Instances of Undecidability in the Semigroup Word Problem

Timothy C. Grosky

University of New Hampshire

Follow this and additional works at: https://scholars.unh.edu/honors

Part of the Algebra Commons

Recommended Citation
Grosky, Timothy C., "Instances of Undecidability in the Semigroup Word Problem" (2024). Honors Theses and Capstones. 855.
https://scholars.unh.edu/honors/855

This Senior Honors Thesis is brought to you for free and open access by the Student Scholarship at University of New Hampshire Scholars' Repository. It has been accepted for inclusion in Honors Theses and Capstones by an authorized administrator of University of New Hampshire Scholars' Repository. For more information, please contact Scholarly.Communication@unh.edu.
INSTANCES OF UNDECIDABILITY IN THE SEMIGROUP WORD PROBLEM

TIMOTHY C. GROSKY

Abstract. We will examine the decidability of the word problem in semigroups, which is a yes/no question. We will examine tools that have been developed to help answer it, and then look at some examples where the word problem is decidable or undecidable.

We must first determine what we mean by a problem being called undecidable. We can answer this question by looking at the following quote from page 1 of a research paper [8] by Poonen.

“A family of problems with YES/NO answers is called undecidable if there is no algorithm that terminates with the correct answer for every problem in the family.”

We will examine the question of decidability in the word problem for semigroups, which asks whether we can always tell if two words are equivalent, given that the words are sequences of letters (or generators) and that some sequences can be substituted for others.

1. Literature Review

We have already seen, from a research paper [8] by Poonen, what it means for problems to be undecidable.

We have referenced research papers of Otto and Squier [7], Matiyasevich [5], Lallement [4], Nyberg-Brodda [6], and Collins [2] as examples of semigroups and monoids where the word problem is decidable or undecidable.

We will obtain a result from a research paper [1] by Bauer and Otto about finite complete rewriting systems, which is that the word problem is decidable.

In the next section, we will use a book [9] by Sims, which will provide much of the necessary mathematical background for the word problem. We will also use pages 1-25 of conference notes [3] by Holt, which are more advanced, but nevertheless useful.

Date: May 19, 2024.
2. Background

Let $S$ be a finite set; its elements will be called letters or generators. Define $S^*$ to be the set of all finite sequences (or "strings" or "words") of elements in $S$. We place an operation on $S^*$ in which two strings $\alpha$ and $\beta$ are joined into a single string $\alpha\beta$; this operation is called "concatenation.” The concatenation operation makes $S^*$ into a "semigroup.” Moreover, if we designate the "null string,” i.e., the string $\epsilon$ with no letters, as an element of $S^*$, then $\epsilon$ is an identity element for concatenation, and $S^*$ becomes a semigroup with identity element, called a ”monoid.”

Let $X$ and $Y$ be sets. We have obtained the following definitions from Sims’ book, ([9], pp. 6-8).

**Definition 2.1.** We call the set $X \times Y = \{(x, y)|x \in X, y \in Y\}$ the **cartesian product** of $X$ and $Y$.

**Definition 2.2.** Let $X$ be a finite set. We will denote the **cardinality** of $X$ by $|X|$.

**Definition 2.3.** Let $X$ and $Y$ be sets. We call a subset $R$ of $X \times Y$ a **relation** from $X$ to $Y$. If $X = Y$, we call $R$ a **relation on** $X$.

We now will define some orderings. Let $S$ be a finite set. An ordering on $S^*$ is a relation such that for every pair of distinct words in $S^*$, we are able to call one word ”simpler” than the other. We have obtained these definitions from Sims ([9], pp. 43-51.)

**Definition 2.4.** Let $S$ be a set. Now let $<$ be a relation on $S$ such that for all $a, b \in S$, either $a < b, a = b$, or $b < a$, and for all $a, b, c \in S$, $a < b$ and $b < c$ implies $a < c$. We call $<$ a **linear ordering** on $S$.

Note that for all $a, b \in S$, we will write $a \leq b$ if $a < b$ or $a = b$. Also, we will write $b > a$ and $b \geq a$ if $a < b$ or $a \leq b$, respectively.
Definition 2.5. Let < be a linear ordering on a set $S$. We call < a well-ordering if there are no infinite sequences $s_1, s_2, ..., s_i \in S$, such that for all $i \geq 1$, $s_i > s_{i+1}$.

Definition 2.6. Next, let $A$ be a finite set and let < be an ordering of $A^*$. We call < translation invariant if for all $a, b, u, v \in A^*$, $u < v$ implies $aub < avb$.

Definition 2.7. Let $S$ be a set and $n \in \mathbb{N}$, where $S$ has a linear ordering <. Now we’ll define the left-to-right lexicographic ordering on $S^*$. Let $a_1a_2...a_n < b_1b_2...b_n$ mean that there exists an $i \in \mathbb{N}$ with $1 \leq i \leq n$ and $a_k = b_k$ for all $1 \leq k < i$ and $a_i < b_i$, where all $a_j, b_j \in S$. There also exists a right-to-left lexicographic ordering, but from now on, we will refer to the left-to-right lexicographic ordering as the lexicographic ordering, and not be concerned with this other ordering.

3. Methods

The next two definitions, from Holt [3], tell us what rewrite rules and rewrite systems are.

Definition 3.1. Let $A$ be a finite set, and let $u_1, u_2 \in A^*$. Also, let < be a translation invariant well-ordering. By Sims [9], we require $u_1 > u_2$. We say the ordered pair $(u_1, u_2)$ is a rewrite rule. We call $u_1$ and $u_2$ the left- and right-hand sides of the rewrite rule, respectively. Now whenever we see $u_1$ within a string, we are allowed to remove it and insert $u_2$ into the same location within the string, and say that the strings before and after this process are equivalent.

Definition 3.2. Let $A$ be a finite set. Next, a rewriting system on $A^*$ is a set $S$ consisting of rewrite rules, where every distinct rewrite rule has a distinct left-hand side.

The following definitions describe various behaviors of a rewrite system with a given translation invariant well ordering. We have obtained these definitions from Holt [3].

Now let $A$ be a finite set, and let $S$ be a rewrite system for a given translation invariant well-ordering of $A^*$. 
Let $u, v \in A^*$. Now we will write $u \rightarrow v$ if $u = axb, v = ayb$, and $(x, y) \in S$, for some $a, b, x, y \in A^*$.

Let $u, v \in A^*$. We will write $u \rightarrow^* v$ if there exists $u = u_0 \rightarrow u_1 \rightarrow u_2 \rightarrow ... \rightarrow u_n = v$, with $n \geq 0$ and $u_i \rightarrow u_{i+1}$ for all $i \in [0, n)$.

**Definition 3.3.** We call $u \in A^*$ **irreducible** if there exists no $v \in A^*$ such that $u \rightarrow v$.

**Definition 3.4.** We call $S$ **Noetherian** if there is no infinite chain of strings $u_i \rightarrow u_{i+1}$ for all $i \in \mathbb{N}$, which is the same as for all $u \in A^*$, there exists an irreducible $v \in A^*$ such that $u \rightarrow^* v$.

**Definition 3.5.** $S$ is **confluent** if for all $u, v_1, v_2 \in A^*$ where $u \rightarrow^* v_1$ and $u \rightarrow^* v_2$, there exists $w \in A^*$ such that $v_1 \rightarrow^* w$ and $v_2 \rightarrow^* w$.

**Definition 3.6.** Similarly, we call $S$ **locally confluent** if for all $u, v_1, v_2 \in A^*$ where $u \rightarrow v_1$ and $u \rightarrow v_2$, there exists $w \in A^*$ such that $v_1 \rightarrow^* w$ and $v_2 \rightarrow^* w$.

**Definition 3.7.** $S$ is called **complete** if it is both Noetherian and confluent.

**Definition 3.8.** Let $A$ be a finite set. Now let $u \in A^*$. We say $\text{desc}(u) = \{v \in A^* - \{u\} | u \rightarrow^* v\}$ Note that $\text{desc}(u) = \emptyset$ if and only if $u$ is irreducible.

The following contains some lemmas that will help us to determine a sufficient condition for decidability.

**Lemma 3.9.** ([3], Lemma 2.1) If $S$ is Noetherian, then for all $u \in A^*$, $\text{desc}(u)$ is finite.

**Proof:** Assume, to the contrary, that $\text{desc}(u)$ is infinite. Also assume that $S$ is Noetherian.

Since every rewrite rule has a different left-hand side, and since for any $u$ there are only finitely many substrings of $u$, then there can only be finitely many $w \in A^*$ such that $u \rightarrow w$. 


For \( u \) to have infinitely many descendants, one of these \( u \) must have infinitely many descendants, and we shall call this \( w \ u_2 \).

Now the same logic applies to \( u_2 \), so there must be a \( u_2 \in A^* \) such that \( u_2 \to u_3 \) and \( \text{desc}(u_3) \) is infinite.

Therefore, we get an infinite chain \( \{u_i\} \) with \( u_i \to u_{i+1} \). This contradicts the assumption that \( S \) is Noetherian.

Hence, \( \text{desc}(u) \) is finite. □

Lemma 3.10. ([3], Lemma 2.2) If \( S \) is Noetherian and locally confluent, then \( S \) is complete.

Proof: Suppose that \( S \) is Noetherian and locally confluent. We shall prove that for all \( u \in A^* \), there is a unique irreducible \( w \) with \( u \to^* w \). This would mean that \( S \) is confluent, since if \( u, v_1, v_2 \in A^* \), \( u \to^* v_1 \), and \( u \to^* v_2 \), then \( v_1 \to^* w \) and \( v_2 \to^* w \). We do need the Noetherian property to guarantee that we can stop reducing. We will prove the result by induction on \( |\text{desc}(u)| \). First, note that if \( |\text{desc}(u)| = 0 \), i.e. \( u \) is irreducible, then the result is vacuously true. Second, suppose that \( u = u_0 \to u_1 \to ... \to u_m \) and \( u = u_0 \to u'_1 \to ... \to u'_n \), with \( u_m \) and \( u'_n \) irreducible. By local confluence, since \( u_0 \to u_1 \) and \( u_0 \to u'_1 \), there exists \( v \) with \( u_1 \to^* v \) and \( u'_1 \to^* v \). Let \( v \to^* w \) with \( w \) irreducible. Therefore, \( u_1 \to^* w \) and \( u'_1 \to^* w \). Then, since \( |\text{desc}(u_1)| < |\text{desc}(u)| \) and \( |\text{desc}(u'_1)| < |\text{desc}(u)| \), by the inductive hypothesis \( u_m = w \) and \( u'_n = w \). Thus, \( u_m = u'_n \). Now \( S \) is confluent. Hence, \( S \) is complete, since it is confluent and Noetherian. □

Lemma 3.11. ([3], Lemma 2.4)

Let condition (i) be: if \( u_1 = rs \) and \( u_2 = st \) with \( r, s, t \in A^* \) and \( s \neq \epsilon \), then there exists \( w \in A^* \) with \( t_1 t \to^* w \) and \( rt_2 \to^* w \).

Let condition (ii) be: if \( u_1 = rst \) and \( u_2 = s \) with \( r, s, t \in A^* \) and \( s \neq \epsilon \), then there exists \( w \in A^* \) with \( t_1 \to^* w \) and \( rt_2 t \to^* w \).

\( S \) is locally confluent if and only if conditions (i) and (ii) are satisfied for all pairs of rules \( (u_1, t_1), (u_2, t_2) \in S \).

Proof: ( \( \implies \) ): Suppose that \( S \) is locally confluent.

Let \( (u_1, t_1), (u_2, t_2) \in S \) be rules.
We will consider the following cases.

Case 1. Assume $u_1 = rs$ and $u_2 = st$ with $r, s, t \in A^*$ and $s \neq \epsilon$.

Now the pair of rules is $(rs, t_1)$ and $(st, t_2)$.

So $rst \rightarrow t_1 t$ and $rst \rightarrow rt_2$.

Then by the definition of local confluence, there exists $w \in A^*$ with $t_1 t \rightarrow^* w$ and $rt_2 \rightarrow^* w$.

Thus, (i) is satisfied.

Case 2. Assume $u_1 = rst$ and $u_2 = s$ with $r, s, t \in A^*$ and $s \neq \epsilon$.

Now the pair of rules is $(rst, t_1)$ and $(s, t_2)$.

So $rst \rightarrow t_1$ and $rst \rightarrow rt_2 t$.

Then by the definition of local confluence, there exists $w \in A^*$ with $t_1 t \rightarrow^* w$ and $rt_2 t \rightarrow^* w$.

Thus, (ii) is satisfied.

Hence, both (i) and (ii) are satisfied.

( $\iff$ ): Assume that (i) and (ii) are satisfied for all pairs of rules $(u_1, t_1), (u_2, t_2) \in S$. Let $u$ be a string with $u \rightarrow v_1$ and $u \rightarrow v_2$.

Then $u$ must have two substrings $u_1$ and $u_2$ with rules $(u_1, t_1), (u_2, t_2) \in S$, because otherwise there would be no way to separately reduce to $v_1$ and $v_2$.

This is because each rule has a unique left hand side.

We will consider the following cases.

Case 1. Assume $u_1$ and $u_2$ do not overlap in $u$.

So $u = ru_1su_2t$ for some strings $r, s, t$.

Then let $v_1 = rt_1su_2t$ and $v_2 = ru_1st_2t$, so that $u \rightarrow v_1$ and $u \rightarrow v_2$.

Let $w = rt_1st_2t$.

Now $v_1 \rightarrow w$ and $v_2 \rightarrow w$.

Case 2. Assume $u_1$ and $u_2$ overlap in $u$, i.e. they overlap in $S$.

We may interchange the rules if necessary.

Now we have one of the situations in (i) or (ii).

Note that in (i), the end of $u_1$ overlaps with the beginning of $u_2$, whereas in (ii), all of $u_2$ is contained within $u_1$.

Now by the assumption that (i) and (ii) are valid, it follows that there exists $w \in A^*$ with $v_1 \rightarrow^* w$ and $v_2 \rightarrow^* w$. 
So $S$ is locally confluent.

Hence, $S$ is locally confluent if and only if (i) and (ii) are satisfied for all pairs of rules $(u_1, t_1), (u_2, t_2) \in S$. □

**Theorem 3.12.** ([1], Proposition 2.1)

The word problem for a finite complete rewriting system is decidable.

This is because when we have finite rewriting rules and the rewriting system is complete, we are able to compute the simplest word, according to the ordering of the monoid or semigroup, which is equivalent to the given word.

This theorem is useful because it means that if you are able to reduce two words to their unique irreducible forms, then you are able to tell whether those words are equivalent, because the words are equivalent if and only if their irreducible forms are identical.

From Theorem 3.12, we know that the word problem for a finite complete rewriting system is decidable. Therefore, if we can show that a rewriting system is complete, i.e. it is Noetherian and confluent, and that it has a finite number of generators and relations, then we know that the word problem on that rewriting system is decidable.

Note that if a rewriting system has a finite number of generators and relations, then it is necessarily Noetherian. Since the sets we used to create a semigroup or monoid were finite, the rewriting systems we are considering all have a finite number of generators. Pick any $u \in A^\ast$. Now by Lemma 3.9, desc(u) is finite, so it is impossible to have an infinite chain. Thus, we can say that if a rewriting system is confluent and finite, then the word problem is decidable.

**Lemma 3.13.** If (i) or (ii) from Lemma 3.11 fails for a rewriting system $S$, then $S$ is not confluent.
Proof: We will prove the result contrapositively.
Assume $S$ is confluent.
Let $u, v_1, v_2 \in A^*$ with $u \rightarrow v_1$ and $u \rightarrow v_2$.
Now $u \rightarrow^* v_1$ and $u \rightarrow^* v_2$, as well.
By the definition of confluent, there exists $w \in A^*$ with $v_1 \rightarrow^* w$ and $v_2 \rightarrow^* w$. 
So $S$ is locally confluent.
By Lemma 3.11, (i) and (ii) are satisfied.
Hence, if (i) or (ii) fails, then $S$ is not confluent. □

4. Examples and Analysis

Example 4.1. ([7], Example 1)
We will now give an example of a rewriting system which does have decidable word problem, but does not have an equivalent finite and complete rewriting system. (Squier and Otto)
Let $A = \{a, b\}$ and $S = \{(bab, aba)\}

Now we will prove that this rewriting system has decidable word problem by considering whether each pair of rules satisfies conditions (i) and (ii) from Lemma 3.11. If all pairs of rules do, then we will be able to determine that $S$ is locally confluent, and from that, that it has a decidable word problem.

Let $r = ba$, $s = b$, and $t = ab$, and let $u_1 = rs = bab$ and $u_2 = st = bab$.
Now let $t_1 = aba$ and $t_2 = aba$, so that $(u_1, t_1)$, $(u_2, t_2) \in S$.
Then $t_1 t = abat = abaab$ and $rt_2 = raba = baaba$. Both of these are irreducible, since they do not have any $bab$ substrings. So add $(baaba, abaab)$ as a rule to $S$, since $abaab < baaba$.

Let $r = baa$, $s = ba$, and $t = b$, and let $u_1 = rs = baaba$ and $u_2 = st = bab$.
Now let $t_1 = abaab$ and $t_2 = aba$, so that $(u_1, t_1)$, $(u_2, t_2) \in S$.
Then $t_1 t = ababt = ababbb$ and $rt_2 = raba = baaaba$. Both of these are irreducible, since they do not have any $bab$ or $baaba$ substrings. So add $(baaba, abaabb)$ as a rule to $S$, since $abaabb < baaaba$. 


Now let \( r = ba^n, s = ba, \) and \( t = b, \) and let \( u_1 = rs = ba^nba \) and \( u_2 = st = bab. \)

Assume \((ba^nba, abaab^{n-1})\) is a rule in \( S.\)

Next, let \( t_1 = abaab^{n-1} \) and \( t_2 = abaab^{n-1}, \) so that \((u_1, t_1), (u_2, t_2) \in S.\)

Then \( t_1 t = abaab^{n-1}t = abaab^{n-1}b = abaab^n \) and \( rt_2 = raba = ba^naba = ba^{n+1}ba. \) Both of these are irreducible, since they do not have any of the previously added left-hand sides of rules or \( bab \) as substrings. So add \((ba^{n+1}ba, abaab^n)\) as a rule to \( S, \) since \( abaab^n < ba^{n+1}ba.\)

Now we will check to make sure we have seen all pairs.

Note that now \( S = \{(bab, aba)\} \cup \bigcup_{n=2}^{\infty} \{(ba^nba, abaab^{n-1})\}.\)

First, note that no left-hand side of a rule is contained as a substring of a left-hand side of another rule, so condition (ii) of Lemma 3.11 is satisfied for all pairs of rules. Thus, we will only consider condition (i) of Lemma 3.11.

Then note that all rules we have obtained so far are the result of comparing \((bab, aba)\) with either itself or another rule using condition (i). But for all cases involving \((bab, aba)\) besides comparing this rule with itself, there is one other way we can check these pairings where condition (i) applies.

Let \( r = ba, s = b, \) and \( t = a^nba, \) and let \( u_1 = rs = bab \) and \( u_2 = st = ba^nba, \) where \( n \in \mathbb{N} \) such that \( n \geq 2.\)

Let \( t_1 = abaab^{n-1} \) and \( t_2 = abaab^n, \) so that \((u_1, t_1), (u_2, t_2) \in S.\)

Then \( t_1 t = abaab^nba = abaab^{n+1}ba = abaab^n \) and \( rt_2 = rabaab^{n-1} = abaabab^{n-1} = abaabba^{n-1} = abaab^n. \) Since these are the same, condition (i) is satisfied here.

Next, we will consider pairing two other rules not including \((bab, aba).\) Let \( m, n \in \mathbb{N} \) such that \( m, n \geq 2. \) Assume, without loss of generality, that \( n \geq m. \) Now \((ba^nba, abaab^{n-1})\) and \((ba^mba, abaab^{m-1})\) are rules in \( S.\)

First, let \( r = ba^n, s = ba, \) and \( t = a^{m-1}ba, \) and let \( u_1 = rs = ba^nba \) and \( u_2 = st = ba^{m-1}ba = ba^mba. \)

Then let \( t_1 = abaab^{n-1} \) and \( t_2 = abaab^{m-1}, \) so that \((u_1, t_1), (u_2, t_2) \in S.\)

We will sometimes use parentheses as notation here to indicate what will be changed in the next step. Note that \( b^0 = \epsilon.\)

Then \( t_1 t = abaab^{n-1}t = abaab^{n-1}a^{m-1}ba \) and \( rt_2 = rabaab^{m-1} = b(a^n)aabaab^{m-1} = (ba^nba)ab^{m-1} = aaba(ab)^{m-1} = abaab^{n-1}(bab)b^{m-2} = abaab^{n-1}aba(b^{m-2}) = abaab^{n-1}a(bab)b^{m-3} = abaab^{n-1}(aa)ab^{m-3} \)
\[ abaab^{n-1}a^2ba(b^{m-3}) = \ldots = abaab^{n-1}a^{2+(m-3)}ba(b^0) = abaab^{n-1}a^{m-1}ba. \] Since these are the same, condition (i) is satisfied here.

Second, let \( r = ba^m, s = ba \), and \( t = a^{n-1}ba \). Define \( u_1 = rs \) and \( u_2 = st \), and then define \( t_1 \) and \( t_2 \) in such a way that \( (u_1, t_1), (u_2, t_2) \in S \).

This case is symmetrical to the first, and so the argument is nearly the same.

Thus, conditions (i) and (ii) are satisfied in all cases.

Therefore, the rewrite system \( S \) with this infinite set of rules is locally confluent by Lemma 3.11, since (i) and (ii) are satisfied for all pairs of rules.

Note that all rules in \( S \) replace one string of letters by another of equal length. We call this a homogeneous rewrite system. Therefore, since \( A \) is well-ordered, and each word has only finitely many letters, we can only make finitely many substitutions before we reach some irreducible word which comes before all others that it is equivalent to in the well-ordering. So \( S \) is Noetherian.

Then by Lemma 3.10, since \( S \) is Noetherian and locally confluent, \( S \) is complete.

Finally, by Theorem 3.12, the rewriting system has decidable word problem, since it is finite and complete.

Now there does not exist a finite complete rewriting system that is equivalent to \( S \) but which also has decidable word problem.

**Example 4.2.** ([5], Page 42)

Let \( A = \{ a, b, c, d, e \} \).

Let \( S = \{ ac = ca, ad = da, bc = cb, bd = db, eca = ce, edb = de, cdca = cdcae, caaa = aaa, daaa = aaa \} \)

"Tseytin ... proved that [there] is no algorithm to decide, given a word \( H \), whether it is equivalent to word \( aaa \) or not."

Thus, the rewriting system has undecidable word problem.

**Example 4.3.** ([4], Example 2a)
Let $A = \{a, b\}$.
Let $S = \{ab = \epsilon\}$.

"Each word in $A^*$ can be transformed into a word of the form $b^m a^n$ ($m \geq 0, n \geq 0$)."

"The word problem is decidable, essentially because each word $b^m a^n$ is the unique word of shortest length in its equivalence class."

Note that an equivalence class is the set of all words that are equivalent to a word with respect to the given relations.

We can demonstrate this transformation with an example, by letting some word $P = a^i b^j a^k \in A^*$, with $i, j, k \in \mathbb{N}$. If $i \leq j$, then $P = a^i b^j b^{-i} a^k = a^{i-1} ab b^{-i} a^k = a^{i-1} b^{j-i} a^k = \ldots = b^{-i} a^k$.

**Example 4.4. ([6], Page 1)**

Let $A = \{a, b, c, d, e\}$
Let $S = \{ac = ca, ad = da, bc = cb, bd = db, eca = ce, edb = de, cca = ccae\}$.

This semigroup observed by Tseytin has undecidable word problem.

Notice that there is commutativity between some pairs of generators, such as $a$ and $c$, but not all pairs. Then, there is almost commutativity between pairs like $e$ and $c$, but it requires the removal of an $a$ after it as well. And $cca$ can be replaced by itself with an extra $e$ at the end.

**Example 4.5. ([5], Page 43)**

Let $A = \{a, b\}$. Let $S = \{aabab = baa, aabb = baa, L = M\}$, where $L$ and $M$ are specific words with 304 and 608 letters, respectively.

The word problem is undecidable.

The author compares this example to those given by Tseytin, and notes that “the cost paid for this reduction of the number of relations was the length of the relations.”

The following example is an example of the word problem in groups. We will give some more necessary mathematical background here, from pages 8-15 of the book [9] by Sims.

Let $S$ be a finite set, let $S^*$ be a monoid with identity element $\epsilon$, and let $x \in S$.

**Definition 4.6.** We define an inverse of $x$ to be $x^{-1} \in S$ such that $xx^{-1} = x^{-1}x = \epsilon$. 
Definition 4.7. We call $S^*$ a group if every $x \in S$ has an inverse in $S$.

Example 4.8. ([2], Page 232)

Let $A = \{a, b, c, d, e, p, q, r, s, t, k\}$. Let $S = \{p^{10}a = ap, p^{10}b = bp, p^{10}c = cp, p^{10}d = dp, p^{10}e = ep, qa = aq^{10}, qb = bq^{10}, qc = cq^{10}, qd = dq^{10}, qe = eq^{10}, ra = ar, rb = br, rc = cr, rd = dr, re = er, pacqr = rpcaq, p^2adq^2r = rp^2daq^2, p^3bcq^3r = rp^3cbq^3, p^4bdq^4r = rp^4dbq^4, p^5ceq^5r = rp^5ecaq^5, p^6deq^6r = rp^6edq^6, p^7edcq^7r = p^7cdceq^7, p^8caaq^8r = rp^8aaaq^8, p^9daaq^9r = rp^9aaaq^9, pt = tp, qt = tq, k(aaa)^{-1}t(aaa) = k(aaa)^{-1}t(aaa)\}$

Then the word problem is undecidable.

Borisov created this simplified group presentation from Boone making a construction based on Tseytin’s presentation given in Example 4.2.

Analysis

It is in the nature of undecidable word problems to be avoidant of spottable patterns, so much of the reason a word problem is undecidable is hidden within seemingly arbitrary exponents. Nevertheless, some glimmers of patterns can be spotted.

Notice that Example 4.8 has many rules that function as allowing for commutativity between two generators, such as $ra = ar$, but that this isn’t the case for all possible pairs of generators. Notice that in the case of rules like $p^{10}a = ap$ or $qd = dq^{10}$, this functions like commutativity in that the generators are swapped, but that it also adds or removes 9 $p$s or $q$s. Then, a lot of the other rules seem to have been constructed using the rules from Example 4.2. For example, $p^3bcq^3r = rp^3cbq3$ in Example 4.8 uses the rule $bc = cb$ from Example 4.2, except that it switches the $r$ from the end to the beginning, as well. Notice that the highest power of $p$ or $q$ in these types of rules is 9, one less than the 10 given in the earlier type of rules.

References


