# THE MONOID $\langle a, b \,|\, aba \Leftrightarrow aa,\, baa \Leftrightarrow aab \rangle$ HAS NO FINITE COMPLETE PRESENTATION

SLAVA PESTOV

Abstract. This article describes a monoid with only two defining relations, but no finite complete presentation over any alphabet. This improves upon what is perhaps the previous smallest known example of three rules.

## 1. Introduction

The *word problem* is central to the theory of finitely-presented monoids:

> **The word problem.** Given a finite monoid presentation $\langle A \,|\, R \rangle$ and two words $x$, $y$ over this alphabet $A$, can we rewrite $x$ into $y$ by applying a finite sequence of rules from $R$?

The word problem is undecidable in the general case [1]. On the other hand, with a *complete* monoid presentation, it suffices to view the rewrite rules as directed reductions, which are repeatedly applied to a word until fixed point. We can then solve the word problem by computing the normal form of both words, and checking for string equality [2]. Various other questions are decidable from a finite complete presentation, such as whether the presented monoid is finite, or a group [3].

The *Knuth-Bendix algorithm* attempts to construct a complete presentation by adding new rules [4, 5, 6]. This either ends with a finite complete presentation, or continues forever. A successful outcome depends both on choice of reduction order, and the alphabet used to present the monoid [7, 8]. It is also known that there are finitely-presented monoids with decidable word problem, but no finite complete presentation over any alphabet. Section 2 starts off with a survey of existing results in this space, Section 3 fixes notation for what follows, and Section 4 contains the main result that $\langle a, b \,|\, aba \Leftrightarrow aa,\, baa \Leftrightarrow aab \rangle$ has no finite complete presentation.

## 2. Related Work

**Example 1.** A monoid with an undecidable word problem cannot have a finite complete presentation. Tseitin's classic example has 5 letters and 7 rules [9, 10]:

$$\begin{aligned}\mathfrak{C}_1 := \langle a, b, c, d, e \,|\, & ac \Leftrightarrow ca,\, ad \Leftrightarrow da,\, bc \Leftrightarrow cb,\, bd \Leftrightarrow db,\\ & eca \Leftrightarrow ce,\, edb \Leftrightarrow de,\\ & cca \Leftrightarrow ccae \rangle\end{aligned}$$

**Example 2.** A result due to Craig C. Squier is that a monoid with a finite complete presentation must satisfy $FP_3$, a property whose definition we do not need. Squier showed that $S_k$ has a decidable word problem, but not $FP_3$ when $k \geq 2$ [11]:

$$\begin{aligned}S_k := \langle a, b, t, x_1, \ldots, x_k, y_1, \ldots, y_k \,|\, & ab \Leftrightarrow 1,\\ & x_1 a \Leftrightarrow a t x_1, \quad \ldots, \quad x_k a \Leftrightarrow a t x_k,\\ & x_1 t \Leftrightarrow t x_1, \quad \ldots, \quad x_k t \Leftrightarrow t x_k,\\ & x_1 b \Leftrightarrow b x_1, \quad \ldots, \quad x_k b \Leftrightarrow b x_k,\\ & x_1 y_1 \Leftrightarrow 1, \quad \ldots, \quad x_k y_k \Leftrightarrow 1 \rangle\end{aligned}$$

**Example 3.** Squier settled the status of $S_1$ in a subsequent paper, by showing that if a monoid has a finite complete presentation, it also has a property that he named *finite derivation type*, while $S_1$, with 5 letters and 5 rules, is not FDT. Thus, it has no finite complete presentation [12]:

$$S_1 := \langle a, b, t, x, y \,|\, ab \Leftrightarrow 1,\ xa \Leftrightarrow atx,\ xt \Leftrightarrow tx,\ xb \Leftrightarrow bx,\ xy \Leftrightarrow 1 \rangle$$

We do not require the definition of FDT for the main proof, but a few more sources are cited for the interested reader. It is known that FDT implies $FP_3$ [13].

**Example 4.** A key fact is that FDT is not a *sufficient* condition for a monoid to admit a finite complete presentation. Katsura and Kobayashi gave an example with 10 letters and 7 rules, having a word problem decidable in linear time, finite derivation type, but still no finite complete presentation [14]:

$$\begin{aligned}\langle a, b_1, c_1, d_1, b_2, c_2, d_2, b_3, c_3, d_3 \,|\, & b_1 a \Leftrightarrow ab_1,\ b_2 a \Leftrightarrow ab_2,\ b_3 a \Leftrightarrow ab_3,\\ & c_1 b_1 \Leftrightarrow c_1 b_1,\ c_2 b_2 \Leftrightarrow c_1 b_1,\\ & b_1 d_1 \Leftrightarrow b_1 d_1,\ b_2 d_2 \Leftrightarrow b_1 d_1 \rangle\end{aligned}$$

Our result was heavily inspired by some of the techniques here, in particular looking at the properties of $\textsc{Irr}(R)$. In their terminology, $\langle a, b \,|\, aba \Leftrightarrow aa,\ baa \Leftrightarrow aab \rangle$ does not admit a *finite s-closed transversal*.

**Example 5.** Cain et. al. found an example with only 3 letters and 3 rules. This monoid is not FDT, and thus has no finite complete presentation [15]:

$$\langle a, b, c \,|\, ac \Leftrightarrow ca,\ bc \Leftrightarrow cb,\ cab \Leftrightarrow cbb \rangle$$

This is the shortest example the author of this article has seen in the literature. It is also notable for it is *homogeneous* (presented by length-preserving rules).

**Example 6.** It is not known if every one-relation monoid has a finite complete presentation, or if the word problem is decidable for all such monoids. An excellent survey of this subject appears in [16], from which we quote:

> ... the smallest monadic one-relation monoid to which no result in the literature appears to be available to solve the word problem for is $\langle a, b \,|\, bababbbabba \Leftrightarrow a \rangle$. The author has not found a finite complete rewriting system for this monoid, but has solved the word problem for this monoid by other means.

Here is a finite complete presentation of $\langle a, b \,|\, bababbbabba \Leftrightarrow a \rangle$ over $\{a, b, c\}$:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | $bababbbabba$ | $\Rightarrow$ | $a$ | | | |
| $caa$ | $\Rightarrow$ | $bacca$ | $cac$ | $\Rightarrow$ | $baccc$ | $cba$ | $\Rightarrow$ | $a$ |
| $bbaa$ | $\Rightarrow$ | $abba$ | $acabc$ | $\Rightarrow$ | $abacc$ | $bbaaa$ | $\Rightarrow$ | $aabba$ |
| $bbaac$ | $\Rightarrow$ | $aabbc$ | $cabaa$ | $\Rightarrow$ | $babacca$ | $ababac$ | $\Rightarrow$ | $cccccc$ |
| $acabaa$ | $\Rightarrow$ | $cccccca$ | $acabac$ | $\Rightarrow$ | $cccccccc$ | $acabba$ | $\Rightarrow$ | $abaa$ |
| $acabbc$ | $\Rightarrow$ | $abac$ | $bbaaca$ | $\Rightarrow$ | $aaba$ | $bbababc$ | $\Rightarrow$ | $abbbabc$ |
| $cababbc$ | $\Rightarrow$ | $babac$ | $cabacca$ | $\Rightarrow$ | $babacccca$ | $ababbbac$ | $\Rightarrow$ | $cccc$ |
| $abbbaaca$ | $\Rightarrow$ | $abaaba$ | $abbbaacc$ | $\Rightarrow$ | $abaabc$ | $acababbc$ | $\Rightarrow$ | $cccccc$ |
| $bbababbc$ | $\Rightarrow$ | $abbbab$ | $bbacccca$ | $\Rightarrow$ | $acca$ | $bbaccccc$ | $\Rightarrow$ | $accc$ |

Every one-relation monoid is FDT [17].

## 3. Preliminaries

This article assumes some familiarity with finitely-presented monoids and string rewriting; a complete (no pun intended) treatment of the subject can be found in [18]. This section will summarize the notation and terminology that we will need, but it is too dense to serve as an introduction to the topic.

**Definition 1.** A *monoid* is a set with an associative binary operation and identity element. If $A$ is any set, the *free monoid* $A^*$ is the set of all finite sequences of elements of $A$.

- The *length* of $x \in A^*$ is denoted by $|x| \geq 0$.
- An $x \in A$ is a *letter* and an $x \in A^*$ is a *word*.
- The unique *empty word* of length 0 is denoted by 1.
- We view each letter $x \in A$ as a word of length 1 in $A^*$.
- The concatenation of words $x$ and $y$ is denoted $xy$ or $x \cdot y$.
- A word $u$ is a *factor* of a word $w$ if $w = xuy$ for some $x,\ y \in A^*$.
- The equality operator $=$ denotes graphical equality of words in $A^*$.
- A *morphism* of free monoids $\phi\colon A^* \to B^*$ satisfies $\phi(x \cdot y) = \phi(x) \cdot \phi(y)$ for all $x,\ y \in A^*$. A morphism is completely determined by the image of each letter $a \in A$. If $\phi(a) \neq 1$ for all $a \in A$, we say $\phi$ is *non-erasing*.

A *monoid presentation* is a pair $\langle A \mid R\rangle$, where $A$ is a set, and $R \subset A^* \times A^*$ is a set of ordered pairs of words. A presentation is *finite* if $A$ and $R$ are finite sets.

- The *one-step monoid congruence* $\Leftrightarrow^1_R$ on $A^*$ relates all pairs of words:
$$xuy \Leftrightarrow^1_R xvy \qquad \text{where } x,\ y \in A^*, \text{ and either } (u, v) \text{ or } (v, u) \in R$$
- The *monoid congruence* $\Leftrightarrow_R$ is the reflexive and transitive closure of $\Leftrightarrow^1_R$.
- The equivalence classes of $\Leftrightarrow_R$ then have the structure of a monoid, with identity element $[\![1]\!]$ and binary operation $[\![x]\!] \cdot [\![y]\!] := [\![x \cdot y]\!]$.
- A *finitely-presented monoid* is one that is isomorphic to a monoid obtained in this way, from a finite presentation.

We also need to consider rewriting steps that only apply a rule from left to right:

- The *one-step reduction relation* $\Rightarrow^1_R$ on $A^*$ relates all pairs of words:
$$xuy \Rightarrow^1_R xvy \qquad \text{where } x,\ y \in A^*, \text{ and } (u, v) \in R$$
- The *reduction relation* $\Rightarrow_R$ is the reflexive and transitive closure of $\Rightarrow^1_R$.
- The reduction relation $\Rightarrow_R$ is *terminating* if there is no infinite sequence of one-step reductions:
$$x_1 \Rightarrow^1_R x_2 \Rightarrow^1_R x_3 \Rightarrow^1_R \cdots$$
- The reduction relation $\Rightarrow_R$ is *confluent* if whenever $x \Rightarrow_R y$ and $x \Rightarrow_R z$, there exists a word $w \in A^*$ such that $y \Rightarrow_R w$ and $z \Rightarrow_R w$.
- A word $x \in A^*$ is *irreducible* if $x \Rightarrow_R y$ implies that $x = y$.
- The set of irreducible words of $\Rightarrow_R$ is denoted by $\textsc{Irr}(R)$.
- If $y$ is irreducible and $x \Rightarrow_R y$, we say that $y$ is a *normal form* for $x$.

A monoid presentation $\langle A \mid R\rangle$ is *complete* if $\Rightarrow_R$ is terminating and confluent. A *finite complete presentation* is one that is both finite, and complete. In this case, every equivalence class of $\Leftrightarrow_R$ has an effectively computable, unique normal form.

**Definition 2.** Let $A^*$ be the free monoid over some set $A$. We define the family of *regular* subsets of $A^*$ as follows:

(1) If $X \subset A^*$ is a finite set of words, then $X$ is regular.
(2) If $X,\ Y \subset A^*$ are regular, their union $X \cup Y$ is regular.
(3) If $X,\ Y \subset A^*$ are regular, their concatenation $XY$ is regular. This is the set of all words $xy$ where $x \in X$ and $y \in Y$.
(4) We write $X^n$ to mean $X \cdots X$, repeated $n$ times, with $X^0 := \{1\}$.
(5) If $X \subset A^*$ is regular, then $X^*$ is regular. This is the infinite union of $X^n$ over all $n \geq 0$:
$$X^* := \{1\} \cup X \cup XX \cup XXX \cdots$$
(6) It is convenient to define $X^+ := XX^* = X^* \setminus \{\{1\}\}$.

We rely on these two well-known closure properties. For proof, we refer to [19]:

- If $X$ is regular, the complement $A^* \setminus X$ is regular.
- If $X$ is regular, the set formed by reversing each word in $X$ is regular.

Next, we need these properties of $\textsc{Irr}(R)$, the set of irreducible words of $\Rightarrow_R$.

**Lemma 3.** Let $\langle A \,|\, R\rangle$ be any monoid presentation.

(1) If $x \notin \textsc{Irr}(R)$ and $y \in A^*$, then $xy \notin \textsc{Irr}(R)$ and $yx \notin \textsc{Irr}(R)$.
(2) If $xy \in \textsc{Irr}(R)$, then $x \in \textsc{Irr}(R)$ and $y \in \textsc{Irr}(R)$.
(3) If $R$ is finite, then $\textsc{Irr}(R)$ is regular.

*Proof.* If $x \notin \textsc{Irr}(R)$, then $x$ has a factor $u$ for some $(u, v) \in R$. Thus, for any $y \in A^*$, both $xy$ and $yx$ also have a factor of $u$, so $xy \notin \textsc{Irr}(R)$ and $yx \notin \textsc{Irr}(R)$. This is (1), and (2) is the contrapositive statement. For (3), suppose that $R$ is finite, and let $n := |R|$. The complement of $\textsc{Irr}(R)$ in $A^*$ is the regular set of all words that contain a left-hand side of $R$ as a factor:

$$A^* \setminus \textsc{Irr}(R) = (A^* \cdot \{u_1\} \cdot A^*) \cup \cdots \cup (A^* \cdot \{u_n\} \cdot A^*)$$

It follows that $\textsc{Irr}(R)$ is a regular set. □

We also use the pumping lemma for regular sets. A proof can be found in [19].

**Lemma 4** (Pumping lemma)**.** Let $X \subseteq A^*$ be a regular set. Then there exists a natural number $\ell > 0$ such that any word $u \in X$ with $|u| \geq \ell$ has a factorization:

$$u = xyz$$

with the property that $|y| > 0$, $|xy| \leq \ell$, and for all $n \geq 0$:

$$xy^n z \in X$$

**Remark 5.** When $X$ is finite, Lemma 4 is vacuously true; we take $\ell$ to be longer than the longest word in $X$.

Now, suppose we have an $u \in X$ with $|u| \geq \ell$. The above statement yields a factorization with $|x| \geq 0$ and $|z| \geq |u| - \ell$. Since the family of regular sets is closed under reversal, there is a dual statement to the above, which produces a factorization where $|yz| \leq \ell$ instead, so $|x| \geq |u| - \ell$ and $|z| \geq 0$. Our proof uses the "prefix" and "suffix" form of the lemma, once each.

We're going to apply the pumping lemma to the irreducible words of a finite complete presentation. The following fact will then become relevant:

**Lemma 6.** Let $\langle A \,|\, R\rangle$ be any monoid presentation. If $xy \in \textsc{Irr}(R)$, $yz \in \textsc{Irr}(R)$, but $xyz \notin \textsc{Irr}(R)$, then there must exist a rule $(u, v) \in R$ with $|u| \geq |y|$.

*Proof.* Since $xyz \notin \textsc{Irr}(R)$, we can write $xyz = x'uz'$ where $u$ is the left-hand side of a rule in $R$, and $x'$, $y' \in A^*$. Suppose that to the contrary, $|u| < |y|$. Either $|xy| \geq |x'u|$, or $|xy| < |x'u|$. If $|xy| \geq |x'u|$, then $xy \in \textsc{Irr}(R)$ has a prefix $x'u \notin \textsc{Irr}(R)$, which is impossible. On the other hand, if $|xy| < |x'u|$, then $|z| \geq |z'|$, and now $uz' \notin \textsc{Irr}(R)$ is a suffix of $yz \in \textsc{Irr}(R)$, which is again impossible. So $|u| \geq |y|$. □

## 4. The Monoid $\langle a, b \,|\, aba \Leftrightarrow aa,\, baa \Leftrightarrow aab\rangle$

Henceforth, $A$ and $R$ will always denote the alphabet and rules of our monoid presentation $\langle a, b \,|\, aba \Leftrightarrow aa,\, baa \Leftrightarrow aab\rangle$:

$$A := \{a, b\}$$
$$R := \{(aba, aa), (baa, aab)\}$$

The monoid congruence generated by $R$ will be denoted by $\Leftrightarrow$.

When we construct certain regular sets below, we will let $a$ and $b$ denote the singleton sets $\{a\}$ and $\{b\}$, so for example, $b^* \cup b^*a$ is the set of all words of the form $b^n$ or $b^n a$, for all $n \geq 0$.

**Lemma 7.** We make use of four facts about $\Leftrightarrow$:

(1) If $x \Leftrightarrow aab^n$, then $x = b^i aab^{n-i}$ or $x = b^i abab^{n-i}$, for some $0 \leq i \leq n$.
(2) Let $\tilde{x}$ denote the reversal of $x \in A^*$. Then $x \Leftrightarrow y$ if and only if $\tilde{x} \Leftrightarrow \tilde{y}$.
(3) Every word in $b^* \cup b^*a$ is a singleton equivalence class of $\Leftrightarrow$.
(4) For all $n \geq 0$, we have $b^n aaa \Leftrightarrow aaa$.

*Proof.* For (1), we apply $baa \Leftrightarrow aab$ and $aba \Leftrightarrow aa$. For (2), it suffices to note that for each $(u, v) \in R$, either $(u, v) = (\tilde{u}, \tilde{v})$, or $(u, v) = (\tilde{v}, \tilde{u})$. For (3), suppose that $x \Leftrightarrow y$, and $x \neq y$. Then $x$ must contain a factor $aa$ or $aba$. No word in the stated set has such a factor. For (4), the statement holds trivially when $n = 0$, so assume that $n > 0$. We apply $baa \Leftrightarrow aab$ to the leftmost $b$, and then do $aba \Leftrightarrow aa$, showing that $b^n aaa \Leftrightarrow b^{n-1}aaba \Leftrightarrow b^{n-1}aaa$. The conclusion follows by induction. □

We have everything we need to prove the main result.

**Theorem 1.** *The monoid $\langle a, b \mid aba \Leftrightarrow aa,\, baa \Leftrightarrow aab \rangle$ does not have a finite complete presentation over any alphabet.*

*Proof.* Assume that $\langle C \mid S \rangle$ is a finite complete presentation of $\langle A \mid R \rangle$. We use the lowercase Greek alphabet to denote words in $C^*$.

We need a little bit of notation for working with $\langle C \mid S \rangle$. We fix a "decoding" morphism $\Psi$ from $C^*$ to $A^*$, such that $\alpha \Leftrightarrow_S \beta$ if and only if $\Psi(\alpha) \Leftrightarrow \Psi(\beta)$:

$$\Psi \colon C^* \to A^*$$

We assume that no letter of $C$ decodes to the empty word in $A^*$, since any such letter can be removed from the presentation. Thus, $\Psi$ is non-erasing.

Finally, we define a function $\Lambda$ that takes a word in $A^*$, encodes it over $C^*$, and computes the $\Rightarrow_S$-normal form, which produces a word in $\textsc{Irr}(S) \subseteq C^*$:

$$\Lambda \colon A^* \to \textsc{Irr}(S)$$

So $\Lambda(x) = \Lambda(y)$ if and only if $x \Leftrightarrow y$, and $\Psi(\Lambda(x)) \Leftrightarrow x$, for all $x, y \in A^*$.

Since $S$ is finite, $\textsc{Irr}(S)$ is regular by Lemma 3. Let $\ell > 0$ be the smallest natural number such that Lemma 4 applies to any word $\lambda \in \textsc{Irr}(S)$ with $|\lambda| \geq \ell$.

The basic idea is that we apply the pumping lemma to $\Lambda(aab^n)$, and one of $\Lambda(b^m a)$ or $\Lambda(ab^m)$; we then "connect" them in a certain way, to get a sequence of distinct irreducible words that all decode to a word equivalent to $aaa$.

First, consider the normal form of $aab^n$, where $n \geq 0$. By part 1 of Lemma 7, there is an $0 \leq i \leq n$ such that:

$$\Psi(\Lambda(aab^n)) = b^i aab^{n-i} \quad \text{or} \quad \Psi(\Lambda(aab^n)) = b^i abab^{n-i}$$

Now, we can choose a large $n > 0$ such that $\Lambda(aab^n)$ has a prefix or suffix of length at least $\ell$ that decodes to a word in $b^+$ by $\Phi$. We then write:

$$\Lambda(aab^n) = \alpha\beta \quad \text{or} \quad \Lambda(aab^n) = \beta\alpha$$

where $\Psi(\alpha) \in b^*aab^* \cup b^*abab^*$, $\Psi(\beta) \in b^+$, and $|\beta| \geq \ell$. We consider the first case where $\Lambda(aab^n) = \alpha\beta$; the other direction is symmetric, by part 2 of Lemma 7. We apply the pumping lemma "at the end" of $\alpha\beta \in \textsc{Irr}(S)$. This yields a factorization:

$$\alpha\beta = \alpha\beta_0\beta_1\beta_2 \in \textsc{Irr}(S)$$

with the property that $|\beta_0| \geq 0$, $|\beta_1| > 0$, $|\beta_1\beta_2| \leq \ell$, and for all $k \geq 0$:

$$\alpha\beta_0\beta_1^k\beta_2 \in \textsc{Irr}(S)$$

Now, $\textsc{Irr}(S)$ is closed under taking prefixes by Lemma 3, so we can drop $\beta_2$:

$$\alpha\beta_0\beta_1^k \in \textsc{Irr}(S) \tag{1}$$

Also, note that $\Psi(\beta_0) \in b^*$, $\Psi(\beta_1) \in b^+$.

Next, consider the normal form of $b^m a$, where $m \geq 0$. (If $\Lambda(aab^n) = \beta\alpha$ above, we would be looking at $ab^m$ instead.) This is a singleton equivalence class of $\Leftrightarrow$, by part 3 of Lemma 7, so $\Psi(\Lambda(b^m a))$ is identically equal to $b^m a$. We choose $m$ large enough so that $|\Lambda(b^m a)| \geq \ell + 1$. Now, since $\Psi$ is non-erasing, we can write:

$$\Lambda(b^m a) = \delta\gamma$$

where $\Psi(\delta) \in b^+$, $\Psi(\gamma) \in b^* a$, and $\gamma \in C$ is a single letter, thus $|\delta| \geq \ell$. We apply the pumping lemma again, this time "at the start" of $\delta\gamma$. This yields a factorization:

$$\delta\gamma = \delta_2\delta_1\delta_0\gamma \in \textsc{Irr}(S)$$

with the property that $|\delta_1| > 0$, $|\delta_2\delta_1| \leq \ell$, and for all $k \geq 0$:

$$\delta_2\delta_1^k\delta_0\gamma \in \textsc{Irr}(S)$$

Now, $\textsc{Irr}(S)$ is closed under taking suffixes, so again we drop $\delta_2$:

$$\delta_1^k\delta_0\gamma \in \textsc{Irr}(S) \tag{2}$$

Also, $\Psi(\delta_0) \in b^*$, $\Psi(\delta_1) \in b^+$.

All that remains is to connect the two pumps. There exist $i$, $j > 0$ such that:

$$\Psi(\beta_1) = b^i$$
$$\Psi(\delta_1) = b^j$$

Thus $\Psi(\beta_1^j) = \Psi(\delta_1^i)$, so $\beta_1^j \Leftrightarrow_S \delta_1^i$, but both are irreducible, so $\beta_1^j = \delta_1^i$. We set:

$$\zeta := \beta_1^j = \delta_1^i \tag{3}$$

Taking (1), (2), and (3) together, we see that for all $k \geq 0$:

$$\alpha\beta_0\zeta^k \in \textsc{Irr}(S)$$
$$\zeta^k\delta_0\gamma \in \textsc{Irr}(S)$$

Furthermore, $S$ is finite, so by Lemma 6, for all sufficiently large $k > 0$:

$$\alpha\beta_0\zeta^k\delta_0\gamma \in \textsc{Irr}(S) \tag{4}$$

We're almost done. Let's apply $\Psi$ to each factor of the above word. We recall that:

$$\begin{aligned}
&\Psi(\alpha) \in b^*aab^* \cup b^*abab^* \\
&\Psi(\beta_0) \in b^* \\
&\Psi(\zeta^k) \in b^+ \\
&\Psi(\delta_0) \in b^* \\
&\Psi(\gamma) \in b^*a
\end{aligned}$$

We concatenate our regular sets, to form the statement:

$$\Psi(\alpha\beta_0\zeta^k\delta_0\gamma) \in b^*aab^+a \cup b^*abab^+a$$

This is to say, for each $k > 0$, there exist $i \geq 0$, $j > 0$ such that:

$$\Psi(\alpha\beta_0\zeta^k\delta_0\gamma) = b^iaab^ja$$

or

$$\Psi(\alpha\beta_0\zeta^k\delta_0\gamma) = b^iabab^ja$$

We have yet to use part 4 of Lemma 7, and this is exactly where we need it:

$$b^iabab^ja \Leftrightarrow b^iaab^ja \Leftrightarrow b^{i+j}aaa \Leftrightarrow aaa$$

So in fact:

$$\Psi(\alpha\beta_0\zeta^k\delta_0\gamma) \Leftrightarrow aaa \tag{5}$$

We have our contradiction, because (4) and (5) cannot both be true. □

## References


[1] E. L. Post, "Recursive unsolvability of a problem of Thue," *The Journal of Symbolic Logic*, vol. 12, no. 1, p. 1–11, 1947. https://www.wolframscience.com/prizes/tm23/images/Post2.pdf

[2] F. Otto and Y. Kobayashi, "Properties of monoids that are presented by finite convergent string-rewriting systems — a survey," in *Advances in Algorithms, Languages, and Complexity*. Springer, 1997, pp. 225–266. https://static.aminer.org/pdf/PDF/000/066/293/properties_of_monoids_that_are_presented_by_finite_convergent_string.pdf

[3] C. C. Sims, *Computation with Finitely Presented Groups*, ser. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 1994. https://www.cambridge.org/us/universitypress/subjects/mathematics/algebra/computation-finitely-presented-groups

[4] D. E. Knuth and P. B. Bendix, "Simple word problems in universal algebras," in *Automation of Reasoning: 2: Classical Papers on Computational Logic 1967–1970*. Springer, 1983, pp. 342–376. https://www.semanticscholar.org/paper/Simple-Word-Problems-in-Universal-Algebras-Knuth-Bendix/94877bdf8313565b90758a5e664764139857b358

[5] G. Huet, "A complete proof of correctness of the Knuth-Bendix completion algorithm," *Journal of Computer and System Sciences*, vol. 23, no. 1, pp. 11–21, 1981. https://www.sciencedirect.com/science/article/pii/0022000081900027

[6] R. V. Book, "Thue systems as rewriting systems," *Journal of Symbolic Computation*, vol. 3, no. 1, pp. 39–68, 1987. https://www.sciencedirect.com/science/article/pii/S0747717187800214

[7] D. Kapur and P. Narendran, "A finite Thue system with decidable word problem and without equivalent finite canonical system," *Theoretical Computer Science*, vol. 35, pp. 337–344, 1985. https://www.sciencedirect.com/science/article/pii/0304397585900234

[8] F. Otto, "Finite complete rewriting systems for the Jantzen monoid and the Greendlinger group," *Theoretical Computer Science*, vol. 32, no. 3, pp. 249–260, 1984. https://www.sciencedirect.com/science/article/pii/0304397584900446

[9] G. S. Tseitin, "An associative calculus with an insoluble problem of equivalence," in *Problems of the constructive direction in mathematics. Part 1.* Moscow–Leningrad: Acad. Sci. USSR, 1958, vol. 52, pp. 172–189. https://www.mathnet.ru/php/archive.phtml?wshow=paper&jrnid=tm&paperid=1317&option_lang=eng

[10] C.-F. Nyberg-Brodda, "G. S. Tseytin's seven-relation semigroup with undecidable word problem," 2024. https://arxiv.org/abs/2401.11757

[11] C. C. Squier, "Word problems and a homological finiteness condition for monoids," *Journal of Pure and Applied Algebra*, vol. 49, no. 1, pp. 201–217, 1987. https://www.sciencedirect.com/science/article/pii/0022404987901290

[12] C. C. Squier, F. Otto, and Y. Kobayashi, "A finiteness condition for rewriting systems," *Theoretical Computer Science*, vol. 131, no. 2, pp. 271–294, 1994. https://www.sciencedirect.com/science/article/pii/0304397594901759

[13] R. Cremanns and F. Otto, "Finite derivation type implies the homological finiteness condition FP3," *Journal of Symbolic Computation*, vol. 18, no. 2, pp. 91–112, 1994. https://www.sciencedirect.com/science/article/pii/S074771718471039X

[14] M. Katsura and Y. Kobayashi, "Constructing finitely presented monoids which have no finite complete presentation," *Semigroup Forum*, vol. 54, pp. 292–302, 1997. https://link.springer.com/article/10.1007/BF02676612

[15] A. J. Cain, R. D. Gray, and A. Malheiro, "On finite complete rewriting systems, finite derivation type, and automaticity for homogeneous monoids," *Information and Computation*, vol. 255, pp. 68–93, 2017. https://www.sciencedirect.com/science/article/pii/S0890540117300937

[16] C.-F. Nyberg-Brodda, "The word problem for one-relation monoids: a survey," *Semigroup Forum*, vol. 103, no. 2, pp. 297–355, 2021. https://link.springer.com/article/10.1007/s00233-021-10216-8

[17] Y. Kobayashi, "Finite homotopy bases of one-relator monoids," *Journal of Algebra*, vol. 229, no. 2, pp. 547–569, 2000. https://www.sciencedirect.com/science/article/pii/S0021869399982510

[18] R. Book and F. Otto, *String-Rewriting Systems*, ser. Monographs in Computer Science. Springer, 2012. https://link.springer.com/book/10.1007/978-1-4613-9771-7

[19] J. Berstel, *Transductions and Context-Free Languages*. Teubner Verlag, 1979. https://www-igm.univ-mlv.fr/~berstel/LivreTransductions/LivreTransductions.html