FI MU

Faculty of Informatics
Masaryk University

On the Pattern Equations

by

Ivana Černá
Ondřej Klíma
Jiří Srba

FI MU Report Series FIMU-RS-99-01

July 1999

On the Pattern Equations*

Ivana Černá[1], Ondřej Klíma[2], and Jiří Srba[1]

[1] Faculty of Informatics MU, Botanická 68a, 602 00 Brno, Czech Republic, {cerna,srba}@fi.muni.cz
[2] Faculty of Science MU, Dept. of Mathematics, Janáčkovo nám. 2a, 662 95 Brno, Czech Republic, klima@math.muni.cz

Abstract. Word equation in a special form $X = A$, where X is a sequence of variables and A is a sequence of constants, is considered. The problem whether $X = A$ has a solution over a free monoid (PATTERN-EQUATION problem) is shown to be NP–complete. It is also shown that disjunction of a special type equation systems and conjunction of the general ones can be eliminated. Finally, the case of stuttering equations where the word identity is read modulo $x^2 = x$ is mentioned.

1 Introduction

In computer science many natural problems lead to solving equations. It is the main topic in several fields such as logic programming and automated theorem proving where especially unification plays a very important role. A number of problems also exploit semantic unification, which is in fact solving word equations in some variety. A very famous result by Makanin (see [10]) shows that the question whether an equation over a free monoid has a solution is decidable. It can be even generalized in the way that existential first-order theory of equations over free monoid is decidable. Moreover adding regular constraints on the variables (i.e. predicates of the form $x \in L$ where L is a regular language) preserves decidability [12].

In this paper we consider a very practical issue of a certain subclass of equations which we call pattern equations. Many problems such as pattern matching and speech recognition/synthesis lead to this kind of equations where we consider on the lefthand side just variables and on the righthand side only constants. This work has been mostly inspired by the papers [4] and [5] where the basic approach – syllable-based speech synthesis – is in assigning prosody attributes to a given text and segmentation into syllable segments. This problem can be modelled by pattern equations over free monoid resp. idempotent semigroup and is trivially decidable. However, we could ask whether a polynomial algorithm exists to find a solution. Unfortunately, this problem appears intractable (supposing P≠NP) since we prove that it is NP–complete. One of the ways how to solve the problem is to use heuristic algorithms. They are

* The paper is supported by the Grant Agency of the Czech Republic, grant No. 201/97/0456 and by grant FRVŠ 409/1999

the current field of interest in speech synthesis. Another approach that could be used for solving the problem is *Concurrent Constraint Programming*. For the background see [1].

We may also ask whether for a system of pattern equations (connected by conjunction resp. disjunction) exists a single equation preserving satisfiability and/or solutions. In the positive case a question of the transformation complexity arises. If the transformation can be done effectively (e.g. in linear time as it is shown in Section 4), we can concentrate on finding heuristics just for a single pattern equation where the situation could be easier to manage. The elimination of conjunction resp. disjunction is generally possible [12]. What we show is that we can find an equation preserving solutions of the system (and thus also satisfiability) which is again of our special type, i.e. it is a pattern equation. We demonstrate that for conjunction no extention of the constant and variable alphabet is necessary and the length growths polynomially where the degree of the polynomial depends on the number of equations. For the practical purposes it is much more convenient to add some new symbols into the alphabet and thus achieve just a linear space extention, which is also manifested in our paper. Similar results are formulated for disjunction.

We also examine the solvability of the equations in the variety of idempotent semigroups (bands) which we call stuttering equations. Their name comes from practical motivations. For example in the speech recognition the speaker sometimes stutters some words and we would like to eliminate this effect and enable the correct variables assignation even in the case of stuttering. Therefore we allow to eliminate multiple occurences of the same constant into only one occurrence, which can be modelled by the identity $x^2 = x$.

Local finiteness of free bands yields decidability of the satisfiability problem even in the general case and we give an exponential upper bound on the length of any solution up to band identities. We also establish a polynomial time decision procedure for the word problem in idempotent semigroups. Consequently, the satisfiability problem of stuttering equations belongs to NEXPTIME. The complexity issues for stuttering *pattern* equations are also discussed.

2 Basic definitions

Let C be a finite set of *constants* and V be a finite set of *variables* such that $C \cap V = \emptyset$. A *word equation* $L = R$ is a pair $(L, R) \in (C \cup V)^* \times (C \cup V)^*$. A *system of word equations* is a finite set of equations of the form $\{L_1 = R_1, \ldots, L_n = R_n\}$ for $n > 0$. A *solution* of such a system is a homomorphism $\alpha : (C \cup V)^* \to C^*$ which behaves as an identity on the letters from C and equates all the equations of the system, i.e. $\alpha(L_i) = \alpha(R_i)$ for all $1 \leq i \leq n$. Such a homomorphism is then fully established by a mapping $\alpha : V \to C^*$. A solution is called *non-singular*, if $\alpha(x) \neq \epsilon$ for all $x \in V$. Otherwise we will call it *singular*. We say that the problem for word equations is satisfiable whenever it has a solution.

Makanin in [10] shows that the satisfiability problem for word equations is decidable. This problem is easily seen to be semidecidable. The decidability is established by giving an upper border on the length of the minimal solution. The decidability was later solved in more general setting by Schulz (see [12]) where for each $\alpha(x)$ is given a regular constraint that must be satisfied.

2.1 Notation

In what follows we will use an uniform notation. The set $C = \{a, b, c, \dots\}$ denotes the alphabet of constants and $V = \{x, y, z, \dots\}$ stands for variables (unknowns) with the assumption that $C \cap V = \emptyset$. We will use the same symbol α for the mapping $\alpha : V \to C^*$ and its unique extention to the homomorphism $\alpha : (C \cup V)^* \to C^*$. Sometimes we write α_x instead of $\alpha(x)$. The symbol for the empty word will be denoted as ϵ (for any $w \in (C \cup V)^*$ holds that $\epsilon.w = w.\epsilon = w$). The length of a word w is denoted as $|w|$, i.e. $|a_1 a_2 \dots a_k| = k$ for $k \geq 0$.

2.2 Pattern equations

In this paper we focus on a special kind of word equations which we call *pattern equations*.

Definition 1. *A* pattern equation system *is the set* $\{X_1 = A_1, \dots, X_n = A_n\}$ *where* $X_i \in V^*$ *and* $A_i \in C^*$ *for all* $1 \leq i \leq n$. *The* solution *(both singular and non-singular) of the pattern equation system is defined as in the general case.*

Two natural decidability problems (PATTERN-EQUATION and NON-SINGULAR-PATTERN-EQUATION problem) appear in this context and are defined bellow.

Definition 2. *Given a pattern equation system* $\{X_1 = A_1, \dots, X_n = A_n\}$ *as an instance of the* PATTERN-EQUATION *problem, the task is to decide whether this system has a solution. If we require the solution to be non-singular we call it the* NON-SINGULAR-PATTERN-EQUATION *problem.*

We give an example of a pattern equation system and demonstrate its solutions.

Example 1. Let us have the following system where $C = \{a, b\}$, $V = \{x, y, z\}$ and the pattern equations are

$$\{xyxy = abbabb, \quad yzy = bbbabbb\}.$$

A singular solution exists

$$\alpha(x) = abb, \quad \alpha(y) = \epsilon, \quad \alpha(z) = bbbabbb,$$

however, there is also a non-singular solution

$$\beta(x) = a, \quad \beta(y) = bb, \quad \beta(z) = bab.$$

There is no reason for having just one solution, which is demonstrated also by our example since

$$\gamma(x) = ab, \quad \gamma(y) = b, \quad \gamma(z) = bbabb$$

is another non-singular solution.

3 NP–completeness of the PATTERN-EQUATION problem

In this section we show that the PATTERN-EQUATION problem is NP–complete. First observe that the problem is in NP since any solution is linearly bounded in length w.r.t. the pattern equation system (each $\alpha(x)$, for an occurrence of the variable x, must be shorter than its righthand side constant string). Thus we can nondeterministically guess for α and in polynomial time we can check whether it is a solution. On the other hand to prove that PATTERN-EQUATION problem is NP–hard we reduce the TRIPARTITE-MATCHING problem to it. This proof has been independently done in more general setting also by Robson and Diekert [2] using the reduction from 3-SAT.

Suppose we have three sets B, G and H (boys, girls and homes) each containing exactly n elements for a natural number n. Let $T \subseteq B \times G \times H$. The TRIPARTITE-MATCHING problem is to find a subset $S \subseteq T$ of n elements such that $\{b \in B \mid \exists g \in G, \exists h \in H : (b, g, h) \in S\} = B$, $\{g \in G \mid \exists b \in B, \exists h \in H : (b, g, h) \in S\} = G$ and $\{h \in H \mid \exists b \in B, \exists g \in G : (b, g, h) \in S\} = H$. That is: each boy is matched to a different girl and they have their own home. The TRIPARTITE-MATCHING problem is known to be NP–complete (see e.g. [11]) and we show a polynomial reduction from it to the PATTERN-EQUATION problem.

Theorem 1. *The* PATTERN-EQUATION *problem is NP–complete.*

Proof. Suppose we have $T \subseteq B \times G \times H$ an instance of the TRIPARTITE-MATCHING problem where $B = \{b_1, \ldots, b_n\}$, $G = \{g_1, \ldots, g_n\}$ and $H = \{h_1, \ldots, h_n\}$. We will find an instance of PATTERN-EQUATION problem which is satisfiable if and only if the TRIPARTITE-MATCHING problem has a solution. Let us suppose that $T = \{T_1, \ldots, T_k\}$ and we introduce a new variable t_i for each T_i where $1 \leq i \leq k$. Let us define

$$\phi_B \equiv \wedge_{i=1}^{n} \vee \{t_j \mid \exists g \in G, \exists h \in H : (b_i, g, h) = T_j\},$$

$$\phi_G \equiv \wedge_{i=1}^{n} \vee \{t_j \mid \exists b \in B, \exists h \in H : (b, g_i, h) = T_j\},$$

$$\phi_H \equiv \wedge_{i=1}^{n} \vee \{t_j \mid \exists b \in B, \exists g \in G : (b, g, h_i) = T_j\}.$$

Let us consider the formula

$$\phi \equiv \phi_B \wedge \phi_G \wedge \phi_H.$$

We can see that the TRIPARTITE-MATCHING problem has a solution if and only if there exists a valuation that satisfies the formula ϕ such that it assigns value true to the exactly one variable in each clause.

Observe that ϕ is of the form

$$\phi \equiv C_1 \wedge C_2 \wedge \ldots \wedge C_{3n}$$

and assume that there is an empty clause in the conjunction (the formula is not satisfiable). Then we assign it the pattern equation system $\{x = a,\ x = b\}$ (this system certainly does not have any solution). In the other case we may suppose that

$$C_i \equiv t_{i,1} \vee t_{i,2} \vee \ldots \vee t_{i,j_i},$$

where $1 \leq j_i$ for all $1 \leq i \leq 3n$. Then we assign it the following pattern equation system $\mathcal{P}$:

$$\begin{array}{llll} \{ & t_{1,1} & \ldots\ t_{1,j_1} & = a, \\ & t_{2,1} & \ldots\ t_{2,j_2} & = a, \\ & \vdots & \quad\vdots & \ \vdots \\ & t_{3n,1} & \ldots\ t_{3n,j_{3n}} & = a\ \ \} \end{array}$$

The situation when the variable $t_{i,j}$ is true corresponds to $\alpha(t_{i,j}) = a$ and if $t_{i,j}$ is false it corresponds to $\alpha(t_{i,j}) = \epsilon$. It is straightforward that ϕ has a valuation that assigns value true to the exactly one variable in each clause if and only if $\mathcal{P}$ is satisfiable. Thus we reduced the TRIPARTITE-MATCHING problem to the PATTERN-EQUATION problem. Together with the fact that PATTERN-EQUATION problem is in NP we get the NP–completeness. □

Using the same proof technique as above we can also easily see the validity of the following theorem.

Theorem 2. *The* NON-SINGULAR-PATTERN-EQUATION *problem is NP–complete.*

Proof. The proof is the same as in the Theorem 1 except for the system of pattern equations which looks as follows.

$$\begin{array}{llll} \{ & t_{1,1} & \ldots\ t_{1,j_1} & = a^{j_1+1}, \\ & t_{2,1} & \ldots\ t_{2,j_2} & = a^{j_2+1}, \\ & \vdots & \quad\vdots & \ \vdots \\ & t_{3n,1} & \ldots\ t_{3n,j_{3n}} & = a^{j_{3n}+1}\ \} \end{array}$$

The value true is represented by $\alpha(t_{i,j}) = aa$ and false by $\alpha(t_{i,j}) = a$, which gives the non-singular solution. □

Remark 1. Observe that for the NP–completeness it is sufficient to fix the constant alphabet just to one letter.

4 Elimination of conjunction and disjunction

In general case we may construct for an arbitrary system of word equations a single equation preserving solutions. For example Diekert in [6] used the following construction: the system $\{L_1 = R_1, \ldots, L_n = R_n\}$ and the equation

$$L_1 a \ldots L_n a L_1 b \ldots L_n b = R_1 a \ldots R_n a R_1 b \ldots R_n b$$

where a, b are distinct constants, have the same set of solutions. However, this construction is useless for the pattern equations. We show the way how to eliminate conjunction of pattern equations in the following theorem.

Theorem 3. *The set of solutions of a pattern equation system* $\{X = A,\ Y = B\}$ *is identical with the set of solutions of the pattern equation* $X^nY^m = A^nB^m$ *where* $n = \max\{|A|, |B|\} + 3$ *and* $m = n + 1$.

Proof. It is evident that each solution of the system $\{X = A,\ Y = B\}$ is also a solution of $X^nY^m = A^nB^m$. We need the following lemma to prove the opposite.

Lemma 1 ([7]). *Let* $A, B \in C^*$, $d = \gcd(|A|, |B|)$. *If two powers* A^p *and* B^q *of* A *and* B *have a common prefix of length at least equal to* $|A| + |B| - d$, *then* A *and* B *are powers of the same word.*

Let α be a solution of the equation $X^nY^m = A^nB^m$. We show that $|\alpha(X)| = |A|$. In such a case $\alpha(X) = A$, $\alpha(Y) = B$ and α is a solution of the system.

First suppose $|\alpha(X)| > |A|$. Then A^n and $\alpha(X)^n$ have a common prefix of length $n \cdot |A|$ and for $|A| > 0$ we get

$$n\,|A| = 2\,|A| + (n-2)\,|A| \geq 2\,|A| + \frac{n+1}{n}(n-3) \geq$$

$$2\,|A| + \frac{m}{n}\,|B| = |A| + \frac{n\,|A| + m\,|B|}{n} \geq |A| + |\alpha(X)|\,.$$

By Lemma 1 we know that $A = D^k$ and $\alpha(X) = D^i$ where $k, i \in \mathbb{N}$, $k < i$, $D \in C^*$ and D is primitive (it means that if $D = E^p$ then $p = 1$). If $|A| = 0$ then trivially $k = 0$ and D is a primitive root of $\alpha(X)$. Hence $D^{(i-k)n}\alpha(Y)^m = B^m$ and by the Lemma 1 (common prefix of the length $(i-k)n\,|D| \geq n\,|D| = |D| + (n-1)\,|D| \geq |D| + |B|$) we have that B and D must be the powers of the same word. Since D is primitive we may write $B = D^l$ where $l \in \mathbb{N}_0$. Finally $\alpha(Y) = D^j$ again by Lemma 1.

If $|\alpha(X)| < |A|$ we have $|\alpha(Y)| > |B|$ and we can similarly deduce the same equalities $\alpha(X) = D^i$, $\alpha(Y) = D^j$, $A = D^k$, $B = D^l$.

Now we solve an equation $ni + mj = nk + ml$ in non-negative integer numbers. The proof is complete if we show that this equation has only one solution, namely $i = k$, $j = l$. We recall that $k, l < n, m$ and $m = n + 1$. If i, j are such that $ni + mj = nk + ml$ then $i \equiv k \pmod{m}$ and if $i < k + m$ then $i = k$ and $j = l$. Suppose $i \geq k + m$. This implies that $ni + mj \geq ni \geq nk + nm > nk + ml$, which is a contradiction. □

Remark 2. The above construction is unfortunately quadratic in space. One can ask whether n and m in the Theorem 3 need to be greater than $|A|$ and $|B|$? The answer is positive and no improvements can be done. If we want to transform the system $\{x = c^k,\ y = c^l\}$ into a single equation (w.l.o.g. suppose that the equation is of the form $x^ny^m = c^p$) then in the case $l > n$ we have $p = nk + ml = n(k+m) + m(l-n)$ and an α defined by $\alpha(x) = c^{k+m}$, $\alpha(y) = c^{l-n}$ is a solution of the equation $x^ny^m = c^p$ whereas α is not a solution of $\{x = c^k,\ y = c^l\}$.

Remark 3. It is easy to see that the proof of the Theorem 3 is correct for an arbitrary n greater than $\max\{|A|, |B|\} + 3$ and $m = n + 1$.

The Remark 2 shows that the construction in Theorem 3 can not be improved and moreover every construction preserving the alphabet of variables and constants requires a quadratic space extention. For a system of n equations where each one is bounded by the maximal length k we can repeatedly use the Theorem 3 pairwise and thus achieve the $\mathcal{O}(k^n)$ bound for the size of the resulted equation. On the other side the problem of conjunction elimination can be solved easily with extention of the sets C and V. This is much more suitable for practical purposes since the following construction is linear in space w.r.t. inputed pattern equation system.

Definition 3. *We say that two systems $\Sigma = \{L_1 = R_1, \ldots, L_n = R_n\}$ over C, V and $\Sigma' = \{L'_1 = R'_1, \ldots, L'_n = R'_n\}$ over C', V' are* equivalent *on the set of variables $\overline{V}$ where $\overline{V} \subseteq V \cap V'$ if and only if the sets of all solutions of the systems Σ and Σ' restricted on $\overline{V}$ are identical.*

Lemma 2. *Let $c \notin C$ be a new constant and $z \notin V$ be a new variable. Then the pattern equation system $\{X = A,\ Y = B\}$ over C, V and the pattern equation*

$$z(zXzY)^2 = c(cAcB)^2 \tag{1}$$

over $C \cup \{c\}$, $V \cup \{z\}$ are equivalent on the set V.

Proof. For every solution α of the system $\{X = A,\ Y = B\}$ we can easily construct a solution α' of the equation (1) such that $\alpha'|_V = \alpha$ and $\alpha'(z) = c$.

Now let α be a solution of the equation (1), i.e.

$$\alpha_z(\alpha_z\alpha_X\alpha_z\alpha_Y)^2 = c(cAcB)^2.$$

If $\mid \alpha_z \mid > 1$ then α_z has the prefix c^2 and on the lefthand side of the equality we have at least ten occurrences of c, however, on the righthand side of the equality only five. If $\alpha_z = \epsilon$ then $(\alpha_X\alpha_Y)^2 = c(cAcB)^2$ and the word on the lefthand side of the equality has even length while the word on the righthand side of the equality has odd length. For that reasons $\alpha(z) = c$, hence $\alpha(X) = A$ and $\alpha(Y) = B$. This means that $\alpha|_V$ is a solution of the system $\{X = A,\ Y = B\}$. □

Remark 4. If we want to find a single equation equivalent to the pattern equation system $\{X_1 = A_1, \ldots, X_n = A_n\}$ we can repeatedly eliminate it pairwise. However, this construction exceeds the linear growth in size. The elimination can be done much more elegantly by

$$z(zX_1zX_2z \ldots zX_n)^2 = c(cA_1cA_2c \ldots cA_n)^2$$

and the proof is similar to the previous one.

For disjunction we cannot expect theorems analogical to those we have given for conjunction. For example the disjunction pattern equation system $\{x = c,\ x = c^2\}$ cannot be replaced by a single equation over $\{c\}$, $\{x\}$.

Definition 4. *We say that a homomorphism α is a* solution of the disjunction pattern equation system $\{X_1 = A_1, \ldots, X_n = A_n\}$ *if and only if $\alpha(X_i) = A_i$ for some i, $1 \le i \le n$. The equivalence of two disjunction pattern equation systems is defined as in Definition 3.*

Lemma 3. *Let $c \notin C$ be a new constant and $z_1, z_2, z_3 \notin V$ be new variables. Then the disjunction pattern equation system $\{X = A,\ X = B\}$ over C, V and the pattern equation*

$$z_1 X^{10} z_1^2 z_2^{10} z_3^2 = cA^{10}c^2 B^{10}(cA^{10}c^2)^2 \tag{2}$$

over $C \cup \{c\}$, $V \cup \{z_1, z_2, z_3\}$ are equivalent on the set V.

Proof. It is easy to see that if $\alpha(X) = A$ then α' defined as $\alpha'|_V = \alpha$, $\alpha'(z_1) = c$, $\alpha'(z_2) = B$ and $\alpha'(z_3) = cA^{10}c^2$ is a solution of the equation (2). If $\alpha(X) = B$ then $\alpha'(z_1) = cA^{10}c^2$, $\alpha'(z_2) = \alpha'(z_3) = \epsilon$ is also a solution.

Let α be a solution of the equation (2), i.e.

$$\alpha_{z_1}\alpha_X^{10}\alpha_{z_1}^2\alpha_{z_2}^{10}\alpha_{z_3}^2 = cA^{10}c^2B^{10}(cA^{10}c^2)^2.$$

The number of occurences of c on the righthand side of the equation implies that $\alpha(X)$ and $\alpha(z_2)$ do not contain any c. Moreover if we denote p (resp. q) the number of occurences of c in $\alpha(z_1)$ (resp. $\alpha(z_3)$) we get $3p + 2q = 9$. This implies that $p = 1$ or $p = 3$. The first case constrains $\alpha(z_1) = c$ and so $\alpha(X) = A$ and the second one gives $\alpha(z_1) = cA^{10}c^2$, hence $\alpha(X) = B$. □

Remark 5. The construction from the Lemma 3 can be used for the disjunction pattern equation system $\{X = A_1, \ldots, X = A_n\}$ where $n = 2^k$. The number of equations equal to 2^k is important since it ensures the same lefthand side after the elimination of a pair of equations. If n is not equal to 2^k for some k we can always add the necessary number of equations $X = A_1$ and then eliminate the system into a single equation.

Corollary 1. *For an arbitrary finite set $\mathcal{S} = \{\alpha_i : V \to C^* | 1 \le i \le n\}$ there is a pattern equation over some C', V' such that the set of all its solutions restricted to V is identical with the given set $\mathcal{S}$.*

Proof. First, for every α_i we construct an equation $X_i = A_i$ with a single solution α_i by using repeatedly the Theorem 3. Moreover, in this construction we can use an universal n and m by Remark 3 and thus achieve the same lefthand sides $X_1 = X_2 = \ldots = X_n$. This yields a disjunction pattern equation system $\{X = A_1, \ldots, X = A_n\}$ which is equivalent to some single pattern equation by Remark 5. □

Note that in the case of non-singular solutions we may substitute in the Lemma 2 the equation (1) with $zXzYz = cAcBc$ and in the Lemma 3 the equation (2) with $z_1z_2z_1X^2z_1z_3z_1 = c^3A^2cB^2c^3$. It is easy to verify that all the theorems in this section are then also valid for the case of non-singular solutions.

5 Stuttering equations

It is sometimes interesting to consider the equations not only over a free monoid but for example in bands. Band is a semigroup where the identity $x^2 = x$ is satisfied. In our context it means that the equalities hold up to multiple occurences of certain substrings, which we call *stuttering*. This means that e.g. the equation $xx = aaa$ has no solution over free monoid but it is solvable over bands, since $\alpha(x) = a$ is a solution.

Let us define a binary relation $\rightarrow \subseteq C^* \times C^*$ such that $uvw \rightarrow uvvw$ for $u, v, w \in C^*$ and let $\sim$ be its symmetric and transitive closure, i.e. $\sim := (\rightarrow \cup \rightarrow^{-1})^*$. Then the identity $u = w$ holds in a free band if and only if $u \sim v$ (completeness of equational logic). Suppose we have a stuttering equation system $\{L_1 = R_1, \ldots, L_n = R_n\}$. A *solution* of such a system is a homomorphism $\alpha : (C \cup V)^* \rightarrow C^*$ which behaves as an identity on the letters from C and equates all the equations of the system, i.e. $\alpha(L_i) \sim \alpha(R_i)$ for all $1 \leq i \leq n$. We call the system a *stuttering pattern equation system* if the equations are of the form $\{X_1 = A_1, \ldots, X_n = A_n\}$.

The solvability problem for a single stuttering pattern equation $X = A$ is trivial since it si always solvable: $\alpha(x) = A$ for all $x \in V$. On the other hand the system is not always solvable: e.g. $\{x = a,\ x = b\}$ has no solution. This immediately gives that conjunction of stuttering pattern equations cannot be eliminated. In what follows we will exploit the fact that the word problem in bands is decidable (see [3] and its generalization [8]), which is mentioned in the next lemma.

Let $w \in C^*$. We define $c(w)$ – the set of all letters that occur in w, $0(w)$ – the longest prefix of w in $card(c(w)) - 1$ letters, $1(w)$ – the longest suffix of w in $card(c(w)) - 1$ letters.

Lemma 4 ([3]). *Let $u, v \in C^*$. Then $u \sim v$ if and only if $c(u) = c(v)$, $0(u) \sim 0(v)$ and $1(u) \sim 1(v)$.*

It is obvious that if a stuttering equation system has a solution then it has always infinitely many solutions, which we show in the following lemma.

Lemma 5. *Let $\{L_1 = R_1, \ldots, L_n = R_n\}$ be a general stuttering equation system and α its solution. Then also any β that satisfies $\alpha(x) \sim \beta(x)$ for all $x \in V$ (we simply write $\alpha \sim \beta$) is a solution.*

Proof. Immediate. □

This gives an idea that we should look just for the minimal representants of the classes up to $\sim$ that are solutions. We introduce the size of the solution α as $size(\alpha) := \max_{x \in V} \mid \alpha(x) \mid$. Given a stuttering equation system it is decidable whether the system is satisfiable because of the local finiteness of free idempotent semigroups. Following theorem just gives a precise exponential upper bound on the size of the minimal solution.

Theorem 4. *Let* $\{L_1 = R_1, \ldots, L_n = R_n\}$ *be a general stuttering equation system where* $card(C) \geq 2$. *If the system is satisfiable then there exists a solution* α *such that* $size(\alpha) \leq 2^{card(C)} + 2^{card(C)-2} - 2$.

Proof. Suppose the system is satisfiable, i.e. there is a solution β. Because of the Lemma 5 we know that any α, $\alpha \sim \beta$, is also a solution. We will find such an α which is small enough. The proof will be done by induction on k where $k = card(C)$.
k=2: The longest minimal word over a two-letter alphabet is of the length 3.
Induction Step: Suppose the IH holds for k and we show its validity for $k+1$. For each $w := \beta(x)$, $x \in V$, we will find some w' such that $w' \sim w$ and $\mid w' \mid \leq 2^{k+1} + 2^{k-1} - 2$. We know that $w \sim 0(w)a_1a_2 1(w)$ where $\{a_1\} = c(w) - c(0(w))$ and $\{a_2\} = c(w) - c(1(w))$ – see Lemma 4. Since $0(w)$ and $1(w)$ are in k letters, the IH can be applied and we can find some u, v of length less or equal $2^k + 2^{k-2} - 2$ such that $u \sim 0(w)$ and $v \sim 1(w)$. Thus we get that $ua_1a_2v \sim w$ and $\mid ua_1a_2v \mid \leq (2^k + 2^{k-2} - 2) + 2 + (2^k + 2^{k-2} - 2) = 2^{k+1} + 2^{k-1} - 2$. □

We can construct for each k, $k \geq 2$, a minimal word w_k which is of the length $2^k + 2^{k-2} - 2$ in the following way. Let $w_2 := a_1a_2a_1$ and $w_{k+1} := w_k a_{k+1} a_k w_k[a_k \mapsto a_{k+1}]$ where $w_k[a_k \mapsto a_{k+1}]$ means a substitution of a_k with a_{k+1} in the word w_k. This shows that the border given by the Theorem 4 is tight.

Corollary 2. *Given a stuttering equation system it is decidable whether the system is satisfiable.*

Proof. We have given an upper bound on the length of the minimal solution so it is sufficient to search for the solution among finitely many cases. □

In general it can be shown that there are stuttering equation systems such that all their minimal solutions are exponentially large w.r.t. number of letters from which it consists. Suppose the following sequence of equations: $z_1 = a_1$ and $z_{i+1} = z_i a_{i+1} z_i$ for a pairwise different sequence of constants $a_1, a_2, \ldots$. There is only one minimal solution α of the system and $\mid \alpha(z_i) \mid = 2^i - 1$.

If we want to investigate the complexity issues for stuttering equations, the first question we have to answer is the complexity of the checking whether some identity holds in bands. The same problem is easily solved in a free semigroup since we have to consider only associativity which means that the problem takes linear time. In the case of bands we can show that the word problem can be decided in polynomial time. Siekmann and Szabo in [13] showed that $\{xx \to x \mid c(x) \neq \emptyset\} \cup \{uvw \to uw \mid \emptyset \neq c(v) \subseteq c(u) = c(w)\}$ is confluent and terminating word rewriting system for bands. If we note that a string of the length k contains $\mathcal{O}(k^2)$ substrings (each substring is identified by its beginning and its length) we get that each reduction can be done in polynomial time. Since every reduction decreases the length of the word, we have a polynomial time decision algorithm for the word problem in bands.

An interesting question is whether a minimal solution of a stuttering *pattern* equation system can be of exponential length. In fact it turns out [9] that it is always of polynomial length. This long and technical proof exploits also the word rewriting system for bands by Siekmann and Szabo in [13]. Moreover, the NP–hardness of the satisfiability problem is shown.

Theorem 5 ([9]). *The decision problem whether a stuttering pattern equation system is satisfiable is NP–complete.*

References

[1] Brim L., Gilbert D.R., Jacquet J-M, Křetínský M.: A Process Algebra for Synchronous Concurrent Constraint Programming, Proceedings of the Fifth International Conference on Algebraic and Logic Programming (ALP'96), LNCS volume **1139** (1996) 165–178, Springer-Verlag.

[2] Diekert V., Robson J.M.: On Quadratic Word Equations, In Proceedings of 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS'99), LNCS volume **1563** (1999) 217–226, Springer-Verlag.

[3] Green J.A., Rees D.: On semigroups in which $x^r = x$, Proc. Camb. Phil. Soc. **48** (1952) 35–40.

[4] Kopeček I.: Automatic Segmentation into Syllable Segments, Proceedings of First International Conference on Language Resources and Evaluation (1998) 1275–1279.

[5] Kopeček I., Pala K.: Prosody Modelling for Syllable-Based Speech Synthesis, Proceedings of the IASTED International Conference on Artificial Intelligence and Soft Computing, Cancun (1998) 134–137.

[6] Lothaire M.: Algebraic Combinatorics on Words, Preliminary version available at `http://www-igm.univ-mlv.fr/~berstel/Lothaire/index.html`

[7] Lothaire, M.: Combinatorics on Words, Volume **17** of Encyclopedia of Mathematics and its Applications (1983) Addison-Wesley.

[8] Kad'ourek J., Polák L.: On free semigroups satisfying $x^r = x$, Simon Stevin **64**, No.1 (1990) 3–19.

[9] Klíma O., Srba J.: Complexity Issues of the Pattern Equations in Idempotent Semigroups, Technical Report FIMU-RS-99-02, Faculty of Informatics, Masaryk University (1999).

[10] Makanin, G. S.: The Problem of Solvability of Equations in a Free Semigroup, Mat. Sbornik. **103(2)** (1977) 147–236. (In Russian) English translation in: Math. USSR Sbornik **32** (1977) 129–198.

[11] Papadimitriou, C.H.: Computational Complexity, Addison-Wesley Publishing Company (1994), Reading, Mass.

[12] Schulz, K. U.: Makanin's Algorithm for Word Equations: Two Improvements and a Generalization, In Schulz, K.–U. (Ed.), Proceedings of Word Equations and Related Topics, 1st International Workshop, IWW-ERT'90, Tübingen, Germany, LNCS volume **572** (1992) 85–150, Berlin-Heidelberg-New York, Springer Verlag.

[13] Siekmann J., Szabó P.: A Noetherian and Confluent Rewrite System for Idempotent Semigroups, Semigroup Forum **25** (1982)

Publications in the FI MU Report Series are in general accessible via WWW and anonymous FTP:

`http://www.fi.muni.cz/informatics/reports/`
`ftp  ftp.fi.muni.cz (cd pub/reports)`

Copies may be also obtained by contacting:

Faculty of Informatics
Masaryk University
Botanická 68a
602 00 Brno
Czech Republic