

The Power of Commuting with Finite Sets of Words

Michal Kunc *

Department of Mathematics, Masaryk University,
Janáčkovo nám. 2a, 602 00 Brno, Czech Republic
kunc@math.muni.cz, <http://www.math.muni.cz/~kunc/>

March 10, 2006

Abstract

We construct a finite language L such that the largest language commuting with L is not recursively enumerable. This gives a negative answer to the question raised by Conway in 1971 and also strongly disproves Conway's conjecture on context-freeness of maximal solutions of systems of semi-linear inequalities.

1 Introduction

In this paper we address the question whether the largest solution of any language equation of the form $XL = LX$ is regular provided L is a finite or regular language. It is known that in several algebraic structures related to algebras of formal languages, two elements commute if and only if they can be generated by the same element. For instance, two words commute if and only if they are powers of the same word (due to the defect theorem), and such a characterization was proved valid also for polynomials and formal series in non-commuting variables over a field (in particular, for multisets

*Supported by the project MSM0021622409 of the Ministry of Education of the Czech Republic.

of words) by Bergman [3] and Cohn [7], respectively. But in the case of languages the situation is completely different.

Systems of language equations and inequalities were studied mainly in connection with context-free languages since these languages can be described as components of smallest solutions of explicit systems of polynomial equations, i.e. equations with the operations of union and concatenation. Much less attention was devoted to systems of implicit equations. An attempt to initiate development of a unified theory of general language equations has been recently made by Okhotin; in particular, he proved that recursive (recursively enumerable, co-recursively enumerable) languages are exactly languages definable as unique (smallest, largest) solutions of systems of implicit language inequalities using concatenation and all Boolean operations [19] and that each such language can be encoded even in a single explicit equation and precisely defined by an explicit system consisting of two equations [20].

Known results on regularity of maximal solutions of systems of implicit language inequalities are surveyed in [14]. Such issues were first addressed by Conway [8], who observed that inequalities of the form $E \subseteq L$, where E is a regular function of variables and L is a regular language, possess only finitely many maximal solutions, all of them are regular and computable.

Maximal solutions are also simple in the case of inequalities where variables are concatenated only from one side. It is well known that regular languages can be characterized as components of smallest or largest solutions of explicit systems of right-linear equations. Regular solutions of more general systems were studied for example by Leiss [17]. For systems of implicit right-linear inequalities, i.e. inequalities of the form

$$K \cup K_1 X_1 \cup \dots \cup K_n X_n \subseteq L \cup L_1 X_1 \cup \dots \cup L_n X_n,$$

where K, K_1, \dots, K_n and L, L_1, \dots, L_n are constant languages, it is known that their largest solutions are always regular provided all constant languages on right-hand sides are regular [16]. Moreover, if all constant languages occurring in the system are regular, then computability of the largest solution follows from Rabin's results on MSO logic over infinite trees [21]; actually, the computation of the solution is an ExpTime-complete problem [1, 5, 2].

The problem of regularity of the largest language commuting with a given regular language was formulated already by Conway [8, p. 55 and 124] in 1971. There are actually two variants of the problem depending on whether we allow the resulting language to contain the empty word or not: The

largest solution of the equation $XL = LX$ is denoted $\mathcal{C}(L)$ and its largest solution without the empty word is denoted $\mathcal{C}_+(L)$. The languages $\mathcal{C}(L)$ and $\mathcal{C}_+(L)$ are in general different and no direct relation between them (except for the obvious inclusion $\mathcal{C}_+(L) \subseteq \mathcal{C}(L)$) has been found yet. The problem was recently studied in several articles (e.g. [22, 6, 9, 10]), but affirmative answers were given only for regular codes [10] and at most ternary sets of words [9]. Moreover, it even remained an open problem whether the largest language commuting with a given finite set of words is recursive. On the other hand, the complements of languages $\mathcal{C}(L)$ and $\mathcal{C}_+(L)$ are always recursively enumerable provided the language L is recursive; this is a special case of a general result about systems of language equations, which is proved in [19]. A summary of known results concerning commutation of languages and some examples can be found in the recent survey [12].

In this paper we give the most negative possible answer to Conway's problem by showing that there exists a finite language L such that $\mathcal{C}(L)$ is not recursively enumerable. More precisely, we show that the complement of the language computed by an arbitrary Minsky machine can be encoded into a solution of a commutation equation. This contrasts with the fact that the largest solution of the inequality $XK \subseteq LX$ is regular provided the language L is regular, as demonstrated in [13]. On the other hand, negative results analogous to those for commutation equations were proved in [15] for systems consisting of two such inequalities.

We formulate our results for the case of languages $\mathcal{C}_+(L)$ too, and further we show that for a regular language L the difference $\mathcal{C}(L) \setminus \mathcal{C}_+(L)$ also does not have to be recursively enumerable, which answers a question posed in [11]. In addition, our results disprove Conway's conjecture [8, p. 129] stating that every maximal solution of a system of so-called semi-linear inequalities is context-free.

Before dealing with the main result of this paper, we demonstrate the techniques employed in its proof by proving several weaker results. In Section 3 we consider the situation where L is only required to be a star-free language. First, we give an example of a star-free language L such that $\mathcal{C}(L)$ is non-regular, and then we describe how the construction can be improved to show that $\mathcal{C}(L)$ does not even have to be recursively enumerable. Let us mention that this is in accord with the results obtained in [13]: complicated cases arise for star-free languages (or equivalently, languages recognizable by aperiodic monoids), whereas maximal solutions of such equations with constant languages recognizable by finite simple semigroups (in particular, by

finite groups) are always regular. Based on the constructions presented in this section, we additionally prove that it is undecidable whether two star-free languages are conjugated via some language containing the empty word. This result is a step towards dealing with basic problems about conjugacy of languages formulated in [4].

Section 4 is devoted to the case of finite languages; by encoding the example from the beginning of Section 3 into finitely many words, we show that the language $\mathcal{C}(L)$ can be non-regular even for a finite language L . Finally, in Section 5 we combine the techniques of Sections 3 and 4 to obtain a finite set of words L such that $\mathcal{C}(L)$ is not recursively enumerable.

Basic notions employed in our considerations are recalled in the following section. For a more comprehensive introduction to the theory of formal languages the reader is referred to [23].

2 Preliminaries

We denote the sets of positive and non-negative integers by \mathbb{N} and \mathbb{N}_0 , respectively. Throughout the paper we consider a certain finite alphabet A . As usual, we write A^+ for the set of all non-empty finite words over A , and A^* for the set obtained from A^+ by adding the empty word ε . For a letter $a \in A$ and a positive integer $n \in \mathbb{N}$, we denote the set $\{\varepsilon, a, \dots, a^n\}$ by $a^{\leq n}$. A word $u \in A^*$ is called a *factor* of $v \in A^*$ if $v = wu\hat{w}$ for some words $w, \hat{w} \in A^*$; it is called a *prefix* (*suffix*) of v if $v = uw$ ($v = wu$, respectively) for some $w \in A^*$.

Languages over the alphabet A are arbitrary subsets of A^* , and we say that a language $L \subseteq A^*$ is ε -free if $\varepsilon \notin L$. The *reverse* of a language L is defined as

$$\{a_n \cdots a_1 \mid a_1, \dots, a_n \in A, a_1 \cdots a_n \in L\}$$

and denoted $\text{rev}(L)$. The basic operation on languages is concatenation defined by the rule $K \cdot L = \{uv \mid u \in K, v \in L\}$, and we use the standard notation $L^+ = \bigcup_{m \in \mathbb{N}} L^m$ and $L^* = L^+ \cup \{\varepsilon\}$. *Regular* languages are languages definable by finite automata, or equivalently, by rational expressions. The basic tool for proving non-regularity of languages is the well-known pumping lemma (see e.g. [23]). A language $L \subseteq A^*$ is called *star-free* if it can be obtained from finite languages using the operations of union, complementation and concatenation; in particular, for every $B \subseteq A$, the languages B^+ and B^* are star-free.

For every language L over A we denote by $\mathcal{C}(L)$ the largest language over A which commutes with L and by $\mathcal{C}_+(L)$ the largest ε -free language over A which commutes with L . Such languages $\mathcal{C}(L)$ and $\mathcal{C}_+(L)$ certainly exist for every language L since the union of arbitrarily many languages commuting with L commutes with L as well. It is clear that we always have $\mathcal{C}_+(L) \subseteq \mathcal{C}(L)$, $L^* \subseteq \mathcal{C}(L)$ and $L^+ \subseteq \mathcal{C}_+(L)$. Further, the languages $\mathcal{C}(L)$ and $\mathcal{C}_+(L)$ are easily seen to be closed under concatenation and so they form a submonoid and a subsemigroup of the free monoid A^* , respectively. Another interesting property of the languages $\mathcal{C}(L)$ and $\mathcal{C}_+(L)$ is that they remain unchanged when we replace L with its closure under concatenation, i.e. $\mathcal{C}(L) = \mathcal{C}(L^+)$ and $\mathcal{C}_+(L) = \mathcal{C}_+(L^+)$.

Intuitively, we can view the commutation equation $XL = LX$ as a game of two players, the attacker and the defender. A position of the game is an arbitrary word $w \in A^*$. At each step of the game, the attacker adds any word from L to an arbitrary side of w , and the defender has to respond by removing some word belonging to L from the opposite side of the resulting word. The word thus obtained is a new position of the game. The attacker wins the game if the defender has no move available, and the defender wins if he manages to continue playing forever. Then it is easy to observe that the largest solution of the equation $XL = LX$ is exactly the set of all positions of the game where the defender has a winning strategy.

This intuitive view is also reflected by the structure of proofs in this paper, which consist of two parts: Proving that a given word w does not belong to $\mathcal{C}(L)$ amounts to finding a winning strategy for the attacker on w . And conversely, in order to prove that w lies in $\mathcal{C}(L)$, we describe a set of positions containing w such that no matter how the attacker plays in one of these positions, the defender is always able to return to such a position.

3 Star-Free Languages

The aim of this section is to construct a star-free language L such that the largest solution of the equation $XL = LX$ is not recursively enumerable. This is achieved by encoding an arbitrary Minsky machine \mathcal{M} into a star-free language L in such a way that $\mathcal{C}(L) \cap uv^*w = \{uv^nw \mid n \notin \mathcal{L}(\mathcal{M})\}$, where u , v and w are certain words and $\mathcal{L}(\mathcal{M}) \subseteq \mathbb{N}_0$ is the set computed by the machine \mathcal{M} . Because the construction is rather technical, let us first present it in a simplified form which shows that the largest language commuting with

a given star-free language is not necessarily regular.

Example 1. Let us take the alphabet $A = \{a, b, c, e, \hat{e}, f, \hat{f}, g, \hat{g}\}$. We consider auxiliary languages

$$\begin{aligned} M &= efga^+ba^* \cup ga^*ba^*\hat{g}\hat{f} \cup a^*ba^*\hat{g}\hat{f}\hat{e} \cup fga^*ba^*\hat{g}, \\ L_0 &= (A \setminus \{c\})^*b(A \setminus \{c\})^* \setminus (\{efg, fg, g, \varepsilon\} \cdot a^*ba^* \cdot \{\varepsilon, \hat{g}, \hat{g}\hat{f}, \hat{g}\hat{f}\hat{e}\}) \end{aligned}$$

and define a star-free language

$$L = \{c, ef, ga, e, fg, \hat{f}\hat{e}, a\hat{g}, \hat{e}, \hat{g}\hat{f}, fgba\hat{g}\} \cup A^*bA^*bA^* \cup L_0 \cup cM \cup Mc.$$

With the aim of verifying that the language $\mathcal{C}(L)$ is not regular, let us first observe that $fga^m ba^n \hat{g}\hat{f} \notin \mathcal{C}(L)$ for every $m, n \in \mathbb{N}_0$. Indeed, assuming the converse, we obtain

$$c^2 fga^m ba^n \hat{g}\hat{f} \in L^2 \cdot \mathcal{C}(L) = \mathcal{C}(L) \cdot L^2,$$

which is a contradiction because this word has no suffix belonging to L^2 .

Now we are going to show that $efga^m ba^n \notin \mathcal{C}(L)$ for every $m, n \in \mathbb{N}_0$ satisfying $m < n$. We proceed by induction on m . If $m = 0$ then the converse of this fact implies

$$cefgba^n \in L \cdot \mathcal{C}(L) = \mathcal{C}(L) \cdot L,$$

which is impossible as no suffix of the word $cefgba^n$ lies in L . Let $m \geq 1$ and suppose we already know that $efga^{m-1}ba^{n-1} \notin \mathcal{C}(L)$ holds. By way of contradiction, assume $efga^m ba^n \in \mathcal{C}(L)$. Then

$$efga^m ba^n \hat{g}\hat{f} \in \mathcal{C}(L) \cdot L = L \cdot \mathcal{C}(L).$$

Since there are just two prefixes of $efga^m ba^n \hat{g}\hat{f}$ which belong to L , we have either $fga^m ba^n \hat{g}\hat{f} \in \mathcal{C}(L)$ or $ga^m ba^n \hat{g}\hat{f} \in \mathcal{C}(L)$. Due to our initial observation, the former case is false. From the latter one we deduce

$$ga^m ba^n \hat{g}\hat{f}\hat{e} \in \mathcal{C}(L) \cdot L = L \cdot \mathcal{C}(L).$$

This immediately gives $a^{m-1}ba^n \hat{g}\hat{f}\hat{e} \in \mathcal{C}(L)$ and therefore

$$fga^{m-1}ba^n \hat{g}\hat{f}\hat{e} \in L \cdot \mathcal{C}(L) = \mathcal{C}(L) \cdot L.$$

The word $fga^{m-1}ba^n\hat{g}\hat{f}\hat{e}$ has exactly two suffixes from L , and repeating the previous argument we get $fga^{m-1}ba^n\hat{g} \in \mathcal{C}(L)$. Consequently

$$efga^{m-1}ba^n\hat{g} \in L \cdot \mathcal{C}(L) = \mathcal{C}(L) \cdot L,$$

and since we have $n \geq 2$, this finally leads to $efga^{m-1}ba^{n-1} \in \mathcal{C}(L)$, contradicting our assumption.

On the other hand, the language

$$K = L^* \cup \{efga^nba^n, ga^nba^n\hat{g}\hat{f}, a^{n-1}ba^n\hat{g}\hat{f}\hat{e}, fga^{n-1}ba^n\hat{g} \mid n \in \mathbb{N}\}$$

commutes with L because one can easily calculate that both products KL and LK are equal to the language

$$L^+ \cup \{efga^nba^n\hat{g}\hat{f}, ga^nba^n\hat{g}\hat{f}\hat{e}, fga^{n-1}ba^n\hat{g}\hat{f}\hat{e}, fga^{n-1}ba^n\hat{g} \mid n \in \mathbb{N}\};$$

notice that this holds due to the fact $efgba\hat{g} \in L^2$. Therefore the word $efga^nba^n$ belongs to $\mathcal{C}(L)$ for all $n \in \mathbb{N}$.

Altogether, we have demonstrated both $efga^nba^n \in \mathcal{C}(L)$ for $n \in \mathbb{N}$ and $efga^mba^n \notin \mathcal{C}(L)$ for $m < n$, hence the language $\mathcal{C}(L)$ cannot be regular due to the pumping lemma.

Observe that in the previous example we have in fact encoded into the language L two counters (as powers of a) and the operation of simultaneous decrementation of both counters together with testing whether both counters are equal to zero. The following construction of a language L such that $\mathcal{C}(L)$ is not recursively enumerable is based essentially on the same idea.

Theorem 1. *There exists a star-free language L such that*

1. *the largest language commuting with L is not recursively enumerable.*
2. *the difference between the largest language commuting with L and the largest ε -free language commuting with L is not recursively enumerable.*

Proof. Let \mathcal{M} be a Minsky machine (see [18]) which computes a non-recursive set of non-negative integers. The machine consists of two counters and a finite set of states Q , which is a disjoint union

$$Q = T_1 \cup T_2 \cup I_1 \cup I_2 \cup D_1 \cup D_2 \cup \{1\},$$

where 1 is the terminal state. We assume that the initial state 0 of \mathcal{M} belongs to I_1 . A configuration of the machine is a triple (i, m, n) , where $i \in Q$ and $m, n \in \mathbb{N}_0$ are the values stored in the counters. The step performed by the machine in a given state is determined by the instruction associated with this state:

- From the state $i \in T_k$, $k \in \{1, 2\}$, the machine goes to the state $\tau_0(i)$ if counter k is empty and to the state $\tau_1(i)$ otherwise, where $\tau_0(i) \neq i$ and $\tau_1(i) \neq i$ are distinct states.
- When the machine is in the state $i \in I_k$ (or $i \in D_k$), it increments (decrements, respectively) counter k and goes to the state $\tau(i) \neq i$.
- When the machine reaches the state 1, the computation stops.

Initial configurations of \mathcal{M} are configurations of the form $(0, 0, n)$, $n \in \mathbb{N}_0$. The machine computes the set $\mathcal{L}(\mathcal{M}) \subseteq \mathbb{N}_0$ of all numbers n such that the machine stops in the configuration $(1, 0, n)$ starting from some initial configuration. Since we have chosen \mathcal{M} such that $\mathcal{L}(\mathcal{M})$ is not recursive, its complement $\mathbb{N}_0 \setminus \mathcal{L}(\mathcal{M})$ is not recursively enumerable. In addition, let us assume that the initial state 0 of \mathcal{M} cannot be reached from the other states, and if for some $i, j \in Q$ either $\tau(i) = j$ or $\tau_0(i) = j$ or $\tau_1(i) = j$, then neither $\tau(j) = i$ nor $\tau_0(j) = i$ nor $\tau_1(j) = i$, i.e. given the two states involved in one step of a computation, it is uniquely determined which of these steps is the original one and which is the resulting one.

Consider the alphabet

$$A = \{a, \hat{a}, b, c\} \cup \{d_i, \hat{d}_i \mid i \in Q\} \cup \{e_i, f_i, g_i \mid i \in I_1 \cup I_2 \cup D_1 \cup D_2\}.$$

Every configuration (i, m, n) of the machine will be represented by the word $a^{m+1}b\hat{a}^{n+1}\hat{d}_i^2$. If a configuration is reachable from some initial configuration, then the corresponding word should not belong to $\mathcal{C}(L)$. Since in our encoding using commutation of languages we have no means of directing a computation, the same will hold also for all configurations from which some configuration reachable from an initial configuration is eventually obtained.

In order to construct the language L , we introduce several auxiliary languages first. For each state $i \in Q$, we consider a language M_i which can be used to move occurrences of the letters d_i and \hat{d}_i from one side of a word to the other and in this way enables us to link steps of a computation modifying

different counters: let $M'_0 = a^+ab\hat{a}^+\hat{d}_0^2$ and for $i \in Q \setminus \{0\}$ let

$$M'_i = d_i^2a^+b\hat{a}^+ \cup d_ia^+b\hat{a}^+\hat{d}_i \cup a^+b\hat{a}^+\hat{d}_i^2;$$

then define $M_i = cM'_i \cup M'_ic$ for every $i \in Q$.

The following languages N_i describe words appearing during manipulations corresponding to performing the instruction associated with the state i ; in the case of $i \in I_1 \cup I_2 \cup D_1 \cup D_2$, there are also four short words whose addition and removal transforms one of these words into another.

For $i \in T_1 \cup T_2$ let

$$N'_i = d_ia\hat{b}\hat{a}^+\hat{d}_{\tau_0(i)} \cup d_ia^+ab\hat{a}^+\hat{d}_{\tau_1(i)}$$

and define

$$N_i = cN'_i \cup N'_ic \cup A^*d_ia^+ab\hat{a}^+\hat{d}_{\tau_0(i)}A^* \cup A^*d_ia\hat{b}\hat{a}^+\hat{d}_{\tau_1(i)}A^*.$$

For $i \in I_1 \cup I_2$ let

$$N'_i = g_ia^+b\hat{a}^+\hat{d}_i \cup e_if_ig_ia^+b\hat{a}^+ \cup f_ig_ia^+b\hat{a}^+\hat{d}_{\tau(i)}$$

and define

$$N_i = cN'_i \cup N'_ic \cup \{e_if_i, g_ia, e_i, f_ig_i\} \\ \cup f_ig_ia^+b\hat{a}^+\hat{d}_i^2 \cup g_ia^+b\hat{a}^+\hat{d}_{\tau(i)}^2 \cup b\hat{a}^+\hat{d}_i^2.$$

And dually, for $i \in D_1 \cup D_2$ let

$$N'_i = g_ia^+b\hat{a}^+\hat{d}_{\tau(i)} \cup e_if_ig_ia^+b\hat{a}^+ \cup f_ig_ia^+b\hat{a}^+\hat{d}_i$$

and define

$$N_i = cN'_i \cup N'_ic \cup \{e_if_i, g_ia, e_i, f_ig_i\} \\ \cup f_ig_ia^+b\hat{a}^+\hat{d}_{\tau(i)}^2 \cup g_ia^+b\hat{a}^+\hat{d}_i^2 \cup b\hat{a}^+\hat{d}_{\tau(i)}^2.$$

The next language describes possible consecutive states:

$$L' = \bigcup \{ \{d_i^2, d_i, \varepsilon\} \cdot a^*b\hat{a}^* \cdot \{\varepsilon, \hat{d}_i, \hat{d}_i^2\} \mid i \in Q, i \neq 0 \} \\ \cup \bigcup \{ \{d_i^2, d_i, \varepsilon\} \cdot a^*b\hat{a}^* \cdot \{\varepsilon, \hat{d}_{\tau_1(i)}, \hat{d}_{\tau_1(i)}^2, \hat{d}_{\tau_0(i)}, \hat{d}_{\tau_0(i)}^2\} \mid i \in T_1 \} \\ \cup \bigcup \{ \{d_{\tau_0(i)}^2, d_{\tau_0(i)}, d_{\tau_1(i)}^2, d_{\tau_1(i)}, \varepsilon\} \cdot a^*b\hat{a}^* \cdot \{\varepsilon, \hat{d}_i, \hat{d}_i^2\} \mid i \in T_2 \}$$

$$\begin{aligned} & \cup \bigcup \{ \{ e_i f_i g_i, f_i g_i, g_i, \varepsilon \} \cdot a^* \hat{b} a^* \cdot \{ \varepsilon, \hat{d}_{\tau(i)}, \hat{d}_{\tau(i)}^2, \hat{d}_i, \hat{d}_i^2 \} \mid i \in I_1 \cup D_1 \} \\ & \cup \bigcup \{ \{ d_i^2, d_i, d_{\tau(i)}^2, d_{\tau(i)}, \varepsilon \} \cdot a^* \hat{b} a^* \cdot \{ \varepsilon, g_i, g_i f_i, g_i f_i e_i \} \mid i \in I_2 \cup D_2 \}. \end{aligned}$$

Finally, we define a star-free language

$$\begin{aligned} L = & \{ d_i, \hat{d}_i \mid i \in Q \} \cup \{ c \} \cup A^* b A^* b A^* \\ & \cup (A \setminus \{ c \})^* b (A \setminus \{ c \})^* \setminus L' \\ & \cup \bigcup \{ M_i \mid i \in Q \} \\ & \cup \bigcup \{ N_i \mid i \in T_1 \cup I_1 \cup D_1 \} \\ & \cup \bigcup \{ \psi(\text{rev}(N_i)) \mid i \in T_2 \cup I_2 \cup D_2 \}, \end{aligned}$$

where $\psi: A^* \rightarrow A^*$ is the homomorphism which interchanges letters a and d_i , for $i \in Q$, with letters \hat{a} and \hat{d}_i , respectively.

Let us first briefly sketch out how computations of the Minsky machine are simulated by manipulations of words from A^+ . If some word u obtained during our manipulations is multiplied from some side (say from the left) by a word $v \in L$, one usually gets a word from L with just few exceptions corresponding to correct computations of the machine. Such exceptional products vu then usually have only one suffix v' belonging to L whose removal does produce a word u' which can potentially belong to $\mathcal{C}(L)$. In this way we achieve that u lies in $\mathcal{C}(L)$ if and only if u' lies in $\mathcal{C}(L)$. Therefore every computation of the machine \mathcal{M} preserves the properties that the word corresponding to a configuration belongs, or does not belong, to $\mathcal{C}(L)$.

For instance, starting from a word $u = a^{m+1} b \hat{a}^{n+1} \hat{d}_i^2$ corresponding to the state $i \in I_1$, we can multiply this word by $g_i a \in L$ from the left, and then by removing the word $\hat{d}_i \in L$ from the right we get the word $u' = g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_i$, which is in $\mathcal{C}(L)$ if and only if u is in $\mathcal{C}(L)$. Now the first counter is already incremented in the word u' and the occurrence of the letter g_i on the left ensures that the following manipulations of u' either return us back to u (if we multiply by \hat{d}_i from the right) or continue with simulating the step of \mathcal{M} performed in the state i by adding the word $e_i f_i$ to the left, removing the remaining occurrence of \hat{d}_i on the right, and then multiplying by $\hat{d}_{\tau(i)}$ from the right, eventually reaching the word $a^{m+2} b \hat{a}^{n+1} \hat{d}_{\tau(i)}^2$.

The language L' describes which words are allowed to occur during these manipulations. Each of these words consists of one occurrence of b in the middle surrounded by several occurrences of a and \hat{a} and a block of letters

corresponding to some state of \mathcal{M} on each side; notice that on each side of a word all letters have the same indices. Those words which do not correspond to a correct computation of \mathcal{M} , but can be obtained from such a word by concatenating with a word from L , are included in L .

On the other hand, concatenating with the word $c \in L$ can be used to show that the only eligible results of removing a suffix or a prefix belonging to L are words from the languages M'_i and N'_i . In particular, the letter c serves for proving that all words corresponding to initial configurations do not belong to $\mathcal{C}(L)$, which is consequently true also for all configurations reachable from initial configurations; this is achieved by not including the words from the language $ab\hat{a}^+d_0^2$ into M'_0 .

Restating the previously described ideas formally, in order to show that both sets $\mathcal{C}(L)$ and $\mathcal{C}(L) \setminus \mathcal{C}_+(L)$ are not recursively enumerable, it is enough to verify the following claim:

Claim 1. For every non-negative integer $n \in \mathbb{N}_0$:

$$n \notin \mathcal{L}(\mathcal{M}) \iff ab\hat{a}^{n+1}d_1^2 \in \mathcal{C}(L) \iff ab\hat{a}^{n+1}d_1^2 \in \mathcal{C}(L) \setminus \mathcal{C}_+(L).$$

First we deal with the second equivalence. Notice that if some word of the form $ab\hat{a}^{n+1}d_1^2$ belongs to $\mathcal{C}_+(L)$, then

$$ab\hat{a}^{n+1}d_1^2c \in \mathcal{C}_+(L) \cdot L = L \cdot \mathcal{C}_+(L),$$

which is impossible since $ab\hat{a}^{n+1}d_1^2c \notin LA^+$. Therefore the second condition of the claim is equivalent to the third one.

It remains to prove the equivalence of the first two conditions of Claim 1. Let us first deal with the converse implication. We are going to prove $a^{m+1}b\hat{a}^{n+1}d_i^2 \notin \mathcal{C}(L)$ for every configuration (i, m, n) reachable from some initial configuration; then if we take $i = 1$, $m = 0$ and $n \in \mathcal{L}(\mathcal{M})$, we obtain the desired fact $ab\hat{a}^{n+1}d_1^2 \notin \mathcal{C}(L)$. The proof proceeds by induction with respect to the length of a run of \mathcal{M} reaching (i, m, n) from an initial configuration.

First, let $(0, 0, n)$ be an arbitrary initial configuration. Assuming that $u = ab\hat{a}^{n+1}d_0^2 \in \mathcal{C}(L)$, we easily obtain a contradiction by considering the word $uc \in \mathcal{C}(L) \cdot L$ since uc has no prefix belonging to L and therefore $uc \notin L \cdot \mathcal{C}(L)$.

Before we proceed to consider the inductive step, let us prove that for all $i \in Q \setminus \{0\}$, $m, n \in \mathbb{N}_0$:

$$a^{m+1}b\hat{a}^{n+1}d_i^2 \in \mathcal{C}(L) \iff d_i^2a^{m+1}b\hat{a}^{n+1} \in \mathcal{C}(L). \quad (1)$$

Starting from $a^{m+1}b\hat{a}^{n+1}\hat{d}_i^2 \in \mathcal{C}(L)$, we have

$$d_i^2 a^{m+1} b \hat{a}^{n+1} \hat{d}_i^2 \in L^2 \cdot \mathcal{C}(L) = \mathcal{C}(L) \cdot L^2.$$

As this word has only one suffix belonging to L^2 , namely \hat{d}_i^2 , we conclude $d_i^2 a^{m+1} b \hat{a}^{n+1} \in \mathcal{C}(L)$. The converse implication holds by the same argument.

Let us first consider the step of \mathcal{M} from a configuration $(i, 0, n)$, where $i \in T_1$. We prove that $ab\hat{a}^{n+1}\hat{d}_i^2 \notin \mathcal{C}(L)$ implies $ab\hat{a}^{n+1}\hat{d}_{\tau_0(i)}^2 \notin \mathcal{C}(L)$ by contradiction. If $ab\hat{a}^{n+1}\hat{d}_{\tau_0(i)}^2 \in \mathcal{C}(L)$ then

$$d_i^2 ab\hat{a}^{n+1}\hat{d}_{\tau_0(i)}^2 \in L^2 \cdot \mathcal{C}(L) = \mathcal{C}(L) \cdot L^2.$$

Therefore $d_i^2 ab\hat{a}^{n+1} \in \mathcal{C}(L)$ because $\hat{d}_{\tau_0(i)}^2$ is the only suffix of $d_i^2 ab\hat{a}^{n+1}\hat{d}_{\tau_0(i)}^2$ belonging to L^2 . Hence $ab\hat{a}^{n+1}\hat{d}_i^2 \in \mathcal{C}(L)$ by (1) since $i \neq 0$.

Analogously, one can deal with every configuration (i, m, n) , where $i \in T_1$ and $m \geq 1$.

For a configuration (i, m, n) , where $i \in T_2$ and $n \geq 1$, from the fact $a^{m+1}b\hat{a}^{n+1}\hat{d}_{\tau_1(i)}^2 \in \mathcal{C}(L)$ we deduce $d_{\tau_1(i)}^2 a^{m+1} b \hat{a}^{n+1} \in \mathcal{C}(L)$ by (1). Then

$$d_{\tau_1(i)}^2 a^{m+1} b \hat{a}^{n+1} \hat{d}_i^2 \in \mathcal{C}(L) \cdot L^2 = L^2 \cdot \mathcal{C}(L)$$

and consequently $a^{m+1}b\hat{a}^{n+1}\hat{d}_i^2 \in \mathcal{C}(L)$ as the word $d_{\tau_1(i)}^2$ is the only prefix of $d_{\tau_1(i)}^2 a^{m+1} b \hat{a}^{n+1} \hat{d}_i^2$ from L^2 .

The case of a configuration $(i, m, 0)$, where $i \in T_2$, is again similar.

Now let (i, m, n) be an arbitrary configuration of \mathcal{M} such that $i \in I_1$. Assume $a^{m+2}b\hat{a}^{n+1}\hat{d}_{\tau(i)}^2 \in \mathcal{C}(L)$, which corresponds to the next configuration $(\tau(i), m+1, n)$, and let us prove $a^{m+1}b\hat{a}^{n+1}\hat{d}_i^2 \in \mathcal{C}(L)$. As $f_i g_i \in N_i$, we can see that

$$f_i g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_{\tau(i)}^2 \in L \cdot \mathcal{C}(L) = \mathcal{C}(L) \cdot L.$$

There can be at most three suffixes of $f_i g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_{\tau(i)}^2$ which belong to L , namely $g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_{\tau(i)}^2$, $b \hat{a}^{n+1} \hat{d}_{\tau(i)}^2$ and $\hat{d}_{\tau(i)}$. Therefore either $f_i \in \mathcal{C}(L)$ or $f_i g_i a^{m+2} \in \mathcal{C}(L)$ or $f_i g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_{\tau(i)} \in \mathcal{C}(L)$. In the first case, we get $c f_i \in L \cdot \mathcal{C}(L) = \mathcal{C}(L) \cdot L$, which is impossible. In the second case, we deduce $c f_i g_i a^{m+2} \in L \cdot \mathcal{C}(L) = \mathcal{C}(L) \cdot L$, contradicting the fact that no suffix of this word lies in L . Thus $f_i g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_{\tau(i)} \in \mathcal{C}(L)$ and so

$$e_i f_i g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_{\tau(i)} \in L \cdot \mathcal{C}(L) = \mathcal{C}(L) \cdot L.$$

Now $\hat{d}_{\tau(i)}$ is the only suffix of this word belonging to L and we immediately obtain $e_i f_i g_i a^{m+2} b \hat{a}^{n+1} \in \mathcal{C}(L)$. Let us remark that the previous arguments cannot be performed as one step since $e_i f_i \cdot g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_{\tau(i)}^2 \in L^2$. Continuing with the proof by multiplying the current word by \hat{d}_i , we obtain

$$e_i f_i g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_i \in \mathcal{C}(L) \cdot L = L \cdot \mathcal{C}(L)$$

and therefore

$$f_i g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_i \in \mathcal{C}(L) \quad \text{or} \quad g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_i \in \mathcal{C}(L).$$

In the former case

$$f_i g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_i^2 \in \mathcal{C}(L) \cdot L^2 = L^2 \cdot \mathcal{C}(L),$$

which is impossible because this word has no prefix belonging to L^2 . Hence the latter situation arises. Then

$$g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_i^2 \in \mathcal{C}(L) \cdot L = L \cdot \mathcal{C}(L)$$

and by examining prefixes of this word we easily conclude that $a^{m+1} b \hat{a}^{n+1} \hat{d}_i^2$ belongs to $\mathcal{C}(L)$.

In order to deal with a state $i \in D_1$, consider a configuration $(i, m+1, n)$, where $m, n \in \mathbb{N}_0$. This time we have to prove $a^{m+2} b \hat{a}^{n+1} \hat{d}_i^2 \in \mathcal{C}(L)$ from the assumption $a^{m+1} b \hat{a}^{n+1} \hat{d}_{\tau(i)}^2 \in \mathcal{C}(L)$ corresponding to the configuration $(\tau(i), m, n)$. Using the word $g_i a \in N_i$ we obtain

$$g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_{\tau(i)}^2 \in L \cdot \mathcal{C}(L) = \mathcal{C}(L) \cdot L,$$

which means that either $g_i a^{m+2} \in \mathcal{C}(L)$ or $g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_{\tau(i)} \in \mathcal{C}(L)$. The former leads directly to a contradiction since the word $c g_i a^{m+2} \in L \cdot \mathcal{C}(L) = \mathcal{C}(L) \cdot L$ has no suffix belonging to L . In the latter situation we multiply by the word $e_i f_i \in N_i$, which produces

$$e_i f_i g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_{\tau(i)} \in L \cdot \mathcal{C}(L) = \mathcal{C}(L) \cdot L.$$

Then it is easy to verify $e_i f_i g_i a^{m+2} b \hat{a}^{n+1} \in \mathcal{C}(L)$. Therefore

$$e_i f_i g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_i \in \mathcal{C}(L) \cdot L = L \cdot \mathcal{C}(L),$$

and as exactly two prefixes of this word lie in L , we have

$$g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_i \in \mathcal{C}(L) \quad \text{or} \quad f_i g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_i \in \mathcal{C}(L).$$

The former case is impossible because then the word

$$g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_i c^2 \in \mathcal{C}(L) \cdot L^2 = L^2 \cdot \mathcal{C}(L)$$

has in fact no prefix from L^2 . In the latter case we deduce

$$f_i g_i a^{m+2} b \hat{a}^{n+1} \hat{d}_i^2 \in \mathcal{C}(L) \cdot L = L \cdot \mathcal{C}(L).$$

Since $f_i g_i$ is the only suffix of this word belonging to L , we reach the desired conclusion $a^{m+2} b \hat{a}^{n+1} \hat{d}_i^2 \in \mathcal{C}(L)$.

Finally, the cases of $i \in I_2$ and $i \in D_2$ can be managed symmetrically due to (1).

To complete the proof, we have to show that the direct implication of Claim 1 holds. Let C be the set of all configurations (i, m, n) of \mathcal{M} , where $i \in Q$, $m, n \in \mathbb{N}_0$, such that there is no computation of \mathcal{M} transforming (i, m, n) to a configuration reachable from some initial configuration. We use the set C to construct a language K which commutes with L and contains the word $ab\hat{a}^{n+1}\hat{d}_1^2$ for all $n \notin \mathcal{L}(\mathcal{M})$. In the definition of K we employ the same auxiliary languages as in the definition of L .

For $m, n \in \mathbb{N}_0$, let us denote

$$K_{m,n} = (A \setminus \{a\})^* a^{m+1} b \hat{a}^{n+1} (A \setminus \{\hat{a}\})^*.$$

Then define $K = L^* \cup K'$, where

$$\begin{aligned} K' = & \bigcup \{M'_i \cap K_{m,n} \mid (i, m, n) \in C\} \\ & \cup \bigcup \{N'_i \cap K_{m,n} \mid i \in T_1 \cup D_1, (i, m, n) \in C\} \\ & \cup \bigcup \{N'_i \cap K_{m,n} \mid i \in I_1, (\tau(i), m, n) \in C\} \\ & \cup \bigcup \{\psi(\text{rev}(N'_i)) \cap K_{m,n} \mid i \in T_2 \cup D_2, (i, m, n) \in C\} \\ & \cup \bigcup \{\psi(\text{rev}(N'_i)) \cap K_{m,n} \mid i \in I_2, (\tau(i), m, n) \in C\}. \end{aligned}$$

The rest of the proof is devoted to verifying the equality $KL = LK$. Because this equality implies that $K \subseteq \mathcal{C}(L)$, and because for all $n \notin \mathcal{L}(\mathcal{M})$ we have

$$ab\hat{a}^{n+1}\hat{d}_1^2 \in M'_1 \cap K_{0,n} \subseteq K,$$

we immediately deduce $ab\hat{a}^{n+1}\hat{d}_1^2 \in \mathcal{C}(L)$, which concludes the proof.

In order to calculate the product of K and L , we introduce several auxiliary languages similar to the languages M'_i and N'_i .

For all $i \in Q$ let

$$\hat{M}'_i = d_i^2 a^+ b \hat{a}^+ \hat{d}_i \cup d_i a^+ b \hat{a}^+ \hat{d}_i^2,$$

for $i \in T_1 \cup T_2$ let

$$\hat{N}'_i = d_i^2 a b \hat{a}^+ \hat{d}_{\tau_0(i)} \cup d_i a b \hat{a}^+ \hat{d}_{\tau_0(i)}^2 \cup d_i^2 a^+ a b \hat{a}^+ \hat{d}_{\tau_1(i)} \cup d_i a^+ a b \hat{a}^+ \hat{d}_{\tau_1(i)}^2,$$

for $i \in I_1 \cup I_2$ let

$$\hat{N}'_i = g_i a^+ b \hat{a}^+ \hat{d}_i^2 \cup e_i f_i g_i a^+ b \hat{a}^+ \cdot \{\hat{d}_i, \hat{d}_{\tau(i)}\} \cup f_i g_i a^+ b \hat{a}^+ \hat{d}_{\tau(i)}^2$$

and for $i \in D_1 \cup D_2$ let

$$\hat{N}'_i = g_i a^+ b \hat{a}^+ \hat{d}_{\tau(i)}^2 \cup e_i f_i g_i a^+ b \hat{a}^+ \cdot \{\hat{d}_i, \hat{d}_{\tau(i)}\} \cup f_i g_i a^+ b \hat{a}^+ \hat{d}_i^2.$$

Finally define $P = L^+ \cup P'$, where

$$\begin{aligned} P' = & \bigcup \{ \hat{M}'_i \cap K_{m,n} \mid (i, m, n) \in C \} \\ & \cup \bigcup \{ \hat{N}'_i \cap K_{m,n} \mid i \in T_1 \cup D_1, (i, m, n) \in C \} \\ & \cup \bigcup \{ \hat{N}'_i \cap K_{m,n} \mid i \in I_1, (\tau(i), m, n) \in C \} \\ & \cup \bigcup \{ \psi(\text{rev}(\hat{N}'_i)) \cap K_{m,n} \mid i \in T_2 \cup D_2, (i, m, n) \in C \} \\ & \cup \bigcup \{ \psi(\text{rev}(\hat{N}'_i)) \cap K_{m,n} \mid i \in I_2, (\tau(i), m, n) \in C \}. \end{aligned}$$

We claim $KL = LK = P$. This fact is based on the observation that if (i, m, n) is a configuration of \mathcal{M} and (j, m', n') is the configuration obtained from (i, m, n) by one step of \mathcal{M} , then we have $(i, m, n) \in C$ if and only if $(j, m', n') \in C$. Since the computation of the product consists mainly in considering many cases according to the definitions of the languages K , L and P , let us present here only a few of them to demonstrate the most crucial arguments employed during the computation. The cases not mentioned below are either easy to deal with or similar to some of those considered here.

First notice that since both languages $K' \cdot (A^*bA^*)$ and $(A^*bA^*) \cdot K'$ are subsets of $A^*bA^*bA^* \subseteq L \subseteq P$, it is enough to show

$$P' \subseteq K' \cdot (L \setminus A^*bA^*) \subseteq P, \quad (2)$$

$$P' \subseteq (L \setminus A^*bA^*) \cdot K' \subseteq P, \quad (3)$$

where

$$L \setminus A^*bA^* = \{d_i, \hat{d}_i \mid i \in Q\} \cup \{c\} \cup \{e_i f_i, g_i a, e_i, f_i g_i \mid i \in I_1 \cup D_1\} \quad (4) \\ \cup \{g_i f_i, e_i, \hat{a} g_i, f_i e_i \mid i \in I_2 \cup D_2\}.$$

If we have $u \in K'$ and $v \in L \setminus A^*bA^*$, $v \neq c$, then in fact uv and vu belong to $L \subseteq P$ provided they do not lie in L' . This covers most of the cases; the remaining ones have to be considered one by one. In what follows let us concentrate on the equality $LK = P$; the arguments used to verify $KL = P$ are similar.

The most complicated situations arise for the word

$$u = a^{m+1}b\hat{a}^{n+1}\hat{d}_i^2 \in M'_i \cap K_{m,n} \subseteq K,$$

where $(i, m, n) \in C$ and $i \neq 0$.

Taking the word $d_i \in L$, we can see $d_i u \in \hat{M}'_i \cap K_{m,n} \subseteq P$; notice that in this way we get all words of the form $d_i a^{m+1}b\hat{a}^{n+1}\hat{d}_i^2 \in \hat{M}'_i \cap K_{m,n}$, where $(i, m, n) \in C$, except for $i = 0$.

For a word $d_j \in L$, where $j \in T_1$ and $i = \tau_0(j)$, we distinguish two cases. If $m = 0$ then one step of the machine \mathcal{M} transforms the configuration (j, m, n) to the configuration (i, m, n) , and therefore $(j, m, n) \in C$, which implies $d_j u \in \hat{N}'_j \cap K_{m,n} \subseteq P$. On the other hand, every word of the form $d_j a b \hat{a}^{n+1} \hat{d}_{\tau_0(j)}^2 \in \hat{N}'_j \cap K_{m,n}$, where $(j, 0, n) \in C$, can be obtained in this way since from $(j, 0, n) \in C$ it follows that $(\tau_0(j), 0, n) \in C$, and $\tau_0(j) \neq 0$ is one of the assumptions on our Minsky machine. If $m \geq 1$ then $d_j u \in N_j \subseteq L \subseteq P$.

For a word $d_j \in L$, where $j \in T_1$ and $i = \tau_1(j)$, the arguments are analogous to the case $i = \tau_0(j)$ with the conditions $m = 0$ and $m \geq 0$ interchanged.

Considering the situation $i \in T_2$, for the word $d_{\tau_0(i)} \in L$, again two cases have to be distinguished. If $n = 0$ then $d_{\tau_0(i)} u \in \psi(\text{rev}(\hat{N}'_i)) \cap K_{m,n} \subseteq P$; notice that all words of the form $d_{\tau_0(i)} a^{m+1}b\hat{a}^n \hat{d}_i^2 \in \psi(\text{rev}(\hat{N}'_i)) \cap K_{m,n}$ are obtained in this way because $0 \notin T_2$. If $n \geq 1$ then $d_{\tau_0(i)} u \in \psi(\text{rev}(N_i)) \subseteq L \subseteq P$. For the word $d_{\tau_1(i)} \in L$ everything is analogous.

For any word $d_j \in L$, where $j \in Q$ satisfies none of the conditions considered above, we have $d_j u \in L \subseteq P$.

For the word $c \in L$, trivially $cu \in M_i \subseteq L \subseteq P$.

Taking the word $v = f_i g_i \in L$, we have $vu \in N_i \subseteq L \subseteq P$ if $i \in I_1$ and $vu \in \hat{N}'_i \cap K_{m,n} \subseteq P$ if $i \in D_1$.

Similarly, for $v = f_j g_j \in L$ such that $i = \tau(j)$, we have $vu \in N_j \subseteq L \subseteq P$ if $j \in D_1$ and $vu \in \hat{N}'_j \cap K_{m,n} \subseteq P$ if $j \in I_1$.

Now consider the word $g_i a \in L$ and assume first that $i \in I_1$. Then $g_i a u \in \hat{N}'_i \cap K_{m+1,n} \subseteq P$ because the configuration $(\tau(i), m+1, n)$ belongs to C since it is obtained from $(i, m, n) \in C$ by one step of the machine \mathcal{M} . Conversely, all words of the form $g_i a^{m+1} b \hat{a}^{n+1} \hat{d}_i^2 \in \hat{N}'_i \cap K_{m,n}$, where $i \in I_1 \setminus \{0\}$ and $(\tau(i), m, n) \in C$, belong to LK because for $m = 0$ we have $g_i a \cdot b \hat{a}^{n+1} \hat{d}_i^2 \in N_i^2 \subseteq L^2$.

In the case $i \in D_1$, we get $g_i a u \in N_i \subseteq L \subseteq P$.

For a word $g_j a \in L$, where $j \in I_1$ and $i = \tau(j)$, clearly $g_j a u \in N_j \subseteq L \subseteq P$ holds.

And for a word $g_j a \in L$, where $j \in D_1$ and $i = \tau(j)$, we have $g_j a u \in \hat{N}'_j \cap K_{m+1,n} \subseteq P$ because from the configuration $(j, m+1, n)$ the machine \mathcal{M} goes to the configuration $(i, m, n) \in C$ and therefore $(j, m+1, n) \in C$.

For the remaining words $v \in L \setminus A^* b A^*$, it is clear that $vu \in L \subseteq P$.

The verification for the word $u = a^{m+1} b \hat{a}^{n+1} \hat{d}_0^2 \in K$, where $(0, m, n) \in C$, is even simpler than in the case of $i \neq 0$; in order to see that all words in P of the form $g_0 a^{m+1} b \hat{a}^{n+1} \hat{d}_0^2$, where $(\tau(0), m, n) \in C$, are in LK , notice that $m \neq 1$ since the configuration $(\tau(0), 1, n)$, which is obtained in one step from the initial configuration $(0, 0, n)$, does not lie in C .

Finally, another interesting case is that of the word

$$u = a^{m+1} b \hat{a}^{n+1} g_i f_i e_i \in \psi(\text{rev}(N'_i)) \cap K_{m,n} \subseteq K,$$

where $i \in D_2$ and $(i, m, n) \in C$. Then both words $d_i u$ and $d_{\tau(i)} u$ lie in the set $\psi(\text{rev}(\hat{N}'_i)) \cap K_{m,n} \subseteq P$, and for all words $v \in L \setminus A^* b A^*$ other than d_i and $d_{\tau(i)}$ we have $vu \in L \subseteq P$. \square

The following lemma establishes a connection between largest solutions and largest ε -free solutions of commutation equations. Namely, we construct for each language L a language \hat{L} such that the largest language commuting with L can be easily reconstructed from the largest ε -free language commuting with \hat{L} .

Lemma 1. *Let L be an arbitrary ε -free language over an alphabet A and let $\# \notin A$ be a new letter. Consider the alphabet $\hat{A} = A \cup \{\#\}$ and injective homomorphisms $\lambda, \rho: A^* \rightarrow \hat{A}^*$ defined by the rules $\lambda(a) = \#a$ and $\rho(a) = a\#$ for every $a \in A$. Then the language*

$$\hat{L} = \lambda(L) \cup \rho(L) \cup \{\#\} \cup \hat{A}^* \cdot \#^2 \cdot \hat{A}^*$$

over \hat{A} satisfies

$$\mathcal{C}_+(\hat{L}) = \lambda(\mathcal{C}(L)) \cdot \# \cup \hat{L}^+.$$

In particular, since $\hat{L}^+ \cap \lambda(A^*) \cdot \# = \lambda(L^*) \cdot \#$, this implies

$$\mathcal{C}_+(\hat{L}) \cap \lambda(A^*) \cdot \# = \lambda(\mathcal{C}(L)) \cdot \#.$$

Proof. Let us denote $K = \lambda(\mathcal{C}(L)) \cdot \# \cup \hat{L}^+$. We have to prove $\mathcal{C}_+(\hat{L}) = K$. One can easily verify that

$$(\hat{L} \setminus \lambda(L)) \cdot \lambda(\mathcal{C}(L)) \subseteq \hat{A}^* \cdot \#^2 \cdot \hat{A}^* \cup (\hat{L} \setminus \lambda(L)) \cdot \varepsilon \subseteq \hat{L}$$

and dually $\rho(\mathcal{C}(L)) \cdot (\hat{L} \setminus \rho(L)) \subseteq \hat{L}$, which immediately implies

$$\hat{L}K = K\hat{L} = \hat{L} \cdot \hat{L}^+ \cup \lambda(L \cdot \mathcal{C}(L)) \cdot \#.$$

Therefore K commutes with \hat{L} , and so it remains to verify $\mathcal{C}_+(\hat{L}) \subseteq K$.

First we prove the inclusion

$$\mathcal{C}_+(\hat{L}) \subseteq \lambda(A^*) \cdot \# \cup \hat{L}^+. \quad (5)$$

Observe that any non-empty word u not belonging to the set $\lambda(A^*) \cdot \# \cup \hat{L}^+$ is of one of the forms u_1abu_2 , u_1a and bu_2 , where $a, b \in A$ and $u_1, u_2 \in \hat{A}^*$. Let us assume that a word u of the form u_1abu_2 or u_1a belongs to $\mathcal{C}_+(\hat{L})$, and take any $v \in L$. Then for every $n \in \mathbb{N}$ the word $u\rho(v^n)$ belongs to $\hat{L}^n\mathcal{C}_+(\hat{L})$. Since there is no factor $\#^2$ in $u\rho(v^n)$, we actually have

$$u\rho(v^n) \in (\lambda(L) \cup \rho(L) \cup \{\#\})^n \cdot \mathcal{C}_+(\hat{L}).$$

As no two consecutive occurrences of letters from A can be found in words from $\lambda(L) \cup \rho(L) \cup \{\#\}$, taking n sufficiently large, we obtain $u_1a \in \hat{L}^+$. Because the fact $bu_2 \in \hat{L}^+$ can be proved analogously, we conclude $u \in \hat{L}^+$ in all three cases, which contradicts the choice of u .

Let us now verify that the language $M = \pi_A(\mathcal{C}_+(\hat{L}) \cap \lambda(A^*) \cdot \#)$ commutes with L , where $\pi_A: \hat{A}^* \rightarrow A^*$ denotes the projection homomorphism. Due to the symmetry, it is enough to verify only one inclusion, say $ML \subseteq LM$. In order to do this, let $u \in M$ and $v \in L$. Then $\lambda(u)\# \in \mathcal{C}_+(\hat{L})$ and $\rho(v) \in \hat{L}$, hence $\lambda(uv)\# \in \mathcal{C}_+(\hat{L}) \cdot \hat{L} = \hat{L} \cdot \mathcal{C}_+(\hat{L})$. Now two possibilities may arise: either the prefix of $\lambda(uv)\#$ belonging to \hat{L} is just $\#$ and $\rho(uv) \in \mathcal{C}_+(\hat{L})$, or there exist words $u' \in L$ and $v' \in A^*$ such that $\lambda(v')\# \in \mathcal{C}_+(\hat{L})$ and $\lambda(uv)\# = \lambda(u')\lambda(v')\#$. In the former case, we have $\rho(uv) \in \hat{L}^+$ due to (5),

consequently $uv \in L^+$, and we deduce $uv \in LM$ since $L^* \subseteq \mathcal{C}(L) \subseteq M$, which holds because $\lambda(\mathcal{C}(L)) \cdot \# \subseteq K \subseteq \mathcal{C}_+(\hat{L})$. The latter case immediately implies $uv = u'v' \in LM$.

The language M considered in the previous paragraph commutes with L , therefore $M \subseteq \mathcal{C}(L)$, and so $\mathcal{C}_+(\hat{L}) \cap \lambda(A^*) \cdot \# \subseteq \lambda(\mathcal{C}(L)) \cdot \#$. Together with (5) this shows $\mathcal{C}_+(\hat{L}) \subseteq K$, as required. \square

Now we apply this lemma to the language L constructed in Theorem 1 in order to obtain a star-free language \hat{L} such that $\mathcal{C}_+(\hat{L})$ is not recursively enumerable. In fact, we will get a stronger result in Section 5 as a by-product of encoding the language L into finitely many words, without using Lemma 1.

Theorem 2. *There exists a star-free language \hat{L} such that the largest ε -free language commuting with \hat{L} is not recursively enumerable.*

Proof. Theorem 1 provides us with a star-free language L such that $\mathcal{C}(L)$ is not recursively enumerable. Let \hat{L} be the language obtained from L by means of Lemma 1. Then a word $u \in A^*$ belongs to $\mathcal{C}(L)$ if and only if the word $\lambda(u)\#$ belongs to $\mathcal{C}_+(\hat{L})$. This implies that the language $\mathcal{C}_+(\hat{L})$ is not recursively enumerable. And since the images of any star-free language under the homomorphisms λ and ρ are star-free, it is clear that \hat{L} is star-free, as required. \square

Recall that two words u and v are *conjugated via* a word w if $uw = vw$. It is well known that u and v are conjugated if and only if there exist words u_1 and u_2 such that $u = u_1u_2$ and $v = u_2u_1$. The same notion of conjugacy has been considered also for languages, but it appears to be much more complicated than in the case of words, and only little is known about it [4]. Now we are going to apply the construction used to prove Theorem 1 to show that it is undecidable whether two star-free languages are conjugated via a language containing the empty word.

Theorem 3. *One cannot algorithmically decide whether for given two star-free languages U and V there exists a language S which contains the empty word and satisfies $US = SV$.*

Proof. In this proof, we use the notation of the proof of Theorem 1.

Let us consider a new extended alphabet $\tilde{A} = A \cup \{h\}$. For every $n \in \mathbb{N}_0$ we define languages $U = L \cup H \cup \{h\}$, where

$$H = \tilde{A}^* h \tilde{A}^* \setminus \text{Pref}(hab\hat{a}^{n+1}\hat{d}_1^2),$$

and $V = U \cup \{hab\hat{a}^{n+1}\hat{d}_1^2\}$ ($\text{Pref}(w)$ stands for the set of all prefixes of w). We claim that there exists such a language S if and only if $n \notin \mathcal{L}(\mathcal{M})$.

If S satisfies the requirements, then $S \cap A^*$ commutes with L because

$$L \cdot (S \cap A^*) = US \cap A^* = SV \cap A^* = (S \cap A^*) \cdot L.$$

From $\varepsilon \in S$ we obtain $hab\hat{a}^{n+1}\hat{d}_1^2 \in SV = US$, and therefore $ab\hat{a}^{n+1}\hat{d}_1^2 \in S$ since h is the only prefix of $hab\hat{a}^{n+1}\hat{d}_1^2$ belonging to U . Putting these facts together, we have $ab\hat{a}^{n+1}\hat{d}_1^2 \in S \subseteq \mathcal{C}(L)$, and by Claim 1 this means that $n \notin \mathcal{L}(\mathcal{M})$.

Conversely, if $n \notin \mathcal{L}(\mathcal{M})$ then we prove that $US = SV$ holds for the language $S = K \cup H$. We have

$$US = (L \cup H \cup \{h\})(K \cup H) = LK \cup H \cup hK$$

because $\tilde{A}^*H\tilde{A}^* \subseteq H$ and $\varepsilon \in K$. In addition, the language hK satisfies

$$h, hab\hat{a}^{n+1}\hat{d}_1^2 \in hK = hL^* \cup hK' \subseteq H \cup \{h, hab\hat{a}^{n+1}\hat{d}_1^2\} \quad (6)$$

since none of the words in L or K' is a proper prefix of $ab\hat{a}^{n+1}\hat{d}_1^2$.

Similarly we get

$$SV = (K \cup H)(L \cup H \cup \{h, hab\hat{a}^{n+1}\hat{d}_1^2\}) = KL \cup H \cup \{h, hab\hat{a}^{n+1}\hat{d}_1^2\}$$

due to the facts $\tilde{A}^*H\tilde{A}^* \subseteq H$ and $\tilde{A}^+h\tilde{A}^* \subseteq H$. Consequently, since the language K commutes with L , the languages US and SV are equal by (6). \square

It is still an open question whether a similar result can be proved also for conjugacy via an arbitrary language or in the case of finite languages. But Theorem 3 already allows us to conclude that existence of solutions is undecidable for very simple systems of language equations with regular constants.

Corollary 1. *It is not algorithmically decidable whether a given system of language equations with only star-free constants and operation of concatenation is solvable.*

Proof. It is sufficient to note that languages U and V over A are conjugated via a language containing the empty word if and only if the system consisting of equations $UX = XV$ and $XA^* = A^*$ possesses a solution. \square

4 Finite Languages

In this section we show how the construction described in Example 1 can be improved to obtain a finite set of words L such that both languages $\mathcal{C}(L)$ and $\mathcal{C}_+(L)$ are non-regular.

Theorem 4. *There exists a finite language L such that neither the largest language commuting with L nor the largest ε -free language commuting with L is regular.*

Proof. The basic idea of the proof is to encode the language L defined in Example 1 into a finite set of words. We achieve this by introducing a new letter s to be used for encoding states of a finite automaton recognizing the language $cM \cup Mc$ of Example 1. With this aim, take $A = \{a, \hat{a}, b, c, e, \hat{e}, f, \hat{f}, g, \hat{g}\}$ and let us consider the alphabet $\tilde{A} = A \cup \{s\}$. Further, let $\varphi: A^* \rightarrow \tilde{A}^*$ be the homomorphism defined by the formula $\varphi(x) = sxs^{17}$ for all $x \in A$.

First we define an auxiliary language

$$L'_9 = \{ef, fg, ga, aa, ab, b\hat{a}, \hat{a}\hat{a}, \hat{a}\hat{g}, \hat{g}\hat{f}, \hat{f}\hat{e}\},$$

which describes pairs of neighbouring letters in the language

$$\{efg, fg, g, \varepsilon\} \cdot a^+b\hat{a}^+ \cdot \{\varepsilon, \hat{g}, \hat{g}\hat{f}, \hat{g}\hat{f}\hat{e}\}$$

similar to the one used in Example 1. Let $\hat{L} = s^{\leq 18} \cdot A \cdot s^{\leq 18} \cup s^{\leq 18}$ and let the language L over \tilde{A} be the union of the following languages L_0 through L_{14} :

$$\begin{aligned} L_0 &= \varphi(\{c, ef, ga, e, fg, \hat{f}\hat{e}, \hat{a}\hat{g}, \hat{e}, \hat{g}\hat{f}, fgab\hat{a}^2\hat{g}\}), \\ L_1 &= \{scs^{18}es^{18}fs^{18}gs^{18}as^{18}as^{16}, s^2as^{16}, s^2bs^{15}, s^3\hat{a}s^{15}, s^3\hat{a}s^{17}\}, \\ L_2 &= \{scs^{18}gs^{14}, s^4as^{14}, s^4bs^{13}, s^5\hat{a}s^{13}, s^5\hat{g}s^{18}\hat{f}s^{17}\}, \\ L_3 &= \{scs^{18}as^{12}, s^6as^{12}, s^6bs^{11}, s^7\hat{a}s^{11}, s^7\hat{g}s^{18}\hat{f}s^{18}\hat{e}s^{17}\}, \\ L_4 &= \{scs^{18}fs^{18}gs^{10}, s^8as^{10}, s^8bs^9, s^9\hat{a}s^9, s^9\hat{g}s^{17}\}, \\ L_5 &= \{ses^{18}fs^{18}gs^{18}as^{18}as^8, s^{10}as^8, s^{10}bs^7, s^{11}\hat{a}s^7, s^{11}\hat{a}s^{18}cs^{17}\}, \\ L_6 &= \{sgs^6, s^{12}as^6, s^{12}bs^5, s^{13}\hat{a}s^5, s^{13}\hat{g}s^{18}\hat{f}s^{18}cs^{17}\}, \\ L_7 &= \{sas^4, s^{14}as^4, s^{14}bs^3, s^{15}\hat{a}s^3, s^{15}\hat{g}s^{18}\hat{f}s^{18}\hat{e}s^{18}cs^{17}\}, \\ L_8 &= \{sfs^{18}gs^2, s^{16}as^2, s^{16}bs, s^{17}\hat{a}s, s^{17}\hat{g}s^{18}cs^{17}\}, \end{aligned}$$

$$\begin{aligned}
L_9 &= \hat{L} \cdot \{xs^{18}y \mid x, y \in A \setminus \{c\}, xy \notin L'_9\} \cdot \hat{L}, \\
L_{10} &= \hat{L} \cdot (A \setminus \{c\}) \cdot s^{18}cs^{18} \cdot (A \setminus \{c\}) \cdot \hat{L}, \\
L_{11} &= \hat{L} \cdot s^{19} \cdot \hat{L}, \\
L_{12} &= \hat{L} \cdot A \cdot s^{\leq 17} \cdot A \cdot \hat{L}, \\
L_{13} &= (s^{\leq 18} \cdot A)^2 \cup (s^{\leq 18} \cdot A)^3, \\
L_{14} &= (A \cdot s^{\leq 18})^2 \cup (A \cdot s^{\leq 18})^3.
\end{aligned}$$

In essence, instead of the words that we used in Example 1, we work here with their φ -images. The languages L_{11} through L_{14} ensure that the language L^+ contains all words which are not a factor of some word from the set $\varphi(A^+)$, i.e. which are not of the form $s^i x_1 s^{18} x_2 s^{18} \dots s^{18} x_k s^j$, for $i, j \in \{0, \dots, 18\}$, $k \in \mathbb{N}_0$ and $x_1, \dots, x_k \in A$. More precisely, every word which possesses a factor belonging to one of the languages L_{11} and L_{12} lies in L^+ because this factor can be multiplied by appropriate words from the languages L_{13} from the left and L_{14} from the right (possibly with some words from L_{11} inserted in order to build blocks of occurrences of s of length more than 18). In this way we include in L^+ all words containing more than 18 consecutive occurrences of s and all words with less than 18 occurrences of s between two letters from A . Similarly we deal with certain sequences of letters from A : factors corresponding to the pairs described by the set L'_9 are determined by the language L_9 , and the language L_{10} serves for dealing with occurrences of c between letters from $A \setminus \{c\}$.

Notice that both languages L_{13} and L_{14} contain only words with at least two occurrences of letters from A . This enables us to apply the same argument that we used in Example 1 where certain words were proved not to belong to $\mathcal{C}(L)$ by concatenating with $c \in L$; the role of this letter is played by the word $\varphi(c)$ in our encoding, and the argument works well only if we ensure that multiplication by $\varphi(c)$ on one side of a word u cannot be compensated by removing a word of the same length as $\varphi(c)$ from the other side of u . This restriction on the languages L_{13} and L_{14} is also the reason for introducing the language \hat{L} to both sides of the languages L_9 through L_{12} ; this language takes care of those words where the factor under consideration is too close to a margin of the word.

The finite set of words from the definition of the language L in Example 1 is encoded in L_0 simply by taking the φ -images. And for each part of the language $cM \cup Mc$ of Example 1, all of its elements are decomposed into finitely many common segments. Then we ensure that these segments can

be put together only in the appropriate order by choosing in the definition of languages L_1 through L_8 several different powers of s ; if incompatible segments are concatenated, a block of occurrences of s of length different from 18 is produced and the resulting word cannot belong to $\varphi(A^+)$.

Finally, the reason why the language $\mathcal{C}_+(L)$ is also non-regular is that every word $u \in \mathcal{C}(L) \setminus L^*$ used in the proof is long enough to guarantee that the languages $L \cdot u$ and $u \cdot L$ contain no words from L , and so the empty word is not essential in $\mathcal{C}(L)$.

Before starting to verify that the language $\mathcal{C}(L)$ is not regular, let us analyze properties of the language L in a series of claims.

First, notice that for every $\alpha \in \mathbb{N}$, $\alpha \geq 38$, we have

$$\begin{aligned} & (s^{\leq 18} \cdot A \cup \{\varepsilon\}) \cdot s^\alpha \cdot (A \cdot s^{\leq 18} \cup \{\varepsilon\}) = \\ & = ((s^{\leq 18} \cdot A \cup \{\varepsilon\}) \cdot s^{19}) \cdot (s^{\alpha-19} \cdot (A \cdot s^{\leq 18} \cup \{\varepsilon\})) \subseteq \\ & \subseteq L_{11} \cdot (s^{\alpha-19} \cdot (A \cdot s^{\leq 18} \cup \{\varepsilon\})), \end{aligned}$$

where $\alpha - 19 \geq 19$. Applying this argument inductively, we obtain

$$(s^{\leq 18} \cdot A \cup \{\varepsilon\}) \cdot s^{19} \cdot s^* \cdot (A \cdot s^{\leq 18} \cup \{\varepsilon\}) \subseteq L_{11}^+. \quad (7)$$

Now we are going to verify the fact

$$A\tilde{A}^*A\tilde{A}^* \cup \tilde{A}^*A\tilde{A}^*A \subseteq (L_{11} \cup L_{13} \cup L_{14})^+. \quad (8)$$

Let us consider only a word $u \in A\tilde{A}^*A\tilde{A}^*$; the case of $u \in \tilde{A}^*A\tilde{A}^*A$ is symmetric. We proceed by induction on the length of the word u . First, let $u = xs^\alpha ys^\beta$, where $x, y \in A$ and $\alpha, \beta \in \mathbb{N}_0$. If both α and β are at most 18, then $u \in L_{14}$. If the numbers α and β are greater than 18, then $u = (xs^\alpha) \cdot (ys^\beta) \in L_{11}^+$ by (7). In the situation $\alpha > 18$ and $\beta \leq 18$, one can use (7) to verify $u \in L_{11}^+$. And in the remaining case of $\alpha \leq 18$ and $\beta > 18$, we obtain $u = (xs^\alpha y) \cdot s^\beta \in L_{14} \cdot L_{11}^+$ due to (7).

Second, let us consider a word u of the form $xs^\alpha ys^\beta zs^\gamma$, where $x, y, z \in A$ and $\alpha, \beta, \gamma \in \mathbb{N}_0$. For $\alpha > 18$ we have $u = (xs^\alpha) \cdot (ys^\beta zs^\gamma) \in L_{11}^+ \cdot L^+$ thanks to (7) and the induction hypothesis. The case of $\gamma > 18$ can be handled symmetrically. Assuming $\gamma \leq 18$ and $\beta > 18$, we obtain $u = (xs^\alpha y) \cdot (s^\beta zs^\gamma) \in L^+ \cdot L_{11}^+$ by the same argument. And if none of α , β and γ is greater than 18, the word u belongs to L_{14} .

Finally, let $u \in A\tilde{A}^*A\tilde{A}^*$ contain at least four occurrences of letters from A . Then u can be written as a product of two words belonging to

$A\tilde{A}^*A\tilde{A}^*$, and therefore it lies in $(L_{11} \cup L_{13} \cup L_{14})^*$ by the induction hypothesis.

Now we proceed to prove

$$\tilde{A}^* \cdot s^{19} \cdot \tilde{A}^* \subseteq (L_{11} \cup L_{13} \cup L_{14})^+. \quad (9)$$

First we verify the inclusion $s^{19}\tilde{A}^* \subseteq (L_{11} \cup L_{13} \cup L_{14})^+$. We distinguish several cases according to the number of occurrences of letters from A . Due to (7) we have $s^{19} \cdot s^* \cup s^{19} \cdot s^* \cdot A \cdot s^{\leq 18} \subseteq L_{11}^+$ and $(s^{19} \cdot s^* \cdot A) \cdot (s^{19} \cdot s^*) \subseteq L_{11}^+$. And using both (7) and (8) we get

$$(s^{19} \cdot s^*) \cdot (A\tilde{A}^*A\tilde{A}^*) \subseteq L_{11}^+ \cdot (L_{11} \cup L_{13} \cup L_{14})^+.$$

The verification of the inclusion $\tilde{A}^*s^{19} \subseteq (L_{11} \cup L_{13} \cup L_{14})^+$ is similar. Thus we obtain (9) as follows:

$$\begin{aligned} & \tilde{A}^* \cdot s^{19} \cdot \tilde{A}^* \setminus (s^{19} \cdot \tilde{A}^* \cup \tilde{A}^* \cdot s^{19}) \subseteq (\tilde{A}^*A\tilde{A}^*A \cup \{\varepsilon\}) \cdot \\ & \cdot ((s^{\leq 18} \cdot A \cup \{\varepsilon\}) \cdot s^{19} \cdot s^* \cdot (A \cdot s^{\leq 18} \cup \{\varepsilon\})) \cdot (A\tilde{A}^*A\tilde{A}^* \cup \{\varepsilon\}) \subseteq \\ & \subseteq (L_{11} \cup L_{13} \cup L_{14})^+, \end{aligned}$$

where the last inclusion follows from (7) and (8).

As the next step, let us employ (8) to calculate

$$\begin{aligned} & \tilde{A}^*A \cdot s^{\leq 17} \cdot A\tilde{A}^* \setminus \tilde{A}^* \cdot s^{19} \cdot \tilde{A}^* \subseteq \\ & \subseteq (\tilde{A}^*A\tilde{A}^*A \cup \{\varepsilon\}) \cdot L_{12} \cdot (A\tilde{A}^*A\tilde{A}^* \cup \{\varepsilon\}) \subseteq \\ & \subseteq (L_{11} \cup L_{12} \cup L_{13} \cup L_{14})^+. \end{aligned}$$

Together with (9) this implies

$$\tilde{A}^*A \cdot s^{\leq 17} \cdot A\tilde{A}^* \subseteq (L_{11} \cup L_{12} \cup L_{13} \cup L_{14})^+. \quad (10)$$

Let us now consider any letters $x, y \in A \setminus \{c\}$ such that $xy \notin L'_9$. Then

$$\begin{aligned} & \tilde{A}^*xs^{18}y\tilde{A}^* \setminus (\tilde{A}^* \cdot s^{19} \cdot \tilde{A}^* \cup \tilde{A}^*A \cdot s^{\leq 17} \cdot A\tilde{A}^*) \subseteq \\ & \subseteq (\tilde{A}^*A\tilde{A}^*A \cup \{\varepsilon\}) \cdot L_9 \cdot (A\tilde{A}^*A\tilde{A}^* \cup \{\varepsilon\}) \subseteq \\ & \subseteq (L_9 \cup L_{11} \cup L_{12} \cup L_{13} \cup L_{14})^+ \end{aligned}$$

by (8), and using (9) and (10) we conclude

$$\tilde{A}^*xs^{18}y\tilde{A}^* \subseteq (L_9 \cup L_{11} \cup L_{12} \cup L_{13} \cup L_{14})^+. \quad (11)$$

In the same way one can prove also the inclusion

$$\tilde{A}^* \cdot (A \setminus \{c\}) \cdot s^{18} c s^{18} \cdot (A \setminus \{c\}) \cdot \tilde{A}^* \subseteq (L_{10} \cup L_{11} \cup L_{12} \cup L_{13} \cup L_{14})^+. \quad (12)$$

Now consider the ε -free language

$$K = L^+ \cup \varphi(\{efga^n b \hat{a}^n, ga^n b \hat{a}^n \hat{g} \hat{f}, a^{n-1} b \hat{a}^n \hat{g} \hat{f} \hat{e}, fga^{n-1} b \hat{a}^n \hat{g} \mid n \geq 2\}).$$

Since most of the products of words from K and words from L are in L^+ due to (8)–(12) and every word in the languages $(K \setminus L^+) \cdot L$ and $L \cdot (K \setminus L^+)$ has more occurrences of letters from A than any word from L , it is easy to verify that both products KL and LK are equal to the language

$$L \cdot L^+ \cup \varphi(\{efga^n b \hat{a}^n \hat{g} \hat{f}, ga^n b \hat{a}^n \hat{g} \hat{f} \hat{e}, fga^{n-1} b \hat{a}^n \hat{g} \hat{f} \hat{e}, fga^{n-1} b \hat{a}^n \hat{g} \mid n \geq 2\}),$$

and so the language K commutes with L . Therefore the word $\varphi(efga^n b \hat{a}^n)$ belongs to $\mathcal{C}_+(L)$ for every $n \in \mathbb{N}$, $n \geq 2$.

The rest of the proof is devoted to showing that certain words do not belong to $\mathcal{C}(L)$. First observe that this is the case for the words s and s^{17} ; for instance, $s \notin \mathcal{C}(L)$ is true because from $s \in \mathcal{C}(L)$ we would obtain $scs^{17} \cdot s \in L_0 \cdot \mathcal{C}(L) \subseteq \mathcal{C}(L) \cdot L$, which is impossible as no suffix of this word lies in L . Now we employ the same idea in a more general situation to prove the following fact:

$$\mathcal{C}(L) \cap ((s \cdot A \tilde{A}^* \cdot s^{18} \cup s^{18} \cdot \tilde{A}^* A \cdot s^{17}) \setminus \tilde{A}^* \cdot (L_9 \cup L_{10} \cup L_{11} \cup L_{12}) \cdot \tilde{A}^*) = \emptyset. \quad (13)$$

In order to do this, let $u \in s \cdot A \tilde{A}^* \cdot s^{18}$ and assume it has no factor belonging to the language $L_9 \cup L_{10} \cup L_{11} \cup L_{12}$ (the case of $u \in s^{18} \cdot \tilde{A}^* A \cdot s^{17}$ can be dealt with symmetrically). Let k be the number of occurrences of letters from A in u . If $u \in \mathcal{C}(L)$ holds then we can consider the word

$$v = (scs^{17})^{k+1} \cdot u \in L_0^{k+1} \cdot \mathcal{C}(L) \subseteq \mathcal{C}(L) \cdot L^{k+1}. \quad (14)$$

Then v also does not possess a factor belonging to $L_9 \cup L_{10} \cup L_{11} \cup L_{12}$; in particular, every s -block in v except for the first one is of length 18. Therefore every suffix of v which belongs to L^* must be in fact composed solely of words from the language L_{14} . And since every word in L_{14} contains at least two occurrences of letters from A , no suffix of v can belong to L^{k+1} . This contradicts (14) and hence $u \notin \mathcal{C}(L)$.

The aim of the following considerations is to show $\varphi(efga^mb\hat{a}^n) \notin \mathcal{C}(L)$ for any $m, n \in \mathbb{N}$ satisfying $m < n$. We proceed by induction on m . Let us first consider the case of $m = 1$ and assume $\varphi(efgab\hat{a}^n) \in \mathcal{C}(L)$. Then

$$\varphi(c^{n+3}efgab\hat{a}^n) = (scs^{17})^{n+3} \cdot \varphi(efgab\hat{a}^n) \in L_0^{n+3} \cdot \mathcal{C}(L) \subseteq \mathcal{C}(L) \cdot L^{n+3}.$$

First observe that no factor of the word $\varphi(c^{n+3}efgab\hat{a}^n)$ belongs to the language $L_9 \cup L_{10} \cup L_{11} \cup L_{12}$. Let words $u_0 \in \mathcal{C}(L)$ and $u_1, \dots, u_{n+3} \in L$ satisfy $\varphi(c^{n+3}efgab\hat{a}^n) = u_0u_1 \cdots u_{n+3}$. If $u_k \in L_{14}$ for some $k \in \{1, \dots, n+3\}$, then in fact $u_1, \dots, u_k \in L_{14}$ holds because the length of every s -block in $\varphi(c^{n+3}efgab\hat{a}^n)$ (except for the first one) is 18. But this means that either $u_0 = s$ or $u_0 \in s \cdot A\tilde{A}^* \cdot s^{18}$, which contradicts (13). Therefore we have $u_1, \dots, u_{n+3} \in L \setminus L_{14}$, which is impossible since the only maximal product of words from the language $L \setminus L_{14}$ which is a suffix of $\varphi(c^{n+3}efgab\hat{a}^n)$ is

$$s^2as^{16} \cdot s^2bs^{15} \cdot (s^3\hat{a}s^{15})^{n-1} \cdot s^3\hat{a}s^{17},$$

consisting of only $n + 2$ words. This shows $\varphi(efgab\hat{a}^n) \notin \mathcal{C}(L)$.

Let $m \geq 2$ and suppose we already know that $\varphi(efga^{m-1}b\hat{a}^{n-1}) \notin \mathcal{C}(L)$. By way of contradiction, assume $\varphi(efga^mb\hat{a}^n) \in \mathcal{C}(L)$ holds. Then

$$\varphi(efga^mb\hat{a}^n\hat{g}\hat{f}) \in \mathcal{C}(L) \cdot L_0 \subseteq L \cdot \mathcal{C}(L).$$

If $\varphi(efga^mb\hat{a}^n\hat{g}\hat{f}) = u \cdot v$, where $v \in \mathcal{C}(L)$, then $u \notin L_{13}$ due to (13). Since there are just three prefixes of $\varphi(efga^mb\hat{a}^n\hat{g}\hat{f})$ belonging to $L \setminus L_{13}$, we have either $\varphi(fga^mb\hat{a}^n\hat{g}\hat{f}) \in \mathcal{C}(L)$ or $\varphi(ga^mb\hat{a}^n\hat{g}\hat{f}) \in \mathcal{C}(L)$ or $s^9 \cdot \varphi(a^{m-2}b\hat{a}^n\hat{g}\hat{f}) \in \mathcal{C}(L)$. In the first case, we obtain

$$\varphi(fga^mb\hat{a}^n\hat{g}\hat{f}c^{m+n+3}) \in \mathcal{C}(L) \cdot L_0^{m+n+3} \subseteq L^{m+n+3} \cdot \mathcal{C}(L).$$

By the same argument as in the previous paragraph, we can show that it is enough to consider only prefixes of this word which are products of elements of $L \setminus L_{13}$. There are two such maximal products, namely

$$\begin{aligned} &\varphi(fg) \cdot sas^4 \cdot (s^{14}as^4)^{m-1} \cdot s^{14}bs^3 \cdot (s^{15}\hat{a}s^3)^n, \\ &sf s^{18}gs^2 \cdot (s^{16}as^2)^m \cdot s^{16}bs \cdot (s^{17}\hat{a}s)^n, \end{aligned}$$

both consisting of $m + n + 2$ words, which is a contradiction. Similarly we can disprove the third possibility: this time we have to consider the word

$$s^9 \cdot \varphi(a^{m-2}b\hat{a}^n\hat{g}\hat{f}c^{m+n}) \in \mathcal{C}(L) \cdot L_0^{m+n} \subseteq L^{m+n} \cdot \mathcal{C}(L),$$

possessing only one maximal product of words from $L \setminus L_{13}$ as its prefix, which is of length $m + n - 1$:

$$(s^{10}as^8)^{m-2} \cdot s^{10}bs^7 \cdot (s^{11}\hat{a}s^7)^n.$$

Therefore we have $\varphi(ga^m\hat{b}\hat{a}^n\hat{g}\hat{f}) \in \mathcal{C}(L)$.

We continue in a similar manner. Multiplying the word $\varphi(ga^m\hat{b}\hat{a}^n\hat{g}\hat{f})$ by $\varphi(\hat{e})$, we obtain

$$\varphi(ga^m\hat{b}\hat{a}^n\hat{g}\hat{f}\hat{e}) \in \mathcal{C}(L) \cdot L_0 \subseteq L \cdot \mathcal{C}(L).$$

Again, it suffices to deal only with prefixes belonging to the set $L \setminus L_{13}$. This time, there are two such prefixes, corresponding to the possibilities $s^{11} \cdot \varphi(a^m\hat{b}\hat{a}^n\hat{g}\hat{f}\hat{e}) \in \mathcal{C}(L)$ and $\varphi(a^{m-1}\hat{b}\hat{a}^n\hat{g}\hat{f}\hat{e}) \in \mathcal{C}(L)$. The former one can be proved impossible by considering the word

$$s^{11} \cdot \varphi(a^m\hat{b}\hat{a}^n\hat{g}\hat{f}\hat{e}c^{m+n+2}) \in \mathcal{C}(L) \cdot L_0^{m+n+2} \subseteq L^{m+n+2} \cdot \mathcal{C}(L),$$

because the product

$$(s^{12}as^6)^m \cdot s^{12}bs^5 \cdot (s^{13}\hat{a}s^5)^n$$

is the only maximal product of elements of $L \setminus L_{13}$ which is a prefix of the word $s^{11} \cdot \varphi(a^m\hat{b}\hat{a}^n\hat{g}\hat{f}\hat{e}c^{m+n+2})$. Consequently $\varphi(a^{m-1}\hat{b}\hat{a}^n\hat{g}\hat{f}\hat{e}) \in \mathcal{C}(L)$ holds.

Now we can repeat all of the previous arguments symmetrically, first showing that $\varphi(fga^{m-1}\hat{b}\hat{a}^n\hat{g}) \in \mathcal{C}(L)$, and then $\varphi(efga^{m-1}\hat{b}\hat{a}^{n-1}) \in \mathcal{C}(L)$. This contradicts our initial assumption, and the induction step is proved.

Altogether, we have verified both $\varphi(efga^n\hat{b}\hat{a}^n) \in \mathcal{C}_+(L) \subseteq \mathcal{C}(L)$ for $n \geq 2$ and $\varphi(efga^m\hat{b}\hat{a}^n) \notin \mathcal{C}(L)$ for $m < n$, hence the languages $\mathcal{C}(L)$ and $\mathcal{C}_+(L)$ cannot be regular due to the pumping lemma. \square

5 The Main Result

Theorems 1 and 4 state that there exist a star-free language L such that $\mathcal{C}(L)$ is not recursively enumerable and a finite language L such that $\mathcal{C}(L)$ is not regular, respectively. In this section we combine the constructions of Theorems 1 and 4 in order to prove that even for a finite language L none of the languages $\mathcal{C}(L)$ and $\mathcal{C}_+(L)$ has to be recursively enumerable.

Theorem 5. *There exists a finite language L such that neither the largest language commuting with L nor the largest ε -free language commuting with L is recursively enumerable.*

Proof. In this proof we will use all of the notation and all languages which were introduced in the proof of Theorem 1. In addition, let us assume that the Minsky machine \mathcal{M} has q states, which are numbered by integers, i.e. $Q = \{0, \dots, q-1\}$. The idea of the proof is similar to the one used to prove Theorem 4; but this time we encode into a finite language \tilde{L} the language L defined in the proof of Theorem 1. We consider the alphabet $\tilde{A} = A \cup \{s\}$ and the homomorphism $\varphi: A^* \rightarrow \tilde{A}^*$ defined by the formula $\varphi(x) = s^{18q} x s^{18q}$ for all $x \in A$. Again, we start by introducing several auxiliary languages \tilde{M}_i and \tilde{N}_i , corresponding to the languages M_i and N_i , respectively, used in the proof of Theorem 1.

For $i \in Q \setminus \{0\}$ let

$$\begin{aligned} \tilde{M}_i = & \bigcup_{k=36i+1}^{36i+6} s^k \cdot \{a, \hat{a}, b\} \cdot s^{36q-k} \\ & \cup \{s^{18q} c s^{36q} d_i s^{36q} d_i s^{36q} a s^{36(q-i)-1}, s^{36i+1} \hat{a} s^{18q}\} \\ & \cup \{s^{18q} c s^{36q} d_i s^{36q} a s^{36(q-i)-2}, s^{36i+2} \hat{a} s^{36q} \hat{d}_i s^{18q}\} \\ & \cup \{s^{18q} c s^{36q} a s^{36(q-i)-3}, s^{36i+3} \hat{a} s^{36q} \hat{d}_i s^{36q} \hat{d}_i s^{18q}\} \\ & \cup \{s^{18q} d_i s^{36q} d_i s^{36q} a s^{36(q-i)-4}, s^{36i+4} \hat{a} s^{36q} c s^{18q}\} \\ & \cup \{s^{18q} d_i s^{36q} a s^{36(q-i)-5}, s^{36i+5} \hat{a} s^{36q} \hat{d}_i s^{36q} c s^{18q}\} \\ & \cup \{s^{18q} a s^{36(q-i)-6}, s^{36i+6} \hat{a} s^{36q} \hat{d}_i s^{36q} \hat{d}_i s^{36q} c s^{18q}\}, \end{aligned}$$

and for the initial state let

$$\begin{aligned} \tilde{M}_0 = & \bigcup_{k=1}^2 s^k \cdot \{a, \hat{a}, a s^{36q} b s^{36q} \hat{a}\} \cdot s^{36q-k} \\ & \cup \{s^{18q} c s^{36q} a s^{36q-1}, s \hat{d}_0 s^{36q} \hat{d}_0 s^{18q}\} \\ & \cup \{s^{18q} a s^{36q-2}, s^2 \hat{d}_0 s^{36q} \hat{d}_0 s^{36q} c s^{18q}\}. \end{aligned}$$

Similarly to the construction in the proof of Theorem 4, in the following we will use the language $\hat{L} = s^{\leq 36q} \cdot A \cdot s^{\leq 36q} \cup s^{\leq 36q}$.

For $i \in T_1 \cup T_2$ we first define a language describing words corresponding to incorrect test steps of \mathcal{M} :

$$\begin{aligned}\tilde{N}'_i &= s^{36i+11} \cdot \{a, \hat{a}, as^{36q}as^{36q}bs^{36q}\hat{a}\} \cdot s^{36(q-i)-11} \\ &\cup \hat{L} \cdot d_i s^{36(q-i)-11} \cup s^{36i+11} \hat{d}_{\tau_0(i)} \cdot \hat{L} \\ &\cup \{s^{36i+12} \hat{a} s^{36(q-i)-12}\} \\ &\cup \hat{L} \cdot d_i s^{36q}as^{36q}bs^{36q}\hat{a} s^{36(q-i)-12} \cup s^{36i+12} \hat{d}_{\tau_1(i)} \cdot \hat{L}.\end{aligned}$$

Then we consider the language

$$\begin{aligned}\tilde{N}_i &= \tilde{N}'_i \cup \bigcup_{k=36i+9}^{36i+10} s^k \cdot \{a, \hat{a}, as^{36q}as^{36q}bs^{36q}\hat{a}\} \cdot s^{36q-k} \\ &\cup \{s^{36i+7} \hat{a} s^{36(q-i)-7}, s^{36i+8} \hat{a} s^{36(q-i)-8}\} \\ &\cup \{s^{18q}cs^{36q}d_i s^{36q}as^{36q}bs^{36(q-i)-7}, s^{36i+7} \hat{a} s^{36q} \hat{d}_{\tau_0(i)} s^{18q}\} \\ &\cup \{s^{18q}d_i s^{36q}as^{36q}bs^{36(q-i)-8}, s^{36i+8} \hat{a} s^{36q} \hat{d}_{\tau_0(i)} s^{36q}cs^{18q}\} \\ &\cup \{s^{18q}cs^{36q}d_i s^{36(q-i)-9}, s^{36i+9} \hat{d}_{\tau_1(i)} s^{18q}\} \\ &\cup \{s^{18q}d_i s^{36(q-i)-10}, s^{36i+10} \hat{d}_{\tau_1(i)} s^{36q}cs^{18q}\}.\end{aligned}$$

For $i \in I_1 \cup I_2$ let

$$\begin{aligned}\tilde{N}_i &= \varphi(\{e_i f_i, g_i a, e_i, f_i g_i\}) \cup \bigcup_{k=36i+7}^{36i+14} s^k \cdot \{a, \hat{a}, b\} \cdot s^{36q-k} \\ &\cup \{s^{18q}cs^{36q}g_i s^{36q}as^{36(q-i)-7}, s^{36i+7} \hat{a} s^{36q} \hat{d}_i s^{18q}\} \\ &\cup \{s^{18q}cs^{36q}e_i s^{36q}f_i s^{36q}g_i s^{36q}as^{36(q-i)-8}, s^{36i+8} \hat{a} s^{18q}\} \\ &\cup \{s^{18q}cs^{36q}f_i s^{36q}g_i s^{36q}as^{36(q-i)-9}, s^{36i+9} \hat{a} s^{36q} \hat{d}_{\tau(i)} s^{18q}\} \\ &\cup \{s^{18q}g_i s^{36q}as^{36(q-i)-10}, s^{36i+10} \hat{a} s^{36q} \hat{d}_i s^{36q}cs^{18q}\} \\ &\cup \{s^{18q}e_i s^{36q}f_i s^{36q}g_i s^{36q}as^{36(q-i)-11}, s^{36i+11} \hat{a} s^{36q}cs^{18q}\} \\ &\cup \{s^{18q}f_i s^{36q}g_i s^{36q}as^{36(q-i)-12}, s^{36i+12} \hat{a} s^{36q} \hat{d}_{\tau(i)} s^{36q}cs^{18q}\} \\ &\cup \{s^{18q}f_i s^{36q}g_i s^{36q}as^{36(q-i)-13}, s^{36i+13} \hat{a} s^{36q} \hat{d}_i s^{36q} \hat{d}_i s^{18q}\} \\ &\cup \{s^{18q}g_i s^{36q}as^{36(q-i)-14}, s^{36i+14} \hat{a} s^{36q} \hat{d}_{\tau(i)} s^{36q} \hat{d}_{\tau(i)} s^{18q}\} \\ &\cup \{s^{18q}bs^{36(q-i)-15}, s^{36i+15} \hat{a} s^{36(q-i)-15}, s^{36i+15} \hat{a} s^{36q} \hat{d}_i s^{36q} \hat{d}_i s^{18q}\}.\end{aligned}$$

And dually, for $i \in D_1 \cup D_2$ let

$$\begin{aligned}
\tilde{N}_i = & \varphi(\{e_i f_i, g_i a, e_i, f_i g_i\}) \cup \bigcup_{k=36i+7}^{36i+14} s^k \cdot \{a, \hat{a}, b\} \cdot s^{36q-k} \\
& \cup \{s^{18q} c s^{36q} g_i s^{36q} a s^{36(q-i)-7}, s^{36i+7} \hat{a} s^{36q} \hat{d}_{\tau(i)} s^{18q}\} \\
& \cup \{s^{18q} c s^{36q} e_i s^{36q} f_i s^{36q} g_i s^{36q} a s^{36(q-i)-8}, s^{36i+8} \hat{a} s^{18q}\} \\
& \cup \{s^{18q} c s^{36q} f_i s^{36q} g_i s^{36q} a s^{36(q-i)-9}, s^{36i+9} \hat{a} s^{36q} \hat{d}_i s^{18q}\} \\
& \cup \{s^{18q} g_i s^{36q} a s^{36(q-i)-10}, s^{36i+10} \hat{a} s^{36q} \hat{d}_{\tau(i)} s^{36q} c s^{18q}\} \\
& \cup \{s^{18q} e_i s^{36q} f_i s^{36q} g_i s^{36q} a s^{36(q-i)-11}, s^{36i+11} \hat{a} s^{36q} c s^{18q}\} \\
& \cup \{s^{18q} f_i s^{36q} g_i s^{36q} a s^{36(q-i)-12}, s^{36i+12} \hat{a} s^{36q} \hat{d}_i s^{36q} c s^{18q}\} \\
& \cup \{s^{18q} f_i s^{36q} g_i s^{36q} a s^{36(q-i)-13}, s^{36i+13} \hat{a} s^{36q} \hat{d}_{\tau(i)} s^{36q} \hat{d}_{\tau(i)} s^{18q}\} \\
& \cup \{s^{18q} g_i s^{36q} a s^{36(q-i)-14}, s^{36i+14} \hat{a} s^{36q} \hat{d}_i s^{36q} \hat{d}_i s^{18q}\} \\
& \cup \{s^{18q} b s^{36(q-i)-15}, s^{36i+15} \hat{a} s^{36(q-i)-15}, s^{36i+15} \hat{a} s^{36q} \hat{d}_{\tau(i)} s^{36q} \hat{d}_{\tau(i)} s^{18q}\}.
\end{aligned}$$

Now we give the list of pairs of neighbouring letters from A in words corresponding to configurations of \mathcal{M} :

$$\begin{aligned}
L'_1 = & \{aa, ab, b\hat{a}, \hat{a}\hat{a}\} \cup \{d_i d_i, d_i a, \hat{a} \hat{d}_i, \hat{d}_i \hat{d}_i \mid i \in Q\} \\
& \cup \{e_i f_i, f_i g_i, g_i a \mid i \in I_1 \cup D_1\} \\
& \cup \{\hat{a} g_i, g_i f_i, f_i e_i \mid i \in I_2 \cup D_2\}.
\end{aligned}$$

These pairs are encoded into the language \tilde{L} by means of the sublanguage

$$\begin{aligned}
L_1 = & \hat{L} \cdot \{x s^{36q} y \mid x, y \in A \setminus \{c\}, xy \notin L'_1\} \cdot \hat{L} \\
& \cup \hat{L} \cdot \{d_i s^{36q} d_i s^{36q} d_i, \hat{d}_i s^{36q} \hat{d}_i s^{36q} \hat{d}_i \mid i \in Q\} \cdot \hat{L}.
\end{aligned}$$

Notice that the language L_1 contains also words expressing that there should be no three consecutive occurrences of d_i or \hat{d}_i in words representing configurations.

In a similar manner we can include in \tilde{L} all impossible combinations of states of \mathcal{M} . First we list all pairs corresponding to available instructions:

$$\begin{aligned}
L'_2 = & \{d_i \hat{d}_i \mid i \in Q, i \neq 0\} \\
& \cup \{d_i \hat{d}_{\tau_1(i)}, d_i \hat{d}_{\tau_0(i)} \mid i \in T_1\} \\
& \cup \{d_{\tau_0(i)} \hat{d}_i, d_{\tau_1(i)} \hat{d}_i \mid i \in T_2\}
\end{aligned}$$

$$\begin{aligned} & \cup \{g_i \hat{d}_{\tau(i)}, g_i \hat{d}_i \mid i \in I_1 \cup D_1\} \\ & \cup \{d_i g_i, d_{\tau(i)} g_i \mid i \in I_2 \cup D_2\}. \end{aligned}$$

Then we put the other combinations of d_i , \hat{d}_i and g_i into \tilde{L} ; notice that we have to add also some words containing letters a , \hat{a} and b in order to deal with letters between d_i and \hat{d}_i (or g_i):

$$\begin{aligned} L_2 = & \bigcup_{i \in Q} s^{36i+16} \cdot \{a, \hat{a}, b\} \cdot s^{36(q-i)-16} \\ & \cup \hat{L} \cdot \{d_i s^{36(q-i)-16} \mid i \in Q\} \\ & \cup \{s^{36i+16} \hat{d}_j \mid i, j \in Q, d_i \hat{d}_j \notin L'_2\} \cdot \hat{L} \\ & \cup \{s^{36i+16} g_j \mid i \in Q, j \in I_2 \cup D_2, d_i g_j \notin L'_2\} \cdot \hat{L} \\ & \cup \bigcup_{i \in I_1 \cup D_1} s^{36i+17} \cdot \{a, \hat{a}, b\} \cdot s^{36(q-i)-17} \\ & \cup \hat{L} \cdot \{g_i s^{36(q-i)-17} \mid i \in I_1 \cup D_1\} \\ & \cup \{s^{36i+17} \hat{d}_j \mid i \in I_1 \cup D_1, j \in Q, g_i \hat{d}_j \notin L'_2\} \cdot \hat{L} \\ & \cup \{s^{36i+17} g_j \mid i \in I_1 \cup D_1, j \in I_2 \cup D_2\} \cdot \hat{L}. \end{aligned}$$

The following languages are exact analogues of the languages L_{10} through L_{14} from the proof of Theorem 4:

$$\begin{aligned} L_3 &= \hat{L} \cdot (A \setminus \{c\}) \cdot s^{36q} c s^{36q} \cdot (A \setminus \{c\}) \cdot \hat{L}, \\ L_4 &= \hat{L} \cdot s^{36q+1} \cdot \hat{L}, \\ L_5 &= \hat{L} \cdot A \cdot s^{\leq 36q-1} \cdot A \cdot \hat{L}, \\ L_6 &= (s^{\leq 36q} \cdot A)^2 \cup (s^{\leq 36q} \cdot A)^3, \\ L_7 &= (A \cdot s^{\leq 36q})^2 \cup (A \cdot s^{\leq 36q})^3. \end{aligned}$$

Finally we define

$$\begin{aligned} \tilde{L} = & \{\varphi(d_i), \varphi(\hat{d}_i) \mid i \in Q\} \cup \{\varphi(c)\} \cup \bigcup_{k=1}^7 L_k \\ & \cup \bigcup \{\tilde{M}_i \mid i \in Q\} \\ & \cup \bigcup \{\tilde{N}_i \mid i \in T_1 \cup I_1 \cup D_1\} \\ & \cup \bigcup \{\psi(\text{rev}(\tilde{N}_i)) \mid i \in T_2 \cup I_2 \cup D_2\}, \end{aligned}$$

where $\psi: \tilde{A}^* \rightarrow \tilde{A}^*$ is the homomorphism which interchanges letters a and d_i , for $i \in Q$, with letters \hat{a} and \hat{d}_i , respectively.

Because the only difference between the languages L_3 through L_7 and the languages L_{10} through L_{14} from the proof of Theorem 4 is that the number $36q$ is used instead of 18 , the following facts can be proved analogously to (8) through (12):

$$A\tilde{A}^*A\tilde{A}^* \cup \tilde{A}^*A\tilde{A}^*A \subseteq (L_4 \cup L_6 \cup L_7)^+, \quad (15)$$

$$\tilde{A}^* \cdot s^{36q+1} \cdot \tilde{A}^* \subseteq (L_4 \cup L_6 \cup L_7)^+, \quad (16)$$

$$\tilde{A}^*A \cdot s^{\leq 36q-1} \cdot A\tilde{A}^* \subseteq (L_4 \cup L_5 \cup L_6 \cup L_7)^+, \quad (17)$$

$$\tilde{A}^* \cdot (A \setminus \{c\}) \cdot s^{36q} c s^{36q} \cdot (A \setminus \{c\}) \cdot \tilde{A}^* \subseteq (L_3 \cup L_4 \cup L_5 \cup L_6 \cup L_7)^+, \quad (18)$$

for $i \in T_1 \cup T_2$:

$$\tilde{A}^* d_i s^{18q} \cdot \varphi(a^+ ab \hat{a}^+) \cdot s^{18q} \hat{d}_{\tau_0(i)} \tilde{A}^* \subseteq (\tilde{N}'_i \cup L_4 \cup L_5 \cup L_6 \cup L_7)^+, \quad (19)$$

$$\tilde{A}^* d_i s^{18q} \cdot \varphi(ab \hat{a}^+) \cdot s^{18q} \hat{d}_{\tau_1(i)} \tilde{A}^* \subseteq (\tilde{N}'_i \cup L_4 \cup L_5 \cup L_6 \cup L_7)^+, \quad (20)$$

for all $x, y \in A \setminus \{c\}$:

$$xy \notin L'_1 \implies \tilde{A}^* x s^{36q} y \tilde{A}^* \subseteq (L_1 \cup L_4 \cup L_5 \cup L_6 \cup L_7)^+, \quad (21)$$

for all $i \in Q$:

$$\tilde{A}^* d_i s^{36q} d_i s^{36q} d_i \tilde{A}^* \subseteq (L_1 \cup L_4 \cup L_5 \cup L_6 \cup L_7)^+, \quad (22)$$

$$\tilde{A}^* \hat{d}_i s^{36q} \hat{d}_i s^{36q} \hat{d}_i \tilde{A}^* \subseteq (L_1 \cup L_4 \cup L_5 \cup L_6 \cup L_7)^+, \quad (23)$$

and finally, for every

$$\begin{aligned} x &\in \{d_i \mid i \in Q\} \cup \{g_i \mid i \in I_1 \cup D_1\}, \\ y &\in \{\hat{d}_j \mid j \in Q\} \cup \{g_j \mid j \in I_2 \cup D_2\} \end{aligned}$$

we have

$$xy \notin L'_2 \implies \tilde{A}^* x s^{18q} \cdot \varphi(a^* b \hat{a}^*) \cdot s^{18q} y \tilde{A}^* \subseteq (L_2 \cup L_4 \cup L_5 \cup L_6 \cup L_7)^+. \quad (24)$$

Let us first observe that the language \tilde{L} is related to the language L used to prove Theorem 1 via

$$\varphi(L) \subseteq \tilde{L}^+. \quad (25)$$

This can be verified as follows: $\varphi(A^*bA^*bA^*) \subseteq \tilde{L}^+$ holds due to (18) and (21); the inclusion $\varphi((A \setminus \{c\})^*b(A \setminus \{c\})^* \setminus L') \subseteq \tilde{L}^+$ follows from (21) through (24); for every $i \in Q$, we have $\varphi(M_i) \subseteq \tilde{M}_i^+$; for $i \in I_1 \cup I_2 \cup D_1 \cup D_2$, we similarly obtain $\varphi(N_i) \subseteq \tilde{N}_i^+$; and for $i \in T_1 \cup T_2$, we employ in addition (19) and (20) to get $\varphi(N_i) \subseteq (\tilde{N}_i \cup L_4 \cup L_5 \cup L_6 \cup L_7)^+$.

Analogously to the proof of Theorem 1, we show that none of the sets $\mathcal{C}(L)$ and $\mathcal{C}_+(L)$ is recursively enumerable by verifying the following claim:

Claim 2. For every non-negative integer $n \in \mathbb{N}_0$:

$$n \notin \mathcal{L}(\mathcal{M}) \iff \varphi(ab\hat{a}^{n+1}\hat{d}_1^2) \in \mathcal{C}(\tilde{L}) \iff \varphi(ab\hat{a}^{n+1}\hat{d}_1^2) \in \mathcal{C}_+(\tilde{L}).$$

The proof of the implication

$$\varphi(ab\hat{a}^{n+1}\hat{d}_1^2) \in \mathcal{C}(\tilde{L}) \implies n \notin \mathcal{L}(\mathcal{M}) \quad (26)$$

is almost the same as in the case of Theorem 1; we only have to replace every word employed during the proof with its φ -image. In addition, whenever some word $u \in (A \setminus \{c\})^*$ is concatenated with several copies of $c \in L$ in the proof of Theorem 1 in order to show that it does not belong to $\mathcal{C}(L)$, we have to concatenate the word $\varphi(u)$ with a certain number of copies of $\varphi(c) \in \tilde{L}$, and this number now depends on the length of u , just as in the proof of Theorem 4. In order to see that this method works properly, we have to know how suffixes of the resulting word $\varphi(c^k u)$, which belong to \tilde{L}^+ , look like (for prefixes of the word $\varphi(uc^k)$ all arguments are symmetric).

First notice that in every suffix of $\varphi(c^k u)$ each s -block (except for the initial and terminal one) consists of precisely $36q$ letters. This means that in order to get such a suffix, only words from \tilde{L} having compatible terminal and initial s -blocks (i.e. the sum of their lengths must be $36q$) can be concatenated. Then observe that none of the words $\varphi(c^k u)$ constructed during the proof contains a factor belonging to one of the languages $\varphi(N_i \setminus (cN'_i \cup N'_i c))$, for $i \in T_1$, $\varphi(\psi(\text{rev}(N_i \setminus (cN'_i \cup N'_i c))))$, for $i \in T_2$, $\varphi((A \setminus \{c\})^*b(A \setminus \{c\})^* \setminus L')$, L_1 , L_3 , L_4 and L_5 .

Therefore, if words $u_1, \dots, u_l \in \tilde{L}$ are such that $u_1 \cdots u_l$ is a suffix of $\varphi(c^k u)$ and for some $p \in \{1, \dots, l\}$ we have

$$u_p \in \bigcup_{i \in T_1} \tilde{N}'_i \cup \bigcup_{i \in T_2} \psi(\text{rev}(\tilde{N}'_i)) \cup L_2 \cup L_7,$$

then we can inductively verify that none of the words u_1, \dots, u_p belongs to one of the languages starting on \tilde{L} in the definitions of \tilde{N}'_i , $\psi(\text{rev}(\tilde{N}'_i))$ and L_2 , and

the length of the terminal s -block in each of the words u_1, \dots, u_{p-1} is different from $18q$. In particular, all words u_1, \dots, u_p containing some occurrence of c lie in L_7 , and we can apply the same argument that we used to prove (13) to conclude that $\varphi(u) \notin \mathcal{C}(\tilde{L})$.

Hence, it only makes sense to consider suffixes of $\varphi(c^k u)$ constructed from words belonging to the languages $\{\varphi(d_i), \varphi(\hat{d}_i) \mid i \in Q\}$, $\{\varphi(c)\}$, \tilde{M}_i , for $i \in Q$, $\tilde{N}_i \setminus \tilde{N}'_i$, for $i \in T_1$, $\psi(\text{rev}(\tilde{N}_i \setminus \tilde{N}'_i))$, for $i \in T_2$, \tilde{N}_i , for $i \in I_1 \cup D_1$, and $\psi(\text{rev}(\tilde{N}_i))$, for $i \in I_2 \cup D_2$. And because the word $c^k u$ always belongs to the language $R = (A \setminus \{a, \hat{a}, b\})^* a^* b a^* (A \setminus \{a, \hat{a}, b\})^*$ and we have

$$\begin{aligned} \tilde{M}_i^+ \cap \varphi(R) &= \varphi(M_i^+), & \text{for } i \in Q, \\ (\tilde{N}_i \setminus \tilde{N}'_i)^+ \cap \varphi(R) &= \varphi((cN'_i \cup N'_i c)^+), & \text{for } i \in T_1 \cup T_2, \\ \tilde{N}_i^+ \cap \varphi(R) &= \varphi(N_i^+), & \text{for } i \in I_1 \cup I_2 \cup D_1 \cup D_2, \end{aligned}$$

the arguments about $c^k u$ from the proof of Theorem 1 can be directly translated to the case of the word $\varphi(c^k u)$ and the language \tilde{L} .

Let us now illustrate how the technique works on the proof of the analogue of equivalence (1), i.e. for all $i \in Q \setminus \{0\}$ and $m, n \in \mathbb{N}_0$ we have to prove

$$\varphi(a^{m+1} b \hat{a}^{n+1} \hat{d}_i^2) \in \mathcal{C}(\tilde{L}) \iff \varphi(d_i^2 a^{m+1} b \hat{a}^{n+1}) \in \mathcal{C}(\tilde{L}).$$

Assuming $\varphi(a^{m+1} b \hat{a}^{n+1} \hat{d}_i^2) \in \mathcal{C}(\tilde{L})$, we obtain

$$\varphi(d_i^2 a^{m+1} b \hat{a}^{n+1} \hat{d}_i^2) \in \tilde{L}^2 \cdot \mathcal{C}(\tilde{L}) = \mathcal{C}(\tilde{L}) \cdot \tilde{L}^2.$$

Unlike in the proof of Theorem 1, there exist many suffixes of the word $\varphi(d_i^2 a^{m+1} b \hat{a}^{n+1} \hat{d}_i^2)$ belonging to \tilde{L}^2 , and all of them have to be dealt with. So let us show, for instance, that we do not obtain an element of $\mathcal{C}(\tilde{L})$ by removing from $\varphi(d_i^2 a^{m+1} b \hat{a}^{n+1} \hat{d}_i^2)$ the suffix

$$s^{36(q-i)-11} \hat{d}_i s^{36(q-j)-16} \cdot s^{36j+16} \hat{d}_i s^{18q} \in \psi(\text{rev}(\tilde{N}'_i)) \cdot L_2,$$

where $i \in T_2$ and $j \in Q$ satisfy $d_j \hat{d}_i \notin L'_2$.

By way of contradiction, assume that

$$\varphi(d_i^2 a^{m+1} b \hat{a}^n) \cdot s^{18q} \hat{a} s^{36i+11} \in \mathcal{C}(\tilde{L})$$

and consider the word

$$v = \varphi(c^k d_i^2 a^{m+1} b \hat{a}^n) \cdot s^{18q} \hat{a} s^{36i+11} \in \tilde{L}^k \cdot \mathcal{C}(\tilde{L}) = \mathcal{C}(\tilde{L}) \cdot \tilde{L}^k, \quad (27)$$

where $k > \max(2m + 2n, m + 2n + 4)$. Because $\tau_0(i) \neq i$, no word from the language $\hat{L} \cdot d_{\tau_0(i)} s^{36i+11} \subseteq \psi(\text{rev}(\tilde{N}'_i))$ is a factor of v . Therefore for any suffix of v belonging to \tilde{L}^+ it can be shown that every decomposition of this suffix into elements of \tilde{L} is either of the form $L_7^* \cdot (s^{36(q-i)-11} \hat{a} s^{36i+11})^*$ or of the form

$$L_7^* \cdot (s^{36(q-i)-11} a s^{36i+11})^* \cdot (s^{36(q-i)-11} a s^{36q} b s^{36q} \hat{a} s^{36q} \hat{a} s^{36i+11}) \cdot (s^{36(q-i)-11} \hat{a} s^{36i+11})^* .$$

Then it is easy to see that due to the choice of k no such suffix of v can be decomposed into k elements of \tilde{L} , contradicting (27).

To complete the proof, we have to show that the implication

$$n \notin \mathcal{L}(\mathcal{M}) \implies \varphi(ab\hat{a}^{n+1}\hat{d}_1^2) \in \mathcal{C}_+(\tilde{L}) \quad (28)$$

of Claim 2 holds. Let us consider the language $\tilde{K} = \tilde{L}^+ \cup \varphi(K')$, where K' is the language used in the proof of Theorem 1 for the same purpose. We are going to verify the equality $\tilde{K}\tilde{L} = \tilde{L}\tilde{K}$, which shows that \tilde{K} is an ε -free language commuting with \tilde{L} . Since for every $n \notin \mathcal{L}(\mathcal{M})$ we have $\varphi(ab\hat{a}^{n+1}\hat{d}_1^2) \in \varphi(M'_1 \cap K_{0,n}) \subseteq \tilde{K}$, this implies $\varphi(ab\hat{a}^{n+1}\hat{d}_1^2) \in \mathcal{C}_+(\tilde{L})$, as required.

We claim $\tilde{K}\tilde{L} = \tilde{L}\tilde{K} = \tilde{P}$, where $\tilde{P} = \tilde{L}\tilde{L}^+ \cup \varphi(P')$. Let us verify only $\tilde{K}\tilde{L} = \tilde{P}$, the other equality $\tilde{L}\tilde{K} = \tilde{P}$ can be proved symmetrically. First, one can use (15) through (23) to show that

$$\tilde{K} \cdot (\tilde{L} \setminus \varphi(L \setminus A^*bA^*)) \subseteq \tilde{L}\tilde{L}^+ ,$$

where the language $L \setminus A^*bA^*$ was already considered in the proof of Theorem 1 and is given by (4). Because $\varphi(L \setminus A^*bA^*) \subseteq \tilde{L}$, this means that

$$\tilde{K}\tilde{L} = \tilde{L}\tilde{L}^+ \cup \varphi(K') \cdot \varphi(L \setminus A^*bA^*) .$$

From (2) we get

$$\varphi(P') \subseteq \varphi(K') \cdot (L \setminus A^*bA^*) \subseteq \varphi(L^+) \cup \varphi(P') ,$$

which implies the required equality $\tilde{K}\tilde{L} = \tilde{L}\tilde{L}^+ \cup \varphi(P') = \tilde{P}$ due to (25) and due to the fact that all words in $\tilde{K}\tilde{L} \cap \tilde{L}^+$ clearly belong to $\tilde{L}\tilde{L}^+$.

Now we can obtain Claim 2 by combining (26) and (28) with the obvious inclusion $\mathcal{C}_+(\tilde{L}) \subseteq \mathcal{C}(\tilde{L})$. \square

Let us conclude by mentioning that all results on commutation proved in this paper hold also if we consider only languages over a two-letter alphabet. This is true because any finite alphabet can be encoded into two letters in such a way that solutions of commutation equations are preserved:

Lemma 2. *Let L be an arbitrary ε -free language over an n -letter alphabet $A = \{a_1, \dots, a_n\}$, and let $\varphi: A^* \rightarrow \{a, b\}^*$ be the homomorphism defined by the rule $\varphi(a_i) = a^i b$. Then a language commutes with $\varphi(L)$ if and only if it is of the form $\varphi(K)$ for some language K over A commuting with L .*

Proof. Assume that a language M over $\{a, b\}$ commutes with $\varphi(L)$. Because $(\varphi(L))^n \cdot M = M \cdot (\varphi(L))^n$ for an arbitrarily large n , every non-empty word $w \in M$ is a prefix of some element of $\varphi(L)^+$. On the other hand, w must be also a suffix of a word from $\varphi(L)^+$, which in particular means that its last letter is b . This is possible only if w belongs to $\varphi(A^+)$, and therefore $M = \varphi(K)$ for some language $K \subseteq A^*$. Now the statement immediately follows from the fact that φ defines an isomorphism between the monoids A^* and $\varphi(A^*)$. \square

Acknowledgements.

The idea to use the results on commutation to prove undecidability of conjugacy of languages was suggested to me by Alexander Okhotin. I am also grateful to Ondřej Klíma for several useful comments.

References

- [1] Aiken, A., Kozen, D., Vardi, M., Wimmers, E.: The complexity of set constraints. In *Proc. CSL '93*, LNCS 832, Springer (1994) 1–17.
- [2] Baader, F., Küsters, R.: Unification in a description logic with transitive closure of roles. In *Proc. LPAR 2001*, LNCS 2250, Springer (2001) 217–232.
- [3] Bergman, G.M.: Centralizers in free associative algebras. *Trans. Amer. Math. Soc.* **137** (1969) 327–344.
- [4] Cassaigne, J., Karhumäki, J., Manuch, J.: On conjugacy of languages. *Theor. Inform. Appl.* **35**(6) (2001) 535–550.

- [5] Charatonik, W., Podelski, A.: Co-definite set constraints. In *Proc. RTA-98*, LNCS 1379, Springer (1998) 211–225.
- [6] Choffrut, C., Karhumäki, J., Ollinger, N.: The commutation of finite sets: A challenging problem. *Theoret. Comput. Sci.* **273** (2002) 69–79.
- [7] Cohn, P.M.: Factorization in non-commutative power series rings. *Proc. Cambridge Philos. Soc.* **58** (1962) 452–464.
- [8] Conway, J.H.: *Regular Algebra and Finite Machines*. Chapman and Hall (1971).
- [9] Karhumäki J., Latteux M., Petre, I.: Commutation with ternary sets of words. *Theory Comput. Syst.* **38**(2) (2005) 161–169.
- [10] Karhumäki J., Latteux M., Petre, I.: Commutation with codes. *Theoret. Comput. Sci.* **340**(2) (2005) 322–333.
- [11] Karhumäki, J., Petre, I.: The branching point approach to Conway’s problem. In *Formal and Natural Computing*, LNCS 2300, Springer (2002) 69–76.
- [12] Karhumäki, J., Petre, I.: Two problems on commutation of languages. In *Current Trends in Theoretical Computer Science, The Challenge of the New Century, vol. 2*, World Scientific (2004) 477–494.
- [13] Kunc, M.: Regular solutions of language inequalities and well quasi-orders. *Theoret. Comput. Sci.* **348**(2–3) (2005) 277–293.
- [14] Kunc, M.: Simple language equations. *Bull. Eur. Assoc. Theor. Comput. Sci. EATCS* **85** (2005) 81–102.
- [15] Kunc, M.: On language inequalities $XK \subseteq LX$. In *Proc. DLT 2005*, LNCS 3572, Springer (2005) 327–337.
- [16] Kunc, M.: Largest solutions of left-linear language inequalities. In *Proc. AFL 2005*, University of Szeged (2005) 178–186. Also available at <http://www.math.muni.cz/~kunc/>
- [17] Leiss, E.L.: *Language Equations*. Springer (1999).

- [18] Minsky, M.L.: *Computation: Finite and Infinite Machines*. Prentice-Hall (1967).
- [19] Okhotin, A.: Decision problems for language equations. Submitted for publication, available at <http://www.cs.queensu.ca/home/okhotin/>. Preliminary version in *Proc. ICALP 2003*, LNCS 2719, Springer (2003) 239–251.
- [20] Okhotin, A.: On computational universality in language equations. In *Proc. MCU 2004*, LNCS 3354, Springer (2005) 292–303.
- [21] Rabin, M.O.: Decidability of second-order theories and automata on infinite trees. *Trans. Amer. Math. Soc.* **141** (1969) 1–35.
- [22] Ratoandromanana, B.: Codes et motifs. *RAIRO Inform. Théor. Appl.* **23** (1989) 425–444.
- [23] Rozenberg, G., Salomaa, A. (eds.): *Handbook of Formal Languages*. Springer (1997).