Efficient Solving of the Word Equations in One Variable

S. Eyono Obono, P. Goralcik, and M. Maksimenko

LIR, LITP, Institut Blaise Pascal, France
Université de Rouen, 76134 Mont Saint Aignan Cedex
INSA de Rouen, BP 08, 76131 Mont Saint Aignan Cedex
e-mail: goralcik@litp.ibp.fr

Abstract. A *word equation* in n variables $x_1, \ldots, x_n$ over an alphabet C is a pair $E = (\varphi(x_1, \ldots, x_n), \psi(x_1, \ldots, x_n))$ of words over the alphabet $C \cup \{x_1, \ldots, x_n\}$. A *solution* of E is any n-tuple $(X_1, \ldots, X_n)$ of words over C such that $\varphi(X_1, \ldots, X_n) = \psi(X_1, \ldots, X_n)$. The existence of a solution for any given equation E is decidable, as shown by Yu. I. Khmelevskiĭ [3] for up to four variables and by G. S. Makanin [6] for any number of variables. However, as shown by A. Kościelski and L. Pacholski [4], these impressive decidability results can unfortunately not be matched by efficient algorithms of resolution, except for some restricted classes of equations. In this vein, W. Charatonik and L. Pacholski [1] give a polynomial algorithm, in terms of the equation length $|E| = |\varphi| + |\psi|$, for the equations in two variables and very roughly estimate at $O(|E|^5)$ the time complexity for solving those in one variable. For the latter, using rather fine combinatorial methods, we give an $O(|E| \log |E|)$ algorithm, the best one so far known.

1 Introduction

A word equation in one variable is a very simple object. In order to construct it, we need a set C, called *alphabet of constants*, and just one another letter x, not belonging to C, called *variable*. The words over C, including the empty word ε, constitute the free monoid C^* of *constant words*, while the words over the extended alphabet $C \cup \{x\}$ can rather be seen as functions $\varphi(x)$ of argument x; for each particular value $X \in C^*$ received by the argument x, the function takes as the value the constant word $\varphi(X)$. Now, a word equation in one variable is a pair $E = (\varphi(x), \psi(x))$ of words over $C \cup \{x\}$, and, any constant word $X \in C^*$ such that $\varphi(X) = \psi(X)$ is a *solution* of the equation E. We denote by $Sol(E)$ the set of all solutions of the equation E. Needless to say, it is this set which interests us here.

A very first theoretical question one is naturally brought to ask is whether or not the existence of a solution for a word equation in one variable is decidable. As we know, the solvability of equations in words was one of the chief preoccupations of the Russian logical school in the sixties and their effort has been crowned with success by the famous paper by G. S. Makanin [6], considered by many as one of the most beautiful results of theoretical computer science. Against expectation,

and in contrast to the situation with the Diophantine equations, he has proved the question of solvability of a general equation in words decidable.

The case of equations in one variable had been settled before by Khmelevskiĭ [3]. The cornerstone of his little theory is the existence of a constant χ proportional to the length $|E|$ of the equation, $|E| = |\varphi(x)| + |\psi(x)|$, such that $Sol(E) \neq \emptyset$ if and only if there is a solution $X \in Sol(E)$ of length $|X| \leq \chi$. For all practical pourposes we can take χ to be equal to $4|E|$. Therefore, the problem of solvability of the equation E in one variable has been reduced to the search of a solution in a finite set of candidates for solution, all the words over C of length $\leq 4|E|$. Nobody cared too much, in the sixties, about the procedure of decision being of exponential time complexity.

In fact it is an another key observation due to Khmelevskiĭ which permits to reduce drastically the number of candidates for solution, namely, the periodic form of any solution, determined by the coefficients of the equation. An obvious necessary condition of solvability of E is that it can be brought by cancellation to the following form:

$$A_0 x A_1 x \dots x A_r = x B_1 x \dots x B_s, \tag{1}$$

with $A_0, \dots, A_r, B_1, \dots, B_s \in C^*$, $A_0 \neq \varepsilon$, This cancelled form (1) of the equation makes it obvious that the leading coefficient A_0 must be a period of any solution, that is to say, any solution X of E must be a prefix of some long enough power of A_0. Taking the least integer k with $|A_0^k| \geq 4|E|$, we know now that E is solvable if and only if one of the prefixes of A_0^k is a solution. We have to test a linear number, in terms of the length of the equation, of candidates $X \in Prefix(A_0^k)$. The length of all these candidates also being linear, a single test carried out naïvely takes $O(|E|^2)$ comparisons to check whether or not $\varphi(X)$ is equal to $\psi(X)$. In this way, the decision procedure for solvability becomes cubic at practically no cost.

Further improvement of the procedure comes from a simple observation that when testing the equality $\varphi(X) = \psi(X)$ we need not compare the occurrences of X in $\varphi(X)$ against the occurrences of X in $\psi(X)$ if we precalculate all possible overlaps of X with itself. Such a precalculation can be done in $O(|X|)$ by the algorithm of Morris and Pratt. The test based on this precalculation only compares coefficients against coefficients, which makes the number of comparisons linear in the length of the equation. For a candidate X of linear length, therefore, the test is linear, hence the solvability decision becomes quadratic [7].

It should be said that the solutions of E which are prefixes of A_0^k determine, in a very straightforward way, all the solutions $Sol(E)$, so the above quadratic procedure is also an algorithm of resolution.

Note that in the meanwhile our preoccupation has completely changed: it is no longer the decision 'in principle' of solvability but the actual resolution of the equation, and at as little a cost as possible. We subscribe, so to say, to the research project announced by W. Charatonik and L. Pacholski [1] aiming 'to describe classes of word equations for which fast algorithms, deciding solvability or giving actual solutions, exist'. By 'fast' they mean 'deterministic polynomial

time'. And they add that 'of course, for many actual applications it would be better to consider more restricted classes like linear time or DTIME($|E|\log|E|$)'.

In the present note we establish that the word equations in one variable constitute a complexity class they call for. Innocently as they may look, these equation provide a nice testing ground for subtle methods developed for string matching. Of course, it would be even nicer if we could announce that our $O(|E|\log|E|)$ algorithm of resolution is optimal; which we cannot. We leave it as a challenging problem whose precise formulation is given at the end.

2 Solutions of a Given Form

In this section we show how to find, for a given word equation E in one variable, all the solutions of the form $X_n = (uv)^n u, \quad n \geq 0$, with uv a primitive word over C. This particular form of solution is called a *skew power* of uv and is determined by the factorization of uv into the pair (u, v). The task now consists in determining the set of values of the integer parameter n for which the word $X_n = (uv)^n u$ is a solution of E. Before showing how to go about it let us recall some basic facts about primitive words and conjugacy which can be found in [2,5]:

1. Two words A and B are *conjugate* if $A = uv$ and $B = vu$ for some pair of words u, v. A non void word P is *primitive* if it has $|P|$ distinct conjugates. Put otherwise, a primitive word P has exactly two occurrences in its square PP: one as a prefix and the other one as a suffix. This is a very useful 'synchronizing' property: a power of P can occur in an another power of P only at certain positions, necessarily prefixed by a power of P.
2. Every non void word A is a power of a unique primitive word P, the *primitive root* of A. Two words A and B are conjugate if and only if their primitive roots are conjugate. If two primitive words P and Q are conjugate then the pair of words u, v with $P = uv$ and $Q = vu$ is unique.
3. Let u, v, w be distinct primitive words such that u^2 is a prefix of v^2 and v^2 is a prefix of w^2. Then $|u| + |v| \leq |w|$. Consequently, a word of length n can have at most $\log n$ distinct prefixes which are squares of primitive words, because the lengths of such primitive words will grow at least as fast as the Fibonacci numbers, that is to say, exponentially, with the golden ratio $(1+\sqrt{5})/2$ for base. We say briefly that the number of repetitive primitive prefixes is at most logarithmic.

As for the equation E, we will deal only with the cancelled form (1) of it and, moreover, we will suppose it to be balanced, $r = s \neq 0$. Because if not then only a prefix X of a suitable power of A_0 whose length satisfies

$$|A_0 X A_1 X \ldots X A_r| = |X B_1 X \ldots X B_s|$$

can possibly be a solution. The choice would thus be immediately narrowed to at most one candidate X of length $|X| \leq |E|$.

Proposition 1 *Let $u, v \in C^*$ be such that uv is primitive, of length $|uv| \leq |A_0B_1|$. Then it is possible to determine in an $O(|E|)$ time all integers $n \geq 0$ such that $X = (uv)^n u$ is a solution of E.*

Proof: For an arbitrary word $w \in C^*$, let us define the integer

$$w/uv = max\{k; (uv)^k u \in Prefix(w)\}$$

It is not difficult to calculate $\varphi(X_n)/uv$ and $\psi(X_n)/uv$ for $n \geq 1$. Indeed, if uv is not the primitive root of A_0 then $\varphi(X_n)/uv = A_0uv/uv$, because of the synchronizing property of uv. In the opposite case, if $A_0X_nA_1X_n \ldots A_kX_n$ with $k \geq 1$ is a prefix of a power of uv then each occurrence of X_n in the above word must be prefixed by a power of uv, because of the synchronizing property of the prefix uv of X_n (recall that $n \geq 1$). This leaves us with the following form of the coefficients in the left-hand term $\varphi(x)$ of our equation E:

$$A_0 = (uv)^{t_0},\ A_1 = (vu)^{t_1}v,\ \ldots,\ A_{p-1} = (vu)^{t_{p-1}}v,\ A_p = (vu)^{t_p}A'_p$$

with $A'_p \neq v$ and $vu \notin Prefix(A'_p)$.
Then we have

$$\varphi(X_n) = \alpha(p) + p(n+1) + \sum_{i=0}^{p} t_i$$

where

$$\alpha(p) = \begin{cases} 0 \text{ if } p \neq r \text{ and } vu \in Prefix(A'_puv) \\ -1 \text{ otherwise} \end{cases}$$

Similarly, if we have

$$B_1 = (vu)^{s_1}v,\ \ldots,\ B_{q-1} = (vu)^{s_{q-1}}v,\ B_q = (vu)^{s_q}B'_q$$

with $B'_q \neq v$ and $vu \notin Prefix(B'_q)$, then

$$\psi(X_n) = \beta(q) + q(n+1) + \sum_{j=1}^{q} s_j$$

where

$$\beta(q) = \begin{cases} 0 \text{ if } q \neq r \text{ and } vu \in Prefix(B'_puv) \\ -1 \text{ otherwise} \end{cases}$$

The calculation of $\varphi(X_n)/uv$ and $\psi(X_n)/uv$ thus consists in successive linear time examination of the coefficients until the first one is found which is not of the desired skew power form (except for A_0 whose treatment is slightly different from the rest of the coefficients), therefore the time it takes is proportional to $|A_0A_1 \ldots A_pB_1 \ldots B_q|$.

If $p \neq q$ then the resulting Diophantine equation $\varphi(X_n)/uv = \psi(X_n)/uv$ has at most one integer solution $n \geq 0$. If such a solution exists, then it determines a single candidate X_n of length $|X_n| \leq |E|$ for which a linear time test will decide whether or not it is a solution of E.

If $p = q$ then the unknown n disappears from the Diophantine equation. If the latter is not contradictory, then for every $n \geq 0$, $X_n = (uv)^n u$ is a solution of E if and only if X_n is a solution of the reduced equation E':

$$A''_p x A_{p+1} x \dots x A_r = B''_p x \dots x B_r$$

with the coefficients A''_p and B''_p determined according to the following three cases:

$$A''_p = A'_p \text{ and } B''_p = vuB'_p \text{ if } \alpha(p) = 0 \text{ and } \beta(p) = -1,$$

$$A''_p = A'_p \text{ and } B''_p = B'_p \text{ if } \alpha(p) = \beta(p),$$

$$A''_p = vuA'_p \text{ and } B''_p = B'_p \text{ if } \alpha(p) = -1 \text{ and } \beta(p) = 0.$$

Reasoning by induction on the number of occurrences of x, we can suppose that all the solutions of the prescribed form of the reduced equation E' can be found in $O(|E'|)$ time. The time of solving E is thus the time of the reduction plus the time of solving the reduced equation, which makes $O(|E|)$.

Finally, we must not forget about the solitary candidate $X_0 = u$, which must also be put to a test for solution of E. □

3 Determining the Forms of Solution

Proposition 2 *There is $O(\log |E|)$ pairs (u, v) of words over C such that uv is primitive of length $|uv| \leq |A_0 B_1|$ and every solution of E is of the form $(uv)^n u$, $n \geq 0$, for one of these pairs. Moreover, all these pairs (u, v) can be found in an $O(|E| \log |E|)$ time.*

Proof: Assume that $|A_0| \leq |B_1|$ and denote by B_0 the prefix of B_1 of length $|B_0| = |A_0|$. Then any solution $X \in Sol(E)$ must conjugate A_0 and B_0. Therefore, the primitive roots of A_0 and B_0 are conjugate and equal, respectively, to uv and vu for a unique pair of words (u, v). All solutions of E are of the form $(uv)^n u$ for this unique pair (u, v) and thus can be determined in an $O(|E|)$ time.

Assume next that $|A_0| > |B_1|$. Then any solution X of length $|X| \geq |A_0| - |B_1|$ will conjugate A_0 to $B_1 P$, where P is the prefix of A_0 of length $|P| = |A_0| - |B_1|$, hence the conclusion about the form of such solutions and the time for finding them is the same as above. On the other hand, any solution X of length $|X| \leq |A_0| - |B_1|$ determines a square prefix $B_1 X B_1 X$ of $B_1 A_0 A_0$. The square PP of the primitive root P of $B_1 X$ appears as a prefix of $B_1 A_0 A_0$. Moreover, P determines uniquely the form of X, because there is a unique factorization $P = vu$ such that $B_1 = (vu)^m v$ and $X = (uv)^n u$ for some $m, n \geq 0$. As we know, the number of such primitive repetitive prefixes P is logarithmic in $l = |B_1 A_0 A_0|$. M. Crochemore [2] gives us a method permitting to find all of them in an $O(l \log l)$ time. □

4 Conclusion

The two propositions we have proved serve in an obvious way as a theoretical basis of an algorithm which puts a logarithmic number of solution forms through a linear procedure of selection of candidates, and, the selected candidates to a final linear test for solution. It is not our aim here to write down a concrete implementation of this algorithm. Instead, we would like to formulate a problem whose solution may pave the way towards either a proof of optimality of the given algorithm or plainly to a linear algorithm:
Does there really exist a class of equations of unbounded length such that each equation E in the class has $O(\log |E|)$ solutions of distinct solution forms?
In conclusion, we would like to express our thanks to J. Néraud who kindly explained to us his method of matching one variable patterns [8], thereby putting us on the right way. Finally, the credit for converting us to word equations goes to H. Abdulrab and J.-P. Pécuchet. Also, the recent four months stay of G. S. Makanin in the INSA of Rouen was a powerful spell of inspiration for all of us.

References

1. Charatonik W. and L. Pacholski, Word Equations With Two Variables, *Lecture Notes in Comp. Sci.* 677, Springer-Verlag, Proc. of the Second International Workshop on Word Equations and Related Topics IWWERT'91, Rouen, France, 1991, H. Abdulrab and J.P. Pecuchet (Eds.), 43–57.
2. Crochemore M., An optimal algorithm for computing the repetitions in a word, *Information Proc. Letters* 12(1981), 244–250.
3. Khmelevskiĭ Yu. I., Equations in a Free Semigroup (in Russian), *Trudy Matem. Inst. Steklova*, 107(1971), 1–284.
4. Kościelski A. and L. Pacholski, Complexity of Makanin's Algorithms, *Journal of ACM, to appear.*
5. Lothaire M., Combinatorics on Words, *Encyclopedia of Math. and Appl.*, Addison Wesley, 1983.
6. Makanin G. S., The Problem of Solvability of Equations in a Free Semigroup (in Russian), *Matematicheskiĭ Sbornik* 103(1977), 147–236. English translation in *Math. USSR Sbornik* 32(1977), 129–198.
7. Maksimenko M., Algorithme quadratique de calcul de la solution générale d'équations en mots à une variable, *RAIRO, Submitted.*
8. Néraud J., New Algorithms for Detecting Morphic Images of a Word, *Lecture Notes in Comp. Sci.* 711, Springer Verlag, Proc. of the 18th International Symposium MFCS'93, Gdańsk, Poland, A. M. Borzyszkowski and S Sokolowski (Eds.), 588–597.