Satisfiability of word equations with constants is in $NEXPTIME$

Wojciech Plandowski
Institute of Informatics, Warsaw University,
Banacha 2, 02-097, Warsaw, Poland.
E-mail: wojtekpl@mimuw.edu.pl.

Abstract

We prove that the length of a shortest solution of a word equation of length n can bounded by a double exponential function in n. This applied to the algorithm in [7] proves that the problem of solvability of word equations is in $NEXPTIME$. The best previously known bound for the problem was $EXPSPACE$ [3].

1 Introduction

A satisfiability problem for word equations has a simple formulation: find out whether an input word equation has a solution. The decidability of the problem was proved by Makanin [6]. His decision procedure is one of the most complicated algorithms existing in the literature. The first attempts to determine the complexity of Makanin's algorithm are due to Jaffar and Schultz [2, 8]. Their estimations placed the problem in 4-$NEXPTIME$ (composition of four exponential functions). The algorithm depends on a parameter called index of periodicity (see Section 2). The original estimation of the parameter was double exponential which was improved to single exponential by Koscielski and Pacholski [5]. This improvement allowed to place the problem in 3-$NEXPTIME$. Next, Diekert [1] noticed that the algorithm needs only double exponential space. Quite recently Gutierrez proved that the problem is in $EXPSPACE$ [3] by improving the analysis of the algorithm.

Another approach to the solution of the problem was proposed by Plandowski and Rytter [7]. They proved that minimal solutions of word equations are well compressible and they proposed an algorithm which works nondeterministically in time polynomial in $n \log(N)$ where n is the size of an input equation and N is the size of a minimal solution of the input equation. The analysis of the version of Makanin's algorithm by Gutierrez [3] gives triple exponential upper bound for N (Corollary 1 in [3]). With this estimation for N the algorithm from [7] is less efficient than the one from [3]. Moreover, using this estimation we still need Makanin's algorithm. No other upper bounds for N are stated in the literature. In particular there is no upper bound for N the proof of which does not use Makanin's algorithm.

We prove double exponential upper bound for N which allows, using the algorithm in [7], to place our problem in $NEXPTIME$. Our approach is independent from Makanin's algorithm. It uses the ideas which appear in [7] and [4]. It is based on properties of special factorizations of words called here $\mathcal{D}$-factorizations. $\mathcal{D}$-factorizations were invented by Dr. Filippo Mignosi (University of Palermo, Italy).

STOC '99 Atlanta GA USA

2 Preliminaries

The length of a word w is denoted by $|w|$. A subword of w starting at position i and ending at position j is denoted by $w[i..j]$. A period of a word is a number p such that for all i, $w[i] = w[i+p]$ whenever both sides of the equation are defined. A fundamental result dealing with periods is the following.

Proposition 1 (Fine and Wilf) *Let p, q be two periods of a word w. If $p+q \leq |w|$ then $gcd(p,q)$ is also a period of w, where gcd stands for the greatest common divisor.*

First we prove an auxiliary lemma.

Lemma 2 *Let $i < j < k$ be three consecutive starting positions of occurrences of a word v in w. If $i + |v| \geq k$ then $k-j = j-i$ and $k-j$ is a period of a word $w[i..k+|v|-1]$.*

Proof: Since $i + |v| \geq k$, the occurrences of v at positions i and j overlap. Hence, $j - i$ is a period of v. Similarly, $k - j$ is a period of v. By $i + |v| \geq k$ we have $(k - j) + (j - i) = k - i \leq |v|$. By Proposition 1 $gcd(j - i, k - j)$ is also a period of v. Since there is no occurrence of v between positions i and j, $j - i = gcd(j - i, k - j)$. Similarly we get $k - j = gcd(j - i, k - j)$. Hence, $j - i = k - j$. Let $p = k - j$. Then, since v occur at positions i and j in w, we have $w[s] = w[s + p]$, for $i \leq s < j$. Similarly, since v occur at position j and k in w, we have $w[s] = w[s + p]$, for $j \leq s \leq k + |v| - 1 - p$. Hence $p = k - j$ is a period of $w[i..k + |v| - 1]$. □

Let Σ and Ξ be two disjoint alphabets: alphabet of constants and alphabet of variables. A *word equation* is a pair of words (u, v) (usually denoted by $u = v$) over the alphabet $\Sigma \cup \Xi$. A *length* of a word equation $e : u = v$ is defined by $|u| + |v|$ and denoted by $|e|$. A *solution* of the word equation is a morphism $h : (\Xi \cup \Sigma)^* \to \Sigma^*$ such that $h(a) = a$, for $a \in \Sigma$, and $h(u) = h(v)$. Note that a morphism being a solution of a word equation is uniquely determined by its

values on variables. A solution h of $u = v$ is *minimal* if for all solutions g of $u = v$, $|h(u)| \leq |g(u)|$. There can be several minimal solutions of an equation.

An *index of periodicity* of a word w is a maximal number $p(w)$ such that $u^{p(w)}$, for a nonempty word u, is a subword of w. An *index of periodicity* $p(h)$ *of a solution* h of $u = v$ is $p(h(u))$.

Proposition 3 ([5]) *There is a constant c such that for each minimal solution h of a word equation e, $p(h) \leq 2^{c|e|}$.*

The original proof of Proposiotion 3 uses slightly different definition of a minimal solution and of an index of periodicity but the same proof works also with our definitions.

Denote by $\mathrm{card}(S)$ a cardinality of a set S.

3 $\mathcal{D}$-factorizations

A *factorization of a word* w is a sequence of nonempty words $\mathcal{P} = w_1, w_2, \ldots, w_k$ such that $w = w_1 w_2 \ldots w_k$. The words w_i are called *factors* of $\mathcal{P}$. The number k is the *length* of $\mathcal{P}$ and is denoted by $\mathrm{length}(\mathcal{P})$. In our considerations a *factorization* $\mathcal{F}$ is a function which takes a word and returns a factorization of this word. Denote by $\mathrm{first}_{\mathcal{F}}(w)$ the first factor of $\mathcal{F}(w)$. Similarly, denote by $\mathrm{last}_{\mathcal{F}}(w)$ the last factor of $\mathcal{F}(w)$. Denote by $\ker_{\mathcal{F}}(w)$ a subsequence of $\mathcal{F}(w)$ which is obtained from $\mathcal{F}(w)$ by removing the first and the last factors of $\mathcal{F}(w)$. If $\mathcal{F}(w)$ consists of at most two factors, then $\ker_{\mathcal{F}}(w)$ is undefined. Denote by $\mathcal{F}(w)[i,j]$ a sequence of words which is obtained from $\mathcal{F}(w)$ by cutting off the part of it which corresponds to $w[i..j]$. More precisely, let $\mathcal{F}(w) = w_1, \ldots, w_k$ and let $i_t = |w_1 w_2 \ldots w_{t-1}| + 1$ for $1 \leq t \leq k+1$ (i_t, for $t < k+1$, is a starting position of an occurrence of the word w_t in w). Let s, r be such that $i_r \leq i < i_{r+1}$ and $i_s \leq j < i_{s+1}$. If $s = r$ then $\mathcal{F}(w)[i,j] = w[i..j]$. Otherwise, $\mathcal{F}(w)[i,j] = w', w_{r+1}, w_{r+2}, \ldots, w_{s-1}, w''$ where w' is a suffix of w_r of length $i_{r+1} - i$ and w'' is a prefix of w_s of length $j - i_s + 1$. We say that the subsequence $w_r, w_{r+1}, \ldots, w_s$ of $\mathcal{F}(w)$ *corresponds* to the interval $[i,j]$ in $\mathcal{F}(w)$.

Let $\mathcal{D}$ be a set of words of the same length. *A $\mathcal{D}$-factorization* is a factorization which is defined as follows. If no word of $\mathcal{D}$ occurs in a word w, then $\mathcal{D}(w) = w$. Otherwise, let $i_1 < i_2 < \ldots < i_k$ be the set of all starting positions of occurrences of the words of $\mathcal{D}$ in w. Then

$$\mathcal{D}(w) = w[1..i_1 - 1], w[i_1..i_2 - 1], \ldots, w[i_k..|w|].$$

Lemma 4 *Let $\mathcal{D}$ be a set of words of the same length t. Let $i < j < k$ be three consecutive occurrences of a word $v \in \mathcal{D}$ in a word w. Assume that $i + t \geq k$. Then $\mathcal{D}(w)[i, j-1] = \mathcal{D}(w)[j, k-1]$.*

Proof: By Lemma 2, $k - j = j - i$ and $k - j$ is a period of $u = w[i..k+t-1]$. It is enough to prove that for $0 \leq p < j - i$ the words of length t starting at positions $i + p$ and $j + p$ in w are identical. This is true since these two words are wholy contained in u and the distance between their occurrences in u is equal to $j - i$ which is a period of u. □

For a sequence of words $\mathcal{P}$ and a positive integer t denote by $\mathcal{P}^t$ a sequence of words consisting of t repetitions of $\mathcal{P}$. Let $\mathcal{P}$ be a sequence of words $w_1, \ldots, w_k$. Denote $\mathrm{concat}(\mathcal{P}) = w_1 \ldots w_k$. $\mathrm{concat}(\mathcal{P})$ is the concatenation of all elements of $\mathcal{P}$.

Lemma 5 *Let $\mathcal{D}$ be a set of words of the same length.*

- *If $\ker_{\mathcal{D}}(w[i,j])$ is defined then*

$$\mathcal{D}(w)[i,j] = w_1, \ker_{\mathcal{D}}(w[i,j]), \mathcal{P}_1^{t_1}, \ldots, \mathcal{P}_s^{t_s}$$

where $w_1 = \mathrm{first}_{\mathcal{D}}(w[i,j])$, $\mathcal{P}_i$ are sequences of words, $1 \leq s \leq 2\mathrm{card}(\mathcal{D})$, $1 \leq \mathrm{length}(\mathcal{P}_i) \leq \mathrm{card}(\mathcal{D})$, and $\mathrm{concat}(\mathcal{P}_1^{t_1} \ldots \mathcal{P}_s^{t_s}) = \mathrm{last}_{\mathcal{D}}(w[i,j])$.

- *If $\ker_{\mathcal{D}}(w[i,j])$ is undefined then*

$$\mathcal{D}(w)[i,j] = w_1, \mathcal{P}_1^{t_1}, \ldots, \mathcal{P}_s^{t_s}$$

where w_1 is the first factor of $\mathcal{D}(w)[i,j]$, $\mathcal{P}_i$ are sequences of words, $0 \leq s \leq 2\mathrm{card}(\mathcal{D})$ and $\mathrm{length}(\mathcal{P}_i) \leq \mathrm{card}(\mathcal{D})$.

Proof: Assume that all words in $\mathcal{D}$ are of length t. Let $i_1 < i_2 < \ldots < i_k$ be all starting positions of occurrences of words of $\mathcal{D}$ in w such that $i \leq i_n \leq j$. The occurrences can be divided into two groups: those which are wholy contained in $w[i..j]$ and those which contain position $j+1$. The starting positions of the first ones does not exceed $j - t + 1$ while the starting positions of the second ones exceed $j - t + 1$. We assume that the set of the first occurrences is not empty. The reasoning in the other case is similar. Let m be maximal index satisfying $i_m \leq j - t + 1$. Then $i_1 < \ldots < i_m$ belong to the first group and $i_{m+1} < \ldots < i_k$ to the second one. By the definition of $\mathcal{D}(w[i,j])$ we have

$$\begin{aligned}\mathcal{D}(w[i,j]) \quad = \quad & w[i..i_1 - 1], w[i_1..i_2 - 1], \\ & \ldots, w[i_{m-1}..i_m - 1], w[i_m..j].\end{aligned}$$

Hence if $\ker_{\mathcal{D}}(w[i..j])$ is defined, then

$$\mathcal{D}(w)[i,j] = \mathrm{first}_{\mathcal{D}}(w[i,j]), \ker_{\mathcal{D}}(w[i,j]), \mathcal{P}.$$

where $\mathcal{P}$ is a sequence of factors such that $\mathrm{concat}(\mathcal{P}) = \mathrm{last}_{\mathcal{D}}(w[i,j])$. If $\ker_{\mathcal{D}}(w[i..j])$ is not defined, then

$$\mathcal{D}(w)[i,j] = \mathrm{first}_{\mathcal{D}}(w[i,j]), \mathcal{P}.$$

where $\mathcal{P}$ is a sequence of factors such that $\mathrm{concat}(\mathcal{P}) = \mathrm{last}_{\mathcal{D}}(w[i,j])$. $\mathcal{P}$ is determined by positions $i_{m+1} < \ldots < i_k$ ie. by occurrences of words of $\mathcal{D}$ which contain the position $j + 1$. Note that for three occurrences of the same word from this group of occurrences Lemma 4 becomes applicable. Scan occurrences from left to right searching for earliest two occurrences of the same word. If words occurs only once then $k = \mathrm{length}(\mathcal{P}) \leq \mathrm{card}(\mathcal{D})$ and we are done. Otherwise, let $j_1 < j_2$ be found starting positions of occurrences of a word v. Let $j_3 < \ldots < j_p$ be the other occurrences of v. Then by Lemma 4 $\mathcal{P} = (\mathcal{P}_1)^1, (\mathcal{P}_1')^{p-1}, \mathcal{P}'$ where $\mathrm{length}(\mathcal{P}_1)$, $\mathrm{length}(\mathcal{P}_1') \leq \mathrm{card}(\mathcal{D})$ and $\mathcal{P}'$ is determined by j_p and occurrences starting to the right of j_p. Note that among them there are no occurrences of v. We continue scanning starting from the position to the right of j_p. In this way we obtain

$$\begin{aligned}\mathcal{P} \quad = \quad & (\mathcal{P}_1)^1, (\mathcal{P}_1')^{p_1}, (\mathcal{P}_2)^1, (\mathcal{P}_2')^{t_2}, \\ & \ldots, (\mathcal{P}_{q-1})^1, (\mathcal{P}_{q-1}')^{p_{q-1}}, (\mathcal{P}_q)^1\end{aligned}$$

which is of the required form. It is enough to prove that $q \leq \mathrm{card}(\mathcal{D})$. After each phase of scanning the set of words from $\mathcal{D}$ which occurs at remaining positions loses at least one element. This justifies the inequality $q \leq \mathrm{card}(\mathcal{D})$. □

We will use Lemma 5 in the following way. Let i, j be two starting positions of occurrences of a word v in w. If $\ker_{\mathcal{D}}(v)$

is defined then the sequences of factors corresponding to $[i, i+|v|-1]$, $[j, j+|v|-1]$ in $\mathcal{D}(w)$ consist of a factor followed by a common part $\ker_{\mathcal{D}}(v)$ and ended by a regular shortly presented sequence of factors. If $\ker_{\mathcal{D}}(v)$ is not defined then the sequences corresponding to these intervals in $\mathcal{D}(w)$ form regular shortly presented sequence of factors.

4 $\mathcal{D}$-transformations

Given a $\mathcal{D}$-factorization we define a transformation $\rightarrow_{\mathcal{D}}$ which takes a pair: a word equation and its solution $(u = v, h)$ and produces a triple: a word equation $u' = v'$, its solution h' and a function F such that $F(h') = h$. The resulting triple has the following features:

- the alphabet of constants of $u' = v'$ is **the set of words over** Σ, i.e. Σ^*,
- the set of variables of $u' = v'$ is the set of variables of $u = v$,
- a variable X such that $\ker_{\mathcal{D}}(h(X))$ is undefined does not occur in $u' = v'$,
- for each variable X such that $\ker_{\mathcal{D}}(h(X))$ is defined, $h'(X) = \ker_{\mathcal{D}}(h(X))$,
- $\mathcal{D}(h(u)) = h'(u')$.

Observe here that defining $u' = v'$ we change the alphabet of constants from Σ to Σ^*. Words over Σ^* correspond to sequences of words over Σ. The transformation $\rightarrow_{\mathcal{D}}$ consists of a few steps. First we construct u' by replacing parts of $\mathcal{D}(h(u))$ by variables. We search for all occurrences of variables in u and for an occurrence of a variable X in u at position r we do the following with $\mathcal{D}(h(u))$. If $\ker_{\mathcal{D}}(h(X))$ is not defined, we do nothing. If $\ker_{\mathcal{D}}(h(X))$ is defined, then we write u in form yXz with $|y| = r - 1$. Denote $i = |h(y)| + 1$ and $j = |h(yX)|$. Clearly, $h(u)[i..j] = h(X)$. Take the sequence of words from $\mathcal{D}(h(u))$ corresponding to $[i, j]$ and replace in $\mathcal{D}(h(u))$ its part $\ker_{\mathcal{D}}(h(X))$ by X. To obtain u' we do the above for all occurrences of variables in u. Similarly we define v' on the basis of $\mathcal{D}(h(v))$ $(= \mathcal{D}(h(u))$ since h is a solution of $u = v)$. Clearly, a morphism h' satisfying $h'(X) = \ker_{\mathcal{D}}(h(X))$, for all variables X such that $\ker_{\mathcal{D}}(h(X))$ is defined, is a solution of $u' = v'$.

Let g be a solution of $u' = v'$. We define

$$F(g)(X) = h(X)$$

if $\ker_{\mathcal{D}}(h(X))$ is not defined and

$$F(g)(X) = \text{first}_{\mathcal{D}}(h(X))\text{concat}(g(X))\text{last}_{\mathcal{D}}(h(X)))$$

if $\ker_{\mathcal{D}}(h(X))$ is defined. By the definition of F, h' and concat we have $F(h') = h$.

Lemma 6 *Let $(u = v, h) \rightarrow_{\mathcal{D}} (u' = v', h', F)$. If g is a solution of $u' = v'$ then $F(g)$ is a solution of $u = v$. Moreover, $F(g)(u) = \text{concat}(g(u'))$.*

Proof: Let g be a morphism which takes sequences of words and variables occurring in $u' = v'$, which returns sequences of words and which is identity on sequences of words without variables. We will prove that $F(g)(u) = \text{concat}(g(u'))$. Observe that $g(u')$ is a sequence $\mathcal{D}(h(u))$ in which in some places sequences equal to $\ker_{\mathcal{D}}(h(X))$ are replaced by $g(X)$. The places correspond to occurrences in u of the variables X for which $\ker_{\mathcal{D}}(h(X))$ is defined. On the other hand $F(g)(u)$ is the word $h(u)$ where places corresponding to occurrences in u of a variable X such that $\ker_{\mathcal{D}}(X)$ is not defined are replaced by $h(X)$ ie. those parts of $h(u)$ are unchanged. The places which correspond to occurrences in u a variable X such that $\ker_{\mathcal{D}}(h(X))$ is defined are replaced by

$$\text{first}_{\mathcal{D}}(h(X))\text{concat}(g(X))\text{last}_{\mathcal{D}}(h(X)),$$

ie. $\text{concat}(\ker_{\mathcal{D}}(h(X))$ is replaced by $\text{concat}(g(X))$. Since $\text{concat}(\mathcal{D}(h(u)) = h(u)$ we have $F(g)(u) = \text{concat}(g(u'))$.

Similarly, we prove that $F(g)(v) = \text{concat}(g(v'))$. Let g be a solution of $u' = v'$. Then

$$F(g)(u) = \text{concat}(g(u')) = \text{concat}(g(v')) = F(g)(v).$$

Hence, $F(g)$ is a solution of $u = v$. □

Lemma 7 *Let $(u = v, h) \rightarrow_{\mathcal{D}} (u' = v', h', F)$ and h be minimal. Then for each variable X occurring in $u' = v'$, $h'(X)$ consists of constants of $u' = v'$.*

Proof: Suppose that $h'(X)$ contains a constant which does not occur in $u' = v'$. Let Ψ be a morphism which is an identity on constants of $u' = v'$ and the empty word for other constants. Then $\Psi \circ h'$ ($\circ$-composition of functions) is also a solution of $u' = v'$. Additionally, since $\Psi \circ h'(u')$ is a proper subsequence of $h'(u')$, we have

$$\begin{aligned} |F(\Psi \circ h')(u)| &= |\text{concat}(\Psi \circ h'(u'))| < \\ &< |\text{concat}(h'(u'))| = |h(u)|. \end{aligned}$$

Hence, h cannot be minimal. □

We say that two equations $e_1 : u_1 = v_1$ and $e_2 : u_2 = v_2$ are *isomorphic* if e_1 and e_2 are identical up to renaming constants and variables. Formally, e_1 is isomorphic to e_2 if there is a morphism Φ which maps different constants of e_1 to different constants of e_2, maps different variables of e_1 to different variables of e_2 and such that $\Phi(u_1) = u_2$ and $\Phi(v_1) = v_2$.

An *exponential expression* is an expression which is built on constants, variables, operation of concatenation and operation of exponentiation to a fixed power. The *length of the exponential expression* is the number of symbols we need to put it down. We assumme that each occurrence of a constant and a variable requires one symbol, concatenation zero symbols, exponentiation three symbols (two parentheses and an exponent). An equation of the form $exp_1 = exp_2$ where exp_1 and exp_2 are exponential expressions is called *exponential equation*. The *length of the exponential equation* is the sum of the lengths of exp_1 and exp_2 plus one (for the symbol $=$). An exponential equation e is a representation of an equation which is obtained from e by changing $(\mathcal{P})^t$ by $\mathcal{P}$ repeated t times.

Lemma 8 *Let C and V be two sets such that $\text{card}(C) = \text{card}(V) = n$. Then for each word equation e represented by an exponential equation of length at most n there is a word equation over the alphabet of constants C and the alphabet of variables V which is isomorphic to e and which is represented by an exponential equation of length at most n.*

Proof: Let $const(e)$ and $var(e)$ be the set of constants and the set of variables occurring in e. Clearly, $\text{card}(const(e))$, $\text{card}(const(e)) \leq n$. The required isomorphism is any morphism Φ such that

$$\Phi(const(e)) \subseteq C,\ \Phi(var(e)) \subseteq V,$$

and for $x, y \in const(e) \cup var(e)$ if $x \neq y$ then $\Phi(x) \neq \Phi(y)$. □

Denote by $\Delta(m,n)$ the number of nonisomorphic equations which are produced by transformations $\rightarrow_{\mathcal{D}}$, where $\mathcal{D}$ consists of at most m words of the same length, from pairs (e,h) such that $|e| \leq n$ and h is minimal.

Lemma 9

$$\Delta(n,m) = 2^{O(\max\{n,\log m\}m^2n)}.$$

Proof: Let $(e,h) \rightarrow_{\mathcal{D}} (e',h',F)$. Assume, card$(\mathcal{D}) \leq m$, h is minimal, $|e| \leq n$. e' is described by an expression which is built on constants, variables, the sign =, parantheses (and) (we need them to express $\mathcal{P}^t$), and numbers which are exponents. Since h is minimal and concat$(h'(u')) = h(u)$ all exponents are not greater than 2^{cn}. First we estimate the length of the exponential equation which represents e'. In the worst case (the longest e' is created) each constant of e corresponds to a constant in e' and each variable is replaced by a constant followed by a variable followed by an exponential expression. By Lemma 5 the expression contains at most $2m^2$ constant, $2m$ exponents, and $4m$ parentheses (we express $\mathcal{P}^t$ as $(\mathcal{P})t$). Hence the length of the expression is at most $2m^2+6m$ and the length of the exponential expression for e' is at most $c(n,m) = n(2m^2+6m+2)+1$ (1 is for =). By Lemma 8 it is enough to consider equations over a fixed alphabets of constants and variables both of size $c(n,m)$. At each position of the expression we can put one of at most $3+2^{cn}+c(n,m)+c(n,m)+1$ (symbols (,),=, and numbers which are exponents, and constants, and variables, and additional symbol '#' (SPACE) to obtain expressions which are shorter than $c(n,m)$). Hence, $\Delta(n,m) \leq (3+2^{cn}+2c(n,m)+1)^{c(n,m)}$ and finally $\Delta(n,m) = 2^{O(\max\{n,\log m\}m^2n)}$. □

5 C_l-factorizations

Fix a word equation $e : u = v$ and its minimal solution h. Denote $n = |e|$. A border is defined for a graphical representation of a word being a sequence of symbols written along a straight line. A *border* in a word w is a space between two consecutive symbols of w or a space before or after the word. Each word w contains $|w|+1$ borders. Let $u = u_1u_2$ for some words u_1, u_2. A border between $h(u_1)$ and $h(u_2)$ in $h(u)$ is called *left cut*. Similarly, let $v = v_1v_2$ for some words v_1, v_2. A border between $h(v_1)$ and $h(v_2)$ in $h(v) = h(u)$ is called *right cut*. A *cut* in $h(u)$ is a border being left cut or right cut. Note that the borders before $h(u)$ and after it are left and right cuts and therefore the total number of cuts is at most n. A *characteristic* word of $h(u)$ is a subword w of $h(u)$ such that there is an occurrence of w in $h(u)$ whose starting symbol is to the right of a cut or whose ending symbol is to the left of a cut. Formally, w is a characteristic word if $h(u)$ can be written in form pws where p and s are such that there is a cut between p and ws or pw and s.

Let $\mathcal{C}_l$ be the set of characteristic words of length l. The set $\mathcal{C}_l$ is not empty for $l \leq |h(u)|$. Clearly, for each l, card$(\mathcal{C}_l) \leq 2n$. Our next proposition uses the fact that h is minimal.

Proposition 10 ([7],Lemma 6) *Each subword of $h(u)$ has an occurrence over a cut.*

Lemma 11 *Each factor in $\mathcal{C}_l(h(u))$ is of length at most l.*

Proof: Suppose it is not true ie. there is no occurrence of a word from $\mathcal{C}_l$ at l consecutive positions $i, i+1, \ldots, i+l-1$ for some i. Since a suffix of $h(u)$ of length l is in $\mathcal{C}_l$ we have $i+l-1 \leq |h(u)|-l$, ie. $i+2l-1 \leq |h(u)|$. Consider the word $w = h(u)[i..i+2l-1]$. By Proposition 10 there is an occurrence of w in $h(u)$ which covers a cut. The cut divides the occurrence into two parts. The longer one which is of length at least l contains a characteristic word of length l. This contradicts the definition of i. □

Denote $b(n) = 2^{cn+1}n$.

Lemma 12 *If $|h(u)| \geq lb(n)$ ($\mathcal{C}_{lb(n)}$ is not empty) then*

$$|h(u)| > l * \text{length}(\mathcal{C}_{lb(n)}(h(u)).$$

Proof: Let $1 = i_1 < i_2 < \ldots < i_k$, for $1 \leq k$, be occurrences of the words $\mathcal{C}_{lb(n)}$ in $h(u)$. Clearly $|h(u)| - i_k + 1 = lb(n)$ and length$(\mathcal{C}_{lb(n)}) = k$.

Lemma 13 *$i_{s+2n} - i_s > l*(2n)$, for $1 \geq s$ and $s+2n \leq k$.*

Proof: Since card$(\mathcal{C}_{lb(n)}) \leq 2n$, among the occurrences $i_s < i_{s+1} < \ldots < i_{s+2n}$ there are two, say $i > j$, occurrences of the same word $v \in \mathcal{C}_{lb(n)}$. If these two occurrences do not overlap, then $i - j \geq lb(n) \geq l*(2n)$. Suppose they overlap. Then $i-j$ is a period of v. Since h is minimal, by Proposition 3, the period $i-j$ cannot repeat in v more than 2^{cn} times ie. $(i-j)*2^{cn} > |v| = lb(n)$. This completes the proof of the claim. □

Applying Lemma 13 inductively on t we have $i_{1+2tn} - i_1 \geq l*(2tn)$, for $1+2tn \leq k$ and $t \geq 0$. Take maximal s such that $1+2sn \leq k$. Then $k - 2sn \leq 2n$. We have

$$\begin{aligned} |h(u)| &\geq (i_{1+2sn} - i_1) + (|h(u)| - i_k + 1) \geq \\ &\geq l*(2sn) + lb(n) > l*(2sn) + l(2n) \geq \\ &\geq l*k. \end{aligned}$$

This completes the proof. □

Denote $\mathcal{F}_i = \mathcal{C}_{b(n)^i}$ and let $(u_i = v_i, h_i, F_i)$ be such that $(u = v, h) \rightarrow_{\mathcal{F}_i} (u_i = v_i, h_i)$. Denote by e_i the equation $u_i = v_i$.

Theorem 14 $|h(u)| \leq b(n)^{\Delta(n,2n)}$.

Proof: Suppose that $|h(u)| > b(n)^{\Delta(n,2n)}$. Then the sets $\mathcal{F}_i$ are not empty for $i \leq \Delta(n,2n)$. Since card$(\mathcal{F}_i) \leq 2n$, by the definition of $\Delta(n,m)$, among all pairs (e_i,h_i), for $0 \leq i \leq \Delta(n,2n)$ there are two (e_k,h_k,F_k) and (e_l,h_l,F_l) with $l > k$ such that the equations e_k and e_l are isomorphic. This means that an isomorphic image of h_l is a solution of e_k. Let Φ be the isomorphism between e_k and e_l ie. $\Phi(u_l) = u_k$ and $\Phi(v_l) = v_k$. Denote by h'_l a morphism such that $h'_l(a) = \Phi(h_l(a))$ for each constant a occurring in e_l and $h'_l(\Phi(X)) = \Phi(h_l(X))$ for each variable X of e_l. Observe that Φ is not necessarily well defined on constants which does not occur in e_l. However, by Lemma 7 since h is minimal, for variables X of e_l, $h_l(X)$ consists of constants of e_l. By Lemma 6 $F_k(h'_l)$ is a solution of e. We will prove that $|F_k(h'_l)(u)| < |h(u)|$ which contradicts minimality of h. By Lemma 12 we have

$$|h(u)| > b(n)^{l-1}\text{length}(\mathcal{F}_l(h(u)).$$

Observe that

$$\text{length}(\mathcal{F}_l(h(u)) = \text{length}(h_l(u_l)) = \text{length}(h'_l(u_k))$$

Hence,

$$\begin{aligned} |h(u)| &> b(n)^{l-1}\text{length}(h_l'(u_k)) \geq \\ &\geq b(n)^k\text{length}(h_l'(u_k))) \geq \\ &\geq |\text{concat}(h_l'(u_k))| = |F_k(h_l')(u)|. \end{aligned}$$

Last two inequalities are by Lemma 11, by the fact that $h'(u_k)$ consists of constants of e_k (which are words of length at most $b(n)^k$) and by Lemma 6. This completes the proof. □

As an immediate consequence of Theorem 14 and the result in [7] we have

Corollary 15 *1. Let h be a minimal solution of a word equation $e : u = v$. Then $|h(u)| = 2^{2^{O(|e|^4)}}$.*
2. The problem of solvability of word equations is in NEXPTIME.

Acknowledgements

I would like to thank Prof. Wojciech Rytter (Warsaw University, Poland and University of Liverpool, Great Britain) who aroused my iterest in Makanin's algorithm.

References

[1] Diekert V., Makanin's algorithm, a chapter in a book on combinatorics of words, 1998, personal comunication (Volker.Diekert@informatik.uni-stuttgart.de).

[2] Jaffar J., Minimal and complete word unification, *Journal of the ACM* **37**(1), 47-85, 1990.

[3] Gutierrez C., Satisfiability of word equations with constants is in exponential space, in: *Proc. FOCS'98*, IEEE Computer Society Press, Palo Alto, California.

[4] Karhumäki J., Mignosi F., Plandowski W., The expressibility of languages and relations by word equations, in: *Proc. ICALP'97*, LNCS 1256, 98-109, 1997.

[5] Koscielski A., Pacholski L., Complexity of Makanin's Algorithm, *Journal of the ACM* **43**(4), 670-684.

[6] Makanin G. S., The problem of solvability of equations in a free semigroup, *Mat. Sb.*, **103**(2), 147-236. In Russian; English translation in: *Math. USSR Sbornik*, **32**, 129-198, 1977.

[7] Plandowski W., Rytter W., Application of Lempel-Ziv encodings to the solution of word equations, in: *Proc. ICALP'98*, LNCS 1443, 731-742, 1998.

[8] Schulz K.U., Makanin's algorithm for word equations: two improvements and a generalization, in: *Proc. IWWERT'90*, LNCS 572, 85-150, 1992.