig. 1. A machine M in $\mathsf{DFCM}_{\emptyset}(2, 0, 1)$.

In the next section we define a DFA M' whose states are distinguished copies of states in $Q_{\boldsymbol{\alpha}}$, for any $\boldsymbol{\alpha} \in U^k$ with $Q_{\boldsymbol{\alpha}} \neq \emptyset$. The automaton M' has transitions from a state p in $Q_{\boldsymbol{\alpha}}$ to a state q that belongs to $Q_{\boldsymbol{\alpha}}$ or to $Q_{\boldsymbol{\beta}}$, with $\boldsymbol{\alpha} \prec \boldsymbol{\beta}$.

3 The s-automaton

Let $M = (k, Q, \Sigma, \$, \delta, \dot{q}, \{\ddot{q}\}) \in \mathsf{DFCM}_{\emptyset}(\mathsf{k}, 0, 1)$ and consider a triple $(\boldsymbol{\alpha}, p, \sigma)$, with $\boldsymbol{\alpha} \in U^k$, $p \in Q_{\boldsymbol{\alpha}}$ and $\sigma \in \Sigma$. An *evolution* of $(\boldsymbol{\alpha}, p, \sigma)$ is a sequence $\{(p_i, \boldsymbol{\alpha}_i, \boldsymbol{d}_i)\}_{i=1...r}$ such that:

– $\boldsymbol{d}_i \in \{-1,0,1\}^k$, $\boldsymbol{\alpha}_i \in U^k$, $1 \le i \le r$;
– $p_i \in Q_{\boldsymbol{\alpha}_i}$;
– $\boldsymbol{\alpha} \prec \boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_j \prec \boldsymbol{\alpha}_{j+1}$ for all j with $1 \le j < r$;
– for all j, with $1 \le j \le r$, $\{\boldsymbol{d}_i\}_{i=1\ldots j}$ changes $\boldsymbol{\alpha}$ to $\boldsymbol{\alpha}_j$;
– $\delta(p,\sigma,\nu(\boldsymbol{\alpha})) = (p_1, S, \boldsymbol{d}_1)$, $\delta(p_i,\sigma,\nu(\boldsymbol{\alpha}_i)) = (p_{i+1}, S, \boldsymbol{d}_{i+1})$ for $1 \le i < r-1$, and $\delta(p_{r-1},\sigma,\nu(\boldsymbol{\alpha}_{r-1})) = (p_r, R, \boldsymbol{d}_r)$.

We denote by $\mathrm{Ev}(\boldsymbol{\alpha}, p, \sigma)$ the set of all possible evolutions of a given triple $(\boldsymbol{\alpha}, p, \sigma)$. Notice that this set is finite and can be computed in time $O(3k|Q|)$. If $\boldsymbol{\alpha}$ is the global state of the counters in a configuration $(p, \sigma x\$, \boldsymbol{c})$ of M, then it is immediate that a one-symbol transition $(p, \sigma x\$, \boldsymbol{c}) \Rightarrow (q, x\$, \boldsymbol{c}')$ uniquely identifies an evolution $\{(p_i, \boldsymbol{\alpha}_i, \boldsymbol{d}_i)\}_{i=1\ldots r}$ in $\mathrm{Ev}(\boldsymbol{\alpha}, p, \sigma)$ such that $\boldsymbol{\alpha}_r$ is the global state of the counters in the configuration $(q, x\$, \boldsymbol{c}')$ and $\boldsymbol{c}' = \boldsymbol{c} + \boldsymbol{d}$, where $\boldsymbol{d} = \sum_{i=1}^{r} \boldsymbol{d}_i$.

Our aim is that of defining a suitable DFA M' that uses weighted symbols to simulate a machine $M \in \mathsf{DFCM}_{\not\emptyset}(\mathsf{k},0,1)$. The automaton M' (equipped with a suitable set of linear constraints and a morphism) is used to specify a language L in RCM such that $L = L(M)$, see Sect. 4.

In the counter machine M, a one-symbol transition on $\sigma \in \Sigma$ may act differently on different counters, so the alphabet of M' is a suitable alphabet $\Sigma' \ne \Sigma$ that takes into account increments/decrements. So, consider a triple $(\boldsymbol{\alpha}, p, \sigma)$, with $p \in Q_{\boldsymbol{\alpha}}$, and let $\mathrm{Ev}(\boldsymbol{\alpha}, p, \sigma)$ contain an evolution $E = \{(p_i, \boldsymbol{\alpha}_i, \boldsymbol{d}_i)\}_{i=1\ldots r}$, with $\boldsymbol{\alpha}_r[l] = 3$ if and only if $\boldsymbol{\alpha}[l] = 3$. Notice that in E no new counter is set to zero. In this case, a symbol $\sigma_{\boldsymbol{d}}$ (with $\boldsymbol{d} = \sum_i \boldsymbol{d}_i$) is added to Σ' to simulate E. Furthermore, if $\mathrm{Ev}(\boldsymbol{\alpha}, p, \sigma)$ contains an evolution E' where the l counters in $G = \{i_1, i_e, \ldots, i_l\}$ change state from a value lesser than 3 to 3 (*i.e.* $\boldsymbol{\alpha}[i_j] < \boldsymbol{\alpha}_r[i_j] = 3$ and $\boldsymbol{d}[i_j] < 0$ for $1 \le j \le l$), then Σ' should contain a symbol $\sigma_{\boldsymbol{d}}^{G}$. Such a symbol is called *guess-symbol* as it is used by M' to guess that a one-symbol transition of M resets some counters. The weight of a symbol $\sigma_{\boldsymbol{d}}^{G}$ (resp., $\sigma_{\boldsymbol{d}}$) is $W(\sigma_{\boldsymbol{d}}^{G}) = \boldsymbol{d}$ (resp., $W(\sigma_{\boldsymbol{d}}) = \boldsymbol{d}$). All the previous remarks lead to a particular DFA which is called the *s-automaton* associated with M.

Definition 7 (s-automaton). *Let $M = (k, Q, \Sigma, \$, \delta, \dot{q}, \{\ddot{q}\})$ be a counter machine in $\mathsf{DFCM}_{\not\emptyset}(\mathsf{k},0,1)$. The s-automaton associated with M is the deterministic finite state automaton $M' = (Q', \Sigma', \delta', \dot{q}_{0^k}, F')$ where:*

– $Q' = \{q_{\boldsymbol{\alpha}} | \boldsymbol{\alpha} \in U^k, q \in Q_{\boldsymbol{\alpha}}\}$,
– $\Sigma' = \{\sigma_{\boldsymbol{i}}, \sigma_{\boldsymbol{i}}^{G(\boldsymbol{i})} \mid \sigma \in \Sigma, \boldsymbol{i} \in [-c|Q|, c|Q|]^k, c = 3k+1, G(\boldsymbol{i}) \subseteq \{l \mid \boldsymbol{i}[l] < 0\}\}$,
– $F' = \{q_{\boldsymbol{\alpha}} \mid \boldsymbol{\alpha} \in \{0,3\}^k, Q_{\boldsymbol{\alpha}} \ne \emptyset, (q, \$, 0^k) \xrightarrow{\star} (\ddot{q}, \$, 0^k)$ *in* $M\}$,

and $\delta' : Q' \times \Sigma' \mapsto Q'$ is defined as follows. Let $(\boldsymbol{\alpha}, p, \sigma)$ be a triple of M that admits an evolution $E = \{p_i, \boldsymbol{\alpha}_i, \boldsymbol{d}_i\}_{i=1,\ldots,r}$, with $\boldsymbol{d} = \sum_{i=1}^{r} \boldsymbol{d}_i$. If for all l such that $\boldsymbol{\alpha}_r[l] = 3$ one has $\boldsymbol{\alpha}[l] = 3$, then set $\delta'(p_{\boldsymbol{\alpha}}, \sigma_{\boldsymbol{d}}) = q_{\boldsymbol{\alpha}_r}$, where $q = p_r$. Otherwise, let $G = \{j \mid \boldsymbol{d}[j] < 0 \wedge \boldsymbol{\alpha}[j] < \boldsymbol{\alpha}_r[j] = 3\}$ ($G \ne \emptyset$ since in E the global state of at least one counter changes from e to 3, with $e < 3$) and set $\delta'(p_{\boldsymbol{\alpha}}, \sigma_{\boldsymbol{d}}^{G}) = q_{\boldsymbol{\alpha}_r}$.

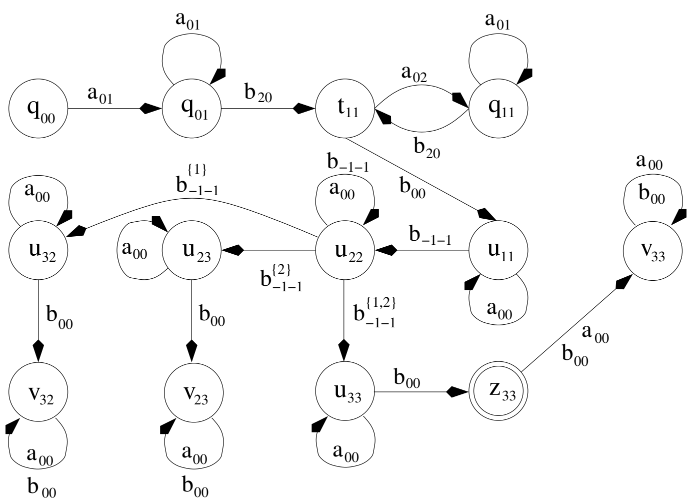

Fig. 2. The s-automaton M'.

Example 2. Figure 2 shows the s-automaton M' associated with the counter machine M of Fig. 1. The initial state is q_{00}.

A word w accepted by M' either belongs to $\Sigma'^{\star}_{0^k} = \{\sigma_{0^k} \mid \sigma \in \Sigma\}^\star$ or it contains at least one symbol $\sigma_{\boldsymbol{d}}$ or $\sigma^G_{\boldsymbol{d}}$ with $\boldsymbol{d}[l] < 0$ for at least one l. In particular, for any l with $1 \leq l \leq k$, the word w contains at most one symbol $\sigma^G_{\boldsymbol{d}}$ with $l \in G$. In other words, M' can guess only once that a particular counter drops to 0. The next section shows how to construct a suitable system of linear constraints to impose that each guess on a set of counters G is made in the right place, *i.e.* when M (during a one-symbol transition) actually resets all the counters in G.

4 $\mathcal{L}_{\mathsf{DFCM}_\emptyset}$ and RCM

In this section we compare RCM to $\mathcal{L}_{\mathsf{DFCM}_\emptyset}$. We recall that RCM is not contained in $\mathcal{L}_{\mathsf{DFCM}}$ [3, Thm. 9], whereas it is contained in $\mathcal{L}_{\mathsf{NFCM}}$ [3, Thm. 10]. In order to prove that $\mathcal{L}_{\mathsf{DFCM}_\emptyset} \subsetneq \mathsf{RCM}$ it is sufficient to show that for any $L \in \mathcal{L}_{\mathsf{DFCM}_\emptyset}$ there exist a regular language R, a set C of linear constraints and a morphism μ (injective on $R \cap [C]$) such that $L = \langle R, C, \mu \rangle$.

Theorem 1. $\mathcal{L}_{\mathsf{DFCM}_\emptyset} \subsetneq \mathsf{RCM}$.

Proof. Since $\mathcal{L}_{\mathsf{DFCM}_\emptyset} \subseteq \mathcal{L}_{\mathsf{DFCM}}$, by [3, Thm. 9] one has $\mathcal{L}_{\mathsf{DFCM}_\emptyset} \neq \mathsf{RCM}$. So, let $M = (k, Q, \Sigma, \$, \delta, \dot{q}, \{\ddot{q}\})$ be a counter machine in $\mathsf{DFCM}_\emptyset(\mathsf{k}, 0, 1)$, and let $M' = (Q', \Sigma', \delta', \dot{q}_{0^k}, F'\})$ be the s-automaton associated with M (see Def. 7). We construct a system C of linear constraints such that $L(M) = \langle L(M'), C, \mu \rangle$, where $\mu : \Sigma'^\star \mapsto \Sigma^\star$ is an injective morphism on $L(M') \cap [C]$ defined by $\mu(\sigma_{\boldsymbol{d}}) = \mu(\sigma^G_{\boldsymbol{d}}) = \sigma$.

Recall that symbols $\sigma_{\boldsymbol{d}}, \sigma^G_{\boldsymbol{d}}$ in Σ' have weight $W(\sigma_{\boldsymbol{d}}) = W(\sigma^G_{\boldsymbol{d}}) = \boldsymbol{d}$. Weights are used to define the system C. Indeed, M' has been defined so that it reads a symbol $\sigma_{\boldsymbol{d}}$ (or $\sigma^G_{\boldsymbol{d}}$) if and only if M adds $\boldsymbol{d}$ to the counters when it reads $\sigma = \mu(\sigma_{\boldsymbol{d}}) = \mu(\sigma^G_{\boldsymbol{d}})$. Hence, the weight of a word w' in $L(M')$ consisting of n symbols, $w' = \sigma'_1\sigma'_2 \cdots \sigma'_n$, is $W(w') = \sum_{j=1}^{n} W(\sigma'_j) = \sum_{\sigma^G_{\boldsymbol{i}} \in \Sigma'} \boldsymbol{i}|w'|_{\sigma^G_{\boldsymbol{i}}} + \sum_{\sigma_{\boldsymbol{i}} \in \Sigma'} \boldsymbol{i}|w'|_{\sigma_{\boldsymbol{i}}}$.

As observed at the end of Sect. 3, a word $w' \in L(M')$ either belongs to $\Sigma'^{\star}_{0^k}$, or, for all l with $1 \le l \le k$, it has at most one occurrence of a symbol $\sigma^G_{\boldsymbol{d}}$ such that $l \in G$. Thus, we consider the system C of linear constraints given by $C_0 \vee (C_{k+1} \bigwedge_{1 \le l \le k} C_l)$, where

$$\begin{aligned}
C_0 : &\quad \sum_{\boldsymbol{i}} |w|_{\sigma^G_{\boldsymbol{i}}} = 0 \wedge \sum_{\boldsymbol{i} \neq 0^k} |w|_{\sigma_{\boldsymbol{i}}} = 0, \\
C_l : &\quad \sum_{\boldsymbol{i}} \sum_{l \in G} |w|_{\sigma^G_{\boldsymbol{i}}} \le 1, \\
C_{k+1} : &\quad \bigwedge_{1 \le l \le k} \left(\sum_{\sigma^G_{\boldsymbol{i}} \in \Sigma'} \boldsymbol{i}[l]|w|_{\sigma^G_{\boldsymbol{i}}} + \sum_{\sigma_{\boldsymbol{i}} \in \Sigma'} \boldsymbol{i}[l]|w|_{\sigma_{\boldsymbol{i}}} = 0 \right).
\end{aligned}$$

The constraint C_0 is satisfied only by words in $\Sigma'^{\star}_{0^k}$, whereas C_l is satisfied only by words with at most one guess-symbol associated with the l-th counter. Lastly, C_{k+1} is satisfied by words of weight 0^k.

Now, we prove that μ is injective on $L(M') \cap [C]$. Suppose that there exist $x_1, x_2 \in L(M') \cap [C]$ such that $x_1 = x\tau_1 z_1$ and $x_2 = x\tau_2 z_2$, with $x, z_1, z_2 \in \Sigma'^{\star}$, $\tau_1, \tau_2 \in \Sigma'$, $\tau_1 \neq \tau_2$, $\mu(\tau_1) = \mu(\tau_2) = \sigma$ and $z = \mu(z_1) = \mu(z_2)$. Let $y = \mu(x)$. Since M is deterministic, there is only one pair $(p, \boldsymbol{c})$, with $p \in Q$ and $\boldsymbol{c} \in \mathbb{N}^k$, such that $(\dot{q}, y\sigma z\$, 0^k) \overset{|y|}{\Rightarrow} (p, \sigma z\$, \boldsymbol{c})$. Furthermore, also the transition $s = (p, \sigma z\$, \boldsymbol{c}) \Rightarrow (\hat{p}, z\$, \boldsymbol{c} + \boldsymbol{i})$ is uniquely determined, as well as $\boldsymbol{i} \in \mathbb{Z}^k$. By construction, $\tau_1 \neq \tau_2$ implies $\tau_1 = \sigma_{\boldsymbol{i}}$ and $\tau_2 = \sigma^G_{\boldsymbol{i}}$, for a suitable set G of indices such that $\boldsymbol{i}[l] < 0$ for all $l \in G$. Thus, the automaton M' reads x and enters a suitable state $p_{\boldsymbol{\alpha}}$. Then the two computations have different evolutions:

1. M' reads $\tau_1 = \sigma_{\boldsymbol{i}}$ and enters $\hat{p}_{\boldsymbol{\beta}}$, with $\boldsymbol{\beta}[l] = 2$ for all l such that $\boldsymbol{i}[l] < 0$;
2. M' reads $\tau_2 = \sigma^G_{\boldsymbol{i}}$ and enters $\hat{p}_{\boldsymbol{\gamma}}$, with $\boldsymbol{\gamma}[l] = 3$ for all $l \in G$ (hence $\boldsymbol{\gamma} \neq \boldsymbol{\beta}$).

Consider Case (2). Once in $\hat{p}_{\boldsymbol{\gamma}}$, if M' has a transition on a symbol of weight $\boldsymbol{j}$ then the condition $\boldsymbol{j}[l] = 0$ necessarily holds for all $l \in G$. This implies $W(z_2)[l] = 0$ for all $l \in G$. If $W(x)[l] \neq -\boldsymbol{i}[l]$ for an integer $l \in G$, then C_{k+1} is not satisfied and $x_2 \notin L(M') \cap [C]$. So, one has $W(x)[l] = -\boldsymbol{i}[l]$ for all $l \in G$, that is, $W(x\tau_2)[l] = 0$.

Now, consider Case (1). Once in $\hat{p}_{\boldsymbol{\beta}}$, in all the following transitions (*i.e.* on reading z_1) M' can read only symbols τ with $W(\tau)[l] \le 0$ for all $l \in G$. Furthermore, in order to enter a final state $\ddot{q}_{\boldsymbol{\alpha}}$ (with $\boldsymbol{\alpha}[l] = 3$ for all $l \in G$), M' has to read a guess symbol τ (in z_1) such that $W(\tau)[l] < 0$ for at least one l in G. This implies $W(z_1)[l] < 0$. Lastly, by recalling that $W(x\tau_2)[l] = W(x\tau_1)[l] = 0$, it follows that $W(x_1)[l] < 0$, hence x_1 does not satisfy C and $x_1 \notin L(M') \cap [C]$.

Next, we proceed to prove $L(M) = \mu(L(M') \cap [C])$.

($L(M) \subseteq \mu(L(M') \cap [C])$) Let $w \in L(M)$. If w is accepted without incrementing any counter then consider the word $\tilde{w}$ obtained from w by replacing a symbol σ with σ_{0^k}, that is, $\tilde{w} \in \Sigma'^{\star}_{0^k}$ and $\mu(\tilde{w}) = w$. By Def. 7, it is immediate that the automaton M' on input $\tilde{w}$ enters the final state $\ddot{q}_{0^k}$, hence $\tilde{w} \in L(M')$. Moreover, one has $\tilde{w} \in [C_0]$, hence $\tilde{w} \in [C]$.

Otherwise, let $G \subseteq \{1, 2 \ldots, k\}$ be the set of counters that are increased (at least once) by M during the computation which accepts $w = \sigma_1 \cdots \sigma_n$ (recall that M accepts a word with all counters equal to 0). For each $l \in G$, let $i_l = e$ if the one-symbol transition consuming σ_e changes the global state of the l-th counter to 3, that is,

$$(\dot{q}, w\$, 0^k) \overset{e-1}{\Rightarrow} (p, \sigma_e \cdots \sigma_n\$, \boldsymbol{c}) \Rightarrow (\hat{p}, \sigma_{e+1} \cdots \sigma_n\$, \boldsymbol{c} + \boldsymbol{i})$$

with $\boldsymbol{c}[l] + \boldsymbol{i}[l] = 0$. Possibly, one has $i_l = i_m$ for $l \neq m$. This means that, for a suitable r with $1 \leq r \leq |G|$, the set G is uniquely partitioned into r disjoint sets $G_1, \ldots, G_r$ by the equivalence relation $l \equiv m$ if and only if $i_l = i_m$. So, each set G_j uniquely identifies a symbol $\sigma_{r(j)}$ such that the $r(j)$-th one-symbol transition (the one that reads $\sigma_{r(j)}$) sets all the counters in G_j to zero, that is, $(p_{r(j)}, \sigma_{r(j)} \cdots \sigma_n\$, \boldsymbol{c}_{r(j)}) \Rightarrow (p_{r(j)+1}, \sigma_{r(j)+1} \cdots \sigma_n\$, \boldsymbol{c}_{r(j)} + \boldsymbol{i}_{r(j)})$ with $\boldsymbol{c}_{r(j)}[d] + \boldsymbol{i}_{r(j)}[d] = 0$ for all $d \in G_j$.

Now, consider the word $w' \in \Sigma'^{\star}$ that is obtained by replacing in w the symbols $\sigma_{r(j)}$ with $\sigma^{G_j}_{\boldsymbol{i}_{r(j)}}$, and by replacing all the remaining symbols σ_e with $\sigma_{\boldsymbol{i}_e}$ ($\boldsymbol{i}_e$ is the effect on the counters when M reads σ_e). By recalling the relation between one-symbol transitions and evolutions (see Sect. 3), it directly follows from Def. 7 that $w' \in L(M')$. Moreover, one has also $w' \in [C]$. Indeed, $w' \notin [C_0]$, whereas for all l one has $w' \in [C_l]$ (only one guess for each counter l) and $w' \in [C_{k+1}]$ (one has $W(w') = 0$ since w' is constructed so that M' guesses the value of each counter in the right place, *i.e.* when M actually resets the counter).

($\mu(L(M') \cap [C]) \subseteq L(M)$) Let $w' \in L(M') \cap [C]$ and $w = \mu(w')$. If $w' \in \Sigma'^{\star}_{0^k}$ then, by Def. 7, in M there exists a suitable $p \in Q$ such that $(\dot{q}, w\$, 0^k) \overset{|w|}{\Rightarrow} (p, \$, 0^k) \overset{*}{\rightarrow} (\ddot{q}, \$, 0^k)$, that is, $w \in L(M)$. Otherwise, w' can be uniquely written as $w' = x_1 \sigma^{G_1}_{\boldsymbol{i}_1} x_2 \sigma^{G_2}_{\boldsymbol{i}_2} \cdots x_r \sigma^{G_r}_{\boldsymbol{i}_r} x_{r+1}$, with $x_j \in \{\sigma_{\boldsymbol{i}} | \sigma \in \Sigma, \boldsymbol{i} \in [-3k|Q|, 3k|Q|]^k\}^{\star}$, $\bigcup_{j=1}^{r} G_j \subseteq \{1, \ldots, k\}$ and $G_p \cap G_q = \emptyset$ for $p \neq q$. Notice that $w' \notin [C_0]$, hence $w' \in \bigcap_{1 \leq l \leq k+1} [C_l]$. This means that for each j, with $1 \leq j \leq r$, and for all $f \in G_j$ one has $W(x_1 \sigma^{G_1}_{\boldsymbol{i}_1} \cdots x_j \sigma^{G_j}_{\boldsymbol{i}_j})[f] = 0$ and $W(x_1 \sigma^{G_1}_{\boldsymbol{i}_1} \cdots x_j)[f] > 0$, that is, $\boldsymbol{i}_j[f] = -W(x_1 \sigma^{G_1}_{\boldsymbol{i}_1} \cdots x_j)[f]$. Furthermore, for all states $p_{\boldsymbol{\alpha}}$ entered by M' on reading $x_{j+1} \sigma^{G_{j+1}}_{\boldsymbol{i}_{j+1}} \cdots x_r \sigma^{G_r}_{\boldsymbol{i}_r} x_{r+1}$ one has $\boldsymbol{\alpha}[l] = 3$ for all $l \in G_j$. In other words, if $\sigma_{\boldsymbol{i}}$ (or $\sigma^{G}_{\boldsymbol{i}}$) is a symbol occurring in w' to the right of $\sigma^{G_j}_{\boldsymbol{i}_j}$, then the condition $\boldsymbol{i}[l] = 0$ necessarily holds for all $l \in G_j$. Remark that M' enters a final state $\ddot{q}_{\boldsymbol{\beta}}$ for a suitable $\boldsymbol{\beta} \in \{0, 3\}^k$ with $\boldsymbol{\beta}[l] = 3$ for $l \in \bigcup_{j=1}^{r} G_j$.

So, it is sufficient to prove that for any h with $1 \leq h \leq |w'|$, if in M' one has $(\dot{q}_{0^k}, w') \overset{h}{\Rightarrow} (p_{\boldsymbol{\beta}}, w'_{>h})$ then in M there exists a sequence of h transitions

$(\dot{q}, w\$, 0^k) \overset{0^k}{\underset{\boldsymbol{\beta}}{\Rightarrow}}{}^h (p, w_{>h}\$, \boldsymbol{c})$ with $\boldsymbol{c} = W(w'_{\leq h})$. Indeed, $w' \in L(M') \cap [C]$ implies that M' accepts w' by entering a final state $\ddot{q}_{\boldsymbol{\beta}}$, with $\boldsymbol{\beta} \in \{0,3\}^k$ and $W(w') = 0^k$: then M enters $\ddot{q}$ with all counters equal to zero (*i.e.* $\boldsymbol{c} = 0^k$), hence $w \in L(M)$.

We reason by induction on h.

($h = 1$). The first symbol of w' is a symbol $\sigma_{\boldsymbol{i}}$, for suitable $\sigma \in \Sigma$ and $\boldsymbol{i} \in \mathbb{N}^k$. So, by Def. 7, if $\delta'(\dot{q}_{0^k}, \sigma_{\boldsymbol{i}}) = q_{\boldsymbol{\beta}}$ then in M one has that $\mathrm{Ev}(0^k, \dot{q}, \sigma)$ contains the evolution $\{(p_i, \boldsymbol{\alpha}_i, \boldsymbol{d}_i)\}_{i=1,\dots,r}$, with $\boldsymbol{i} = \sum_i \boldsymbol{d}_i$, $p_r = q$ and $\boldsymbol{\alpha}_r = \boldsymbol{\beta}$, that is, $(\dot{q}, \sigma w_{>1}\$, 0^k) \overset{0^k}{\underset{\boldsymbol{\beta}}{\Rightarrow}} (q, w_{>1}\$, \boldsymbol{i})$, with $W(w_{\leq 1}) = W(\sigma_{\boldsymbol{i}}) = \boldsymbol{i}$.

($h > 1$). By induction hypothesis, one has $(\dot{q}_{0^k}, w') \overset{h-1}{\Rightarrow} (p_{\boldsymbol{\beta}}, w'_{\geq h})$ (in M') and $(\dot{q}, w\$, 0^k) \overset{0^k}{\underset{\boldsymbol{\beta}}{\Rightarrow}}{}^{h-1} (p, w_{\geq h}\$, \boldsymbol{c})$ (in M), with $\boldsymbol{c} = W(w'_{<h})$ and $\kappa(\boldsymbol{c}) = \nu(\boldsymbol{\beta})$.

Suppose that the h-th symbol of w' is $\sigma_{\boldsymbol{i}}$. By Def. 7, if $\delta'(p_{\boldsymbol{\beta}}, \sigma_{\boldsymbol{i}}) = q_{\boldsymbol{\gamma}}$ then in M one has that $\mathrm{Ev}(\boldsymbol{\beta}, p, \sigma)$ contains the evolution $\{(p_i, \boldsymbol{\alpha}_i, \boldsymbol{d}_i)\}_{i=1,\dots,r}$, with $\boldsymbol{i} = \sum_i \boldsymbol{d}_i$, $p_r = q$, $\boldsymbol{\alpha}_r = \boldsymbol{\gamma}$ and $\boldsymbol{\gamma}[l] < 3$ if $\boldsymbol{\beta}[l] < 3$. Let $\boldsymbol{c}' = \boldsymbol{c} + \boldsymbol{i}$ and consider an index l such that $\boldsymbol{i}[l] < 0$ (hence $\boldsymbol{\gamma}[l] = 2$). Since $W(w'_{\leq h}) = \boldsymbol{c} + \boldsymbol{i}$, if $W(w'_{\leq h})[l] \leq 0$ then $W(w')[l] < 0$ holds and $w' \notin L(M') \cap [C]$. Indeed, once in $q_{\boldsymbol{\gamma}}$ (and in all subsequent states) the automaton M' has not a transition on a symbol of weight $\boldsymbol{e}$ with $\boldsymbol{e}[l] > 0$, whereas it eventually has a transition on a guess-symbol $\sigma^G_{\boldsymbol{d}}$ with $\boldsymbol{d}[l] < 0$ (in order to enter a final state $q_{\boldsymbol{\eta}}$ with $\boldsymbol{\eta}[l] = 3$). Thus, one has $\boldsymbol{c}'[j] \geq 0$ for all j, and in M there exists the sequence of h transitions $(\dot{q}, w\$, 0^k) \overset{0^k}{\underset{\boldsymbol{\beta}}{\Rightarrow}}{}^{h-1} (p, \sigma w_{>h}\$, \boldsymbol{c}) \overset{\boldsymbol{\beta}}{\underset{\boldsymbol{\gamma}}{\Rightarrow}} (q, w_{>h}\$, \boldsymbol{c}')$, with $\boldsymbol{c}' = W(w_{\leq h})$. Furthermore, one has $\kappa(\boldsymbol{c}') = \nu(\boldsymbol{\gamma})$. Indeed, $\boldsymbol{\gamma}[l] = 0$ implies $\boldsymbol{\beta}[l] = 0$ and $\boldsymbol{i}[l] = 0$, hence $\boldsymbol{c}'[l] = 0$ and $\kappa(\boldsymbol{c}')[l] = \nu(\boldsymbol{\gamma})[l] = 0$. Otherwise, if $\boldsymbol{\gamma}[l] = 1$ then either $\boldsymbol{i}[l] > 0$ (hence $\boldsymbol{c}'[l] > 0$ and $\kappa(\boldsymbol{c}')[l] = \nu(\boldsymbol{\gamma})[l] = 1$), or $\boldsymbol{i}[l] = 0$, $\boldsymbol{\beta}[l] = 1$, $\boldsymbol{c}'[l] = \boldsymbol{c}[l] > 0$ and $\kappa(\boldsymbol{c}')[l] = \nu(\boldsymbol{\gamma})[l] = 1$. Recall that $\boldsymbol{\gamma}[l] = 3$ only if $\boldsymbol{\beta}[l] = 3$ (hence $\boldsymbol{i}[l] = 0$, $\boldsymbol{c}'[l] = \boldsymbol{c}[l] = 0$ and $\kappa(\boldsymbol{c}')[l] = \nu(\boldsymbol{\gamma})[l] = 0$). Lastly, consider the case $\boldsymbol{\gamma}[l] = 2$. If $\boldsymbol{i}[l] = 0$ then one necessarily has $\boldsymbol{\beta}[l] = 2$ and $\boldsymbol{c}'[l] = \boldsymbol{c}[l] > 0$, hence $\kappa(\boldsymbol{c}')[l] = \nu(\boldsymbol{\gamma})[l] = 1$. Otherwise, if $\boldsymbol{i}[l] < 0$ then one has $W(w_{\leq h})[l] > 0$ (as shown above) and $\boldsymbol{c}[l] > \boldsymbol{c}'[l] > 0$, hence $\kappa(\boldsymbol{c}')[l] = \nu(\boldsymbol{\gamma})[l] = 1$.

We proceed similarly if the h-th symbol of w' is a guess symbol $\sigma^G_{\boldsymbol{i}}$. The only difference is that for $l \in G$ one has $W(w'_{<h})[l] = -\boldsymbol{i}[l]$, hence $W(w'_{\leq h})[l] = 0$ and $\boldsymbol{c}[l] + \boldsymbol{i}[l] = \boldsymbol{c}'[l] = 0$. Indeed, if $W(w'_{\leq h})[l] \neq 0$ then $W(w')[l] \neq 0$, since by Def. 7 one has $\boldsymbol{\gamma}[l] = 3$ and for any symbol $\sigma_{\boldsymbol{i}}$ (resp., $\sigma^G_{\boldsymbol{j}}$) in $w'_{>h}$ one has $W(\sigma_{\boldsymbol{i}})[l] = \boldsymbol{i}[l] = 0$ (resp., $W(\sigma^G_{\boldsymbol{j}})[l] = \boldsymbol{j}[l] = 0$). $\square$

As an immediate consequence of the previous theorem one has:

Corollary 1. *Let $L \in \mathcal{L}_{\mathsf{DFCM}_\emptyset}$. Then, the generating function $\phi_L(x)$ is holonomic.*

Corollary 1 implies that a language L is not in $\mathcal{L}_{\mathsf{DFCM}_\emptyset}$ if its generating function is not holonomic. For instance, the language $L = \{a^i b^{i^2}\}$ is neither in $\mathcal{L}_{\mathsf{DFCM}_\emptyset}$ nor in RCM since its generating function $\phi_L(x) = \sum_{n \geq 0} a_n x^n \neq 0$ is not holonomic (here, $a_n = |\{w \in L \mid |w| = n\}|$). Indeed, for any holonomic function $f(x) =$

$\sum_{n\geq 0} b_n x^n$ that is not a polynomial, there exists a constant m such that for any $i \geq 0$ at least one of the coefficients in the sequence $b_i, b_{i+1}, \ldots, b_{i+m}$ is not zero (see [12]). It is immediate that such a property does not hold for $\{a_n\}$.

5 Conclusions and further work

We have shown that $\mathcal{L}_{\mathsf{DFCM}_\emptyset} \subsetneq \mathsf{RCM}$. This is a new result concerning the relationship between RCM and other classes of languages defined by means of reversal bounded counter machines. In particular, we are close to solve the conjecture $\mathcal{L}_{\mathsf{DFCM}} \subsetneq \mathsf{RCM}$ stated in [3], since we think that the technique used to deal with multiple counters should work also in the case of negative cycles.

We stress that proving this conjecture would lead to an important result concerning the holonomicity of the generating functions of languages in $\mathcal{L}_{\mathsf{DFCM}}$. As far as we know, apart [11, Cor.1 and Cor.2], there is not a general result regarding the generating functions of languages accepted by suitable classes of reversal bounded counter machines. This makes the previous conjecture of particular interest.

References

1. A. Bertoni, P. Massazza, and N. Sabadini. Holonomic generating functions and context free languages. *Int. J. Found. Comput. Sci.*, 3(2):181–191, 1992.
2. M.l Cadilhac, A. Finkel, and P. McKenzie. Affine parikh automata. *RAIRO-Theor. Inf. Appl.*, 46(4):511–545, 2012.
3. G. Castiglione and P. Massazza. On a class of languages with holonomic generating functions. *Theor. Comput. Sci.*, 658:74–84, 2017.
4. N. Chomsky and M. P. Schützenberger. The algebraic theory of context-free languages. *Computer Programming and Formal Systems*, pages 118–161, 1963.
5. P. Flajolet. Ambiguity and transcendence. In *ICALP1985, Proc.*, volume 194 of *Lect. Notes Comput. Sc.*, pages 179–188. Springer, 1985.
6. P. Flajolet. Analytic models and ambiguity of context-free languages. *Theor. Comput. Sci.*, 49:283–309, 1987.
7. O.H. Ibarra. Reversal-bounded multicounter machines and their decision problems. *J. ACM*, 25(1):116–133, January 1978.
8. F. Klaedtke and H. Rueß. Parikh automata and monadic second-order logics with linear cardinality constraints. Technical report, Dep. of Computer Science, Univ. of Freiburg, 2002.
9. F. Klaedtke and H. Rueß. Monadic second-order logics with cardinalities. In *ICALP2003, Proc.*, volume 2719 of *Lect. Notes Comput. Sc.*, pages 681–696. Springer, 2003.
10. P. Massazza. Holonomic functions and their relation to linearly constrained languages. *RAIRO-Theor. Inf. Appl.*, 27(2):149–161, 1993.
11. P. Massazza. On the conjecture $\mathcal{L}_{\text{dfcm}(1,0,1)} \subsetneq \text{RCM}$. In *CIAA 2017, Proc.*, volume 10329 of *Lect. Notes Comput. Sc.*, pages 175–187. Springer, 2017.
12. R.P. Stanley. Differentiably finite power series. *Eur. J. Combin.*, 1(2):175 – 188, 1980.
13. D. Zeilberger. A holonomic systems approach to special functions identities. *J. Comput. Appl. Math.*, 32(3):321 – 368, 1990.