Equations in the Partial Semigroup of Words with Overlapping Products*

Mari Huova and Juhani Karhumäki

Department of Mathematics and TUCS,
University of Turku, 20014 Turku, Finland
{mahuov,karhumak}@utu.fi

Abstract. We consider an overlapping product of words as a partial operation where the product of two words is defined when the former ends with the same letter as the latter starts, and in this case the product is obtained by merging these two occurrences of letters, for example $aba \bullet ab = abab$. Some basic results on equations of words are established by reducing them to corresponding results of ordinary word equations.

Keywords: combinatorics on words, overlapping product, equations.

1 Introduction

Motivated by bio-operations, or more formally, DNA computing, see [18], we consider an operation of overlapping product of words defined as follows. For two words ua and bv, with a and b letters, we define their overlapping product

$$ua \bullet bv = \begin{cases} uav & \text{if } a = b \ , \\ \text{undefined} & \text{if } a \neq b \ . \end{cases}$$

Consequently, the operation is locally controlled, and clearly a (partial) associative operation on the set of nonempty words Σ^+.

Recently the descriptional complexity of this operation was analyzed in the case of regular languages, see [10]. The same operation and its extensions have been studied in a number of articles motivated by bio-operations in DNA strands, see, e.g., [4], [5], [6], [7], [11], [16] and [17]. We consider this operation in connection with word equations. It turns out that many questions on equations can be transformed, and finally solved, by translating these to related problems on ordinary word equations. The translation is made because, for example, the simple operation of cancellation does not work. Thus, for example, $x \bullet y = x \bullet z \bullet x$ is not equal to $y = z \bullet x$, as explained more closely in Section 3.

More concretely, we solve a few basic equations over overlapping product, introduce a general translation of such equations to a Boolean system of ordinary equations, and as a consequence establish, e.g., that the fundamental result of solvability of the satisfiability problem extends to these new types of equations.

* Supported by the Academy of Finland under the grant 121419.

H. Bordihn, M. Kutrib, and B. Truthe (Eds.): Dassow Festschrift 2012, LNCS 7300, pp. 99–110, 2012.

2 Preliminaries

Let Σ be a finite alphabet. We denote by Σ^+ the set of all nonempty words over Σ and view it as the free semigroup with respect to the product of words. Notation Σ^* is also used refering to the monoid $\Sigma^+ \cup \{1\}$, where 1 denotes the empty word. As general references of the combinatorics on words we refer to [13] and [9].

We define a new *partial* binary operation, so-called *overlapping product*, on Σ^+ as follows: For two words ua and bv, with $a, b \in \Sigma$,we set

$$ua \bullet bv = \begin{cases} uav & \text{if } a = b \ , \\ \text{undefined} & \text{if } a \neq b \ . \end{cases}$$

Clearly, the operation $\bullet$ is associative (partial) operation so that we have

Fact 1. $(\Sigma^+, \bullet)$ is a partial semigroup.

Actually, in $(\Sigma^+, \bullet)$ letters (considered as words of length 1) constitute partial (nonunique) left and right units. Indeed, $a \bullet u$ with any $u \in \Sigma^+$, is equal to u if defined.

Due to the associativity it is justified to write the product without parenthesis:

$$\alpha = \alpha_1 \bullet \alpha_2 \bullet \cdots \bullet \alpha_n \ , \quad \text{for any } \alpha_i \in \Sigma^+. \tag{1}$$

The word α, if defined, as an element of Σ^+ is deduced from (1) as follows. We need one additional notation. For any word $u = a_1 a_2 \cdots a_k$, with $a_i \in \Sigma$, the notation $u(a_k)^{-1}$ refers to the word $a_1 a_2 \cdots a_{k-1}$ and correspondingly $(a_1)^{-1}u$ to the word $a_2 a_3 \cdots a_k$. In order for α to be defined, for each $i = 1, \ldots, n-1$, necessarily

$$\text{last } \alpha_i = \text{first } \alpha_{i+1} \ ,$$

and then

$$\begin{aligned} \alpha &= \alpha_1 (\text{last } \alpha_1)^{-1} \alpha_2 (\text{last } \alpha_2)^{-1} \cdots \alpha_{n-1} (\text{last } \alpha_{n-1})^{-1} \alpha_n \\ &= \alpha_1 (\text{first } \alpha_2)^{-1} \alpha_2 (\text{first } \alpha_3)^{-1} \cdots \alpha_{n-1} (\text{first } \alpha_n)^{-1} \alpha_n \ . \end{aligned}$$

On the other hand any word

$$\alpha = \alpha_1 \alpha_2 \cdots \alpha_n \ \text{ with } \alpha_i \in \Sigma^+$$

can be written as an element of the partial semigroup $(\Sigma^+, \bullet)$ as follows:

$$\alpha = \alpha_1 (\text{first } \alpha_2) \bullet \alpha_2 (\text{first } \alpha_3) \bullet \cdots \bullet \alpha_{n-1} (\text{first } \alpha_n) \bullet \alpha_n \ .$$

It is worth noting that the latter translation is always defined.

These considerations make our goal to consider the theory of word equations over the overlapping product feasible - as described in details in Section 4.

3 Examples of Basic Equations

Common tools for solving word equations such as Levi's Lemma, splitting of equation and length argument are not so straightforward to use with equations containing overlapping products. Problems for using these tools arise from the facts that for overlapping products to be defined the last and the first letters of the adjacent factors have to coincide and when a product is conducted these two letters are unified to a single letter. For example, the first of these reasons causes the following problem.

Example 1. Consider an equation $x \bullet y = x \bullet z \bullet x$ with overlapping products and an equation $xy = xzx$. Equation $xy = xzx$ can be reduced into the form $y = zx$, accordingly we could suppose that $x \bullet y = x \bullet z \bullet x$ equals with equation $y = z \bullet x$. However, for example, $y = abb, z = ab, x = bb$ is a solution for $y = z \bullet x$ but not for the original equation because the overlapping product $x \bullet y = bb \bullet abb$ is not defined.

Example 1 shows that we cannot use Levi's Lemma straightforwardly to eliminate the leftmost or the rightmost unknowns. The same problem arises if we split an equation. Again we may loose the information of the requirements that originated from the overlapping product that was located at the point of splitting.

Unification of the last and the first letters of the adjacent factors complicates the use of length argument. The total length of an expression containing overlapping products depends on the lengths of the factors and on the number of factors, i.e. $|x_1 \bullet \cdots \bullet x_k| = |x_1| + \cdots + |x_k| - (k-1)$. For some equations it may, nevertheless, be easy to detect, for example, the middle of both sides as for example in the equation $x \bullet y \bullet y \bullet x = z \bullet z$. From this we can conclude that $x \bullet y = z, y \bullet x = z$, but the consequences of splitting the equation have to be taken into account.

We proceed by solving some basic equations over the partial semigroup with overlapping product. First we consider the equation $x \bullet y = y \bullet x$, which corresponds to *commutation.*

Example 2. To solve the equation $x \bullet y = y \bullet x$ we first assume that $|x|, |y| > 1$. For the overlapping product to be defined we can assume that $x = ax'a$ and $y = ay'a$, where $a \in \Sigma$ and $x', y' \in \Sigma^*$. Now we can reduce the equation $x \bullet y = y \bullet x$ into an ordinary word equation $x \bullet y = ax'a \bullet ay'a = ax'ay'a = ay'ax'a = ay'a \bullet ax'a = y \bullet x$. From the equation $ax'ay'a = ay'ax'a$ we can notice that $ax'ay' = ay'ax'$, and hence ax' and ay' commute. Now we can write $ax' = t^i$ and $ay' = t^j$, where $t = a\alpha$ with $\alpha \in \Sigma^*$ and $i, j > 0$. From this we get $x = ax'a = t^i a = (a\alpha)^i a$ and $y = ay'a = t^j a = (a\alpha)^j a$, where $a \in \Sigma, \alpha \in \Sigma^*$ and $i, j > 0$. In the case that $|x| = 1$ (resp. $|y| = 1$) we have $x = a$ (resp. $y = a$), with $a \in \Sigma$ and $y = a\alpha a$ or $y = a$ (resp. $x = a\alpha a$ or $x = a$), with $\alpha \in \Sigma^*$. Thus the equation $x \bullet y = y \bullet x$ has solutions

$$\begin{cases} x = (a\alpha)^i a \\ y = (a\alpha)^j a \end{cases}, \quad \text{where } a \in \Sigma, \alpha \in \Sigma^* \text{ and } i, j \geq 0.$$

We remark that the answer of the equation of the previous example could also be written with the help of the overlapping product. For example, if $x = (a\alpha)^2 a, y = (a\alpha)^3 a$ we could also write $x = (a\alpha a) \bullet (a\alpha a), y = (a\alpha a) \bullet (a\alpha a) \bullet (a\alpha a)$. Thus, the words that are solutions of this equation refering to commutation are, in fact, overlapping products of words of the form $a\alpha a$ or letters as a special case.

The second equation we will examine is associated with *conjugation*, i.e. $xz = zy$.

Example 3. We first check two special cases for equation $x \bullet z = z \bullet y$. If $x = a$, with $a \in \Sigma$, then $y = b$, $b \in \Sigma$, and $z = a\alpha b$, $\alpha \in \Sigma^*$, or if $a = b$, then $z = a$ is possible, too. If $x = aa$, with $a \in \Sigma$, then $y = aa$ and $z = a^i$, where $i > 0$.

Now we can assume that $|x|, |y|, |z| > 2$ in equation $x \bullet z = z \bullet y$. As in Example 2 we may assume $x = ax'a$, $y = by'b$ and $z = az'b$, where $a, b \in \Sigma$ and $x', y', z' \in \Sigma^+$. These assumptions are due to the facts that overlapping products have to be defined and x and z have a common first letter and y has a common last letter with z. Reduction now gives now $x \bullet z = ax'az'b = az'by'b = z \bullet y$. From the word equation $x'az' = z'by'$ we can conclude that $x'a$ and by' conjugate. The conjugation property gives that there exist $p, q' \in \Sigma^*$ so that $x'a = pq'$, $by' = q'p$ and $z' = p(q'p)^i$, where $i \geq 0$ and in addition if $q' \neq 1$ then $q' = bqa$ with $q \in \Sigma^*$. Now with these assumptions we have a solution

$$\begin{cases} x = ax'a = apq' = apbqa \\ y = by'b = q'pb = bqapb \\ z = az'b = ap(q'p)^i b = ap(bqap)^i b \end{cases},$$

where $a, b \in \Sigma, p, q \in \Sigma^*$ and $i \geq 0$.

If $q' = 1$ then $p = bp'a$, where $p' \in \Sigma^*$ and solutions are of the form

$$\begin{cases} x = ax'a = ap = abp'a \\ y = by'b = pb = bp'ab \\ z = az'b = a(p)^{i+1} b = a(bp'a)^{i+1} b \end{cases},$$

where $a, b \in \Sigma, p, p' \in \Sigma^*$ and $i \geq 0$.

In fact, these latter solutions are included in the upper formula. Thus equation $x \bullet z = z \bullet y$ has solutions

$$\begin{cases} x = apbqa \\ y = bqapb \\ z = ap(bqap)^i b \end{cases}, \text{ where } a, b \in \Sigma, p, q \in \Sigma^* \text{ and } i \geq 0$$

and special solutions

$$\begin{cases} x = a \\ y = b \\ z = a\alpha b \end{cases}, \begin{cases} x = a \\ y = a \\ z = a^i \end{cases} \text{ and } \begin{cases} x = aa \\ y = aa \\ z = a^i \end{cases},$$

where $a, b \in \Sigma, \alpha \in \Sigma^*$, $i > 0$.

The third basic equation we consider asks when the product of two squares is a square, a problem first studied in [14]. In the case of word equation $x^2y^2 = z^2$ the answer is that the equation has only periodic solutions. If we consider the equation with overlapping product we get a corresponding result.

Example 4. We first assume that $|x|, |y|, |z| > 1$ in the equation $x \bullet x \bullet y \bullet y = z \bullet z$. Because overlapping products have to be defined we can again assume that $x = ax'a$, $y = ay'a$ and $z = az'a$, where $a \in \Sigma$ and $x', y', z' \in \Sigma^*$. Reduction of overlapping products into usual word products gives an equation $ax'ax'ay'ay'a = az'az'a$ from which we get a simpler equation $x'ax'ay'ay' = z'az'$. The length argument gives now that $|x'ay'| = |z'|$, and thus by comparing the beginnings and the ends of both sides on the equation $x'ax'ay'ay' = z'az'$ we conclude $z' = x'ay'$. Now the equation has the form $x'ax'ay'ay' = x'ay'ax'ay'$ which leads to an equation $ax'ay' = ay'ax'$ showing that ax' and ay' commute. From this observation we can conclude that $ax' = t^i$ and $ay' = t^j$ with $t = a\alpha$, $\alpha \in \Sigma^*$ and $i, j > 0$ and hence $x = ax'a = (a\alpha)^i a$, $y = ay'a = (a\alpha)^j a$ and $z = az'a = ax'ay'a = (a\alpha)^{i+j} a$. Again if some of the unknowns equal a letter, then the solution is gained from the following general formula by allowing $i, j \geq 0$. The equation $x \bullet x \bullet y \bullet y = z \bullet z$ has solutions

$$\begin{cases} x = (a\alpha)^i a \\ y = (a\alpha)^j a \\ z = (a\alpha)^{i+j} a \end{cases}, \quad \text{where } a \in \Sigma, \alpha \in \Sigma^* \text{ and } i, j \geq 0.$$

We yet give one example of a basic equation which leads us to analyze the defect property.

Example 5. To solve an equation $x \bullet y = u \bullet v$ we may assume $x = x'a$, $y = ay'$, $u = u'b$ and $v = bv'$ where $a, b \in \Sigma$ and $x', y', u', v' \in \Sigma^*$. With these assumptions we have an ordinary word equation $x'ay' = u'bv'$. We consider only the case $|x'| < |u'|$, the case $|u'| < |x'|$ is symmetric and $|x'| = |u'|$ is clear. The equation $x'ay' = u'bv'$ has now a solution $x' = \alpha$, $y' = \beta b\gamma$, $u' = \alpha a\beta$ and $v' = \gamma$ where $\alpha, \beta, \gamma \in \Sigma^*$. The solution for the original equation with the assumption $|x| < |u|$ can now be given:

$$\begin{cases} x = \alpha a \\ y = a\beta b\gamma \\ u = \alpha a\beta b \\ v = b\gamma \end{cases}, \quad \text{where } a, b \in \Sigma, \alpha, \beta, \gamma \in \Sigma^*.$$

We remark that these four words x, y, u and v of the previous example can be expressed in the form $x = \alpha a$, $y = a\beta b \bullet b\gamma$, $u = \alpha a \bullet a\beta b$ and $v = b\gamma$, thus they can be formed from three words by overlapping product. This implies, as stated in the next theorem, that a so-called *defect property*, see [3], is also valid in $(\Sigma^+, \bullet)$.

Theorem 1. *Let X be a set of n words with $X \cap \Sigma = \emptyset$. If X satisfies a nontrivial equation with overlapping products, then these words can be expressed with $n - 1$ words by using overlapping products.*

Proof. Let $x_1 \bullet x_2 \bullet \cdots \bullet x_k = y_1 \bullet y_2 \bullet \cdots \bullet y_l$ be a nontrivial equation such that $x_i, y_j \in X$ for all $i = 1, \ldots, k$ and $j = 1, \ldots, l$. We may assume that $|x_1| < |y_1|$ and hence y_1 can be written in the form $y_1 = x_1 \bullet (\text{last } x_1)\, y_1'$. Thus, the words of the set X can be expressed with words $X_1 = (X - \{y_1\}) \cup \{(\text{last } x_1)\, y_1'\}$. The number of words in X_1 is clearly at most n and $X_1 \cap \Sigma = \emptyset$. Now the equation corresponding to the original equation can be reduced at least from the beginning with a factor x_1 and hence, the new (nontrivial) equation will be shorter in terms of the total length of an expression which is given by $|x_1 \bullet \cdots \bullet x_k| = |x_1| + \cdots + |x_k| - (k-1)$. We divide the analyzis into two cases.

Case 1. Inductively with respect to the length of the nontrivial equation we will proceed into an equation $u = v_1 \bullet \cdots \bullet v_m$ with $u, v_1, \ldots v_m$ words from the processed set of at most n words. Now it is clear that the word u may be removed from the set and the original words can be expressed with $n-1$ words as claimed.

Case 2. If in some point of the procedure described above the equation will reduce into a trivial equation, the constructed set of words corresponding to that situation contains already at most $n-1$ words. This follows from the fact that the reduction from a nontrivial equation into a trivial equation is possible only if some factor replacing an old word already exists in the considered set of words. □

As a conclusion, the above examples and the theorem show that results for word equations over overlapping product are often similar, but not exactly the same, as in the case of ordinary word equations. Moreover, the proofs reduce to that of ordinary words - as further explained in the next section.

4 Reduction into Word Equations

In this section the reduction of equations over overlapping products to that of ordinary word equations is analyzed in general. The reduction leads to a Boolean combination of word equations, as we shall see in the next result.

Theorem 2. *Let Σ be a finite alphabet, X be the set of unknowns and $e : u = v$ be an equation over X with overlapping products. Then the equation e can be reduced into a Boolean combination of ordinary word equations.*

Proof. Consider the equation $u = x_1 \bullet x_2 \bullet \cdots \bullet x_l = y_1 \bullet y_2 \bullet \cdots \bullet y_m = v$, where $x_i, y_j \in X$ for all $i = 1, \ldots, l$ and $j = 1, \ldots, m$.

Part 1. Assume that the solutions u_i for x_i and v_j for y_j have $|u_i|, |v_j| > 1$, for all $i = 1, \ldots, l$ and $j = 1, \ldots, m$, and hence we can mark the first and the last letters of the words and write

$$x_1 = a_1 x_1' a_2 \ , \ \ x_2 = a_2 x_2' a_3 \ , \ \ \ldots \ , \ \ x_l = a_l x_l' a_{l+1} \ ,$$

$$y_1 = b_1 y_1' b_2 \ , \ \ y_2 = b_2 y_2' b_3 \ , \ \ \ldots \ , \ \ y_m = b_m y_m' b_{m+1} \ ,$$

where $a_i, b_j \in \Sigma$ and x_i' and y_j' are new unknowns from the set X'.

Now we have some restrictions for choosing the letters a_i, b_j. If $x_i = x_j$ then $a_i = a_j$ and $a_{i+1} = a_{j+1}$, and similarly if $y_i = y_j$, then $b_i = b_j$ and $b_{i+1} = b_{j+1}$. Comparing unknowns of the equation e on both sides we have that if $x_i = y_j$, then $a_i = b_j$ and $a_{i+1} = b_{j+1}$, and in addition, $a_1 = b_1$ and $a_{l+1} = b_{m+1}$ always hold.

With these assumptions and markings we have a reduced word equation $e' : u' = v'$ without overlapping products where u' and v' are defined as follows:

$$u = x_1 \bullet x_2 \bullet \cdots \bullet x_l = a_1 x_1' a_2 x_2' a_3 \cdots a_l x_l' a_{l+1} = u'$$

$$v = y_1 \bullet y_2 \bullet \cdots \bullet y_m = b_1 y_1' b_2 y_2' b_3 \cdots b_m y_m' b_{m+1} = v' \ .$$

In fact, to solve the original equation e we have to solve the reduced equation e' with all possible combinations of values for letters a_i and b_j from the set Σ. In other words, the set of solutions of the original equation $u = v$ equals the set of solutions of a Boolean set of equations which is a disjunction of equations without overlapping products.

Part 2. In Part 1 we assumed that each unknown corresponds to a word of length at least two. Now we assume that at least one of the unknowns corresponds to a letter. We proceed as in Part 1 but with a bit different markings. Let $x_i = a_{i,1} x_i' a_{i,2}$ or $x_i = a_{i,12}$, with $a_{i,1}, a_{i,2}, a_{i,12} \in \Sigma$, depending on the length of the solution corresponding to x_i. Because overlapping products have to be defined we have $a_{i,2} = a_{i+1,1}$ or $a_{i,2} = a_{i+1,12}$ and $a_{i,12} = a_{i+1,1}$ or $a_{i,12} = a_{i+1,12}$. We process similarly with y_j's and b's. As in Part 1, we have some apparent additional restrictions for letters a's and b's depending on equation e. With these assumptions and markings we can again form a corresponding reduced word equation $e' : u' = v'$ without overlapping products.

To solve the original equation with assumptions of Part 2 we have again a Boolean combination of word equations to solve. This set is a disjunction of equations of the form e' with all possible combinations such that at least one unknown corresponds to a letter and values of corresponding a's and b's vary in the set Σ.

Part 3. In Part 1 and Part 2 we have only discussed the cases of constant free equations. If some factors in the equation $u = x_1 \bullet x_2 \bullet \cdots \bullet x_l = y_1 \bullet y_2 \bullet \cdots \bullet y_m = v$ are constants we proceed as previously in Parts 1 and 2 but with the additional knowledge of constants. If, for example, x_i is a constant in e and we have marked $x_i = a_i x_i' a_{i+1}$ we treat a_i, a_{i+1} and x_i' in equation e' as constants, too.

As a conclusion we remark that the considered Boolean sets are finite and the set of solutions of the original equation e is the set of solutions of a disjunction of Boolean sets of Part 1 and Part 2, the observations of the third part taken into account if necessary. Equations in this combined Boolean set do not contain overlapping products, and this proves the claim. □

We remark that regardless of equation e having constants or not the equations in the constructed Boolean set have constants because the given reduction takes into consideration the fact that overlapping products have to be defined. The property that the overlapping product is only partially defined also makes the conversion of equations to the other direction difficult. As mentioned in Section 2 it is easy to write a word as the element of this partial semigroup $(\Sigma^+, \bullet)$. But if we try to convert, for example, an equation $xy = z$ we cannot just write $x \bullet y = z$. Instead, the equation $x \bullet y' = z$ with requirements $y' = ay, x = x'a$, with $a \in \Sigma$, would correspond the original equation.

5 Consequences of the Reduction

It is known that any Boolean combination of word equations can be transformed into a single equation, see [12], [3] or [2] as the original source. Another well known result concerning word equations is the satisfiability problem, that is decidability of whether a word equation has a solution or not. The satisfiability problem is shown to be decidable by Makanin [15], see also [19]. We will show that corresponding results are also valid for equations with overlapping products.

Theorem 3. *For any Boolean combination of equations with overlapping products we can construct a single equation without overlapping products such that the sets of solutions of the Boolean combination and the single equation are equal when restricted to unknowns of the original equations.*

Proof. The result of the previous section shows that an equation with overlapping products can be reduced into a Boolean combination of usual word equations. From this it follows that any Boolean combination of equations with overlapping products can be reduced into another Boolean combination of ordinary word equations. This, in turn, as stated above can be transformed into a single equation without overlapping products. □

We remind that combining a conjunction of two word equations into a single equation does not require any extra unknowns but in a case of disjunction two additional unknowns are required in the construction given in [12], see also [3]. Thus, the single equation constructed from the Boolean combination of equations is likely to contain many more unknowns than the original equations because of the disjunctions derived from the reduction method.

We next slightly modificate this old proof for the result of [12] concerning a disjunction of two equations. The new result shows that, in fact, two additional unknowns are enough to combine a disjunction of a finite set of equations into a single equation.

Theorem 4. *Let $e_1 : u_1 = v_1, \ldots, e_n : u_n = v_n$ be a finite set of equations. A disjunction of these equations, i.e. the property expressible by e_1 or e_2 or $\ldots$ or e_n, can be transformed into a single equation with only two additional unknowns.*

Proof. We may assume that the right hand sides of the equations are the same because the disjunctions of the equations of the following two sets S_1 and S_2 are equivalent:

$$S_1: \begin{array}{c} u_1 = v_1 \\ u_2 = v_2 \\ \vdots \\ u_n = v_n \end{array} \quad \text{and} \quad S_2: \begin{array}{c} u_1 v_2 v_3 \cdots v_n = v_1 v_2 \cdots v_n \\ v_1 u_2 v_3 \cdots v_n = v_1 v_2 \cdots v_n \\ \vdots \\ v_1 v_2 \cdots v_{n-1} u_n = v_1 v_2 \cdots v_n \end{array} .$$

Thus, we may assume that $v_1 = v_2 = \cdots = v_n = v$ holds for equations $e_1, \ldots, e_n$.

To complete the proof we will outline the necessary constructions, the justifications can be deduced as in [12]. First we define a function $\langle\ \rangle$ by

$$\langle \alpha \rangle = \alpha a \alpha b \ , \quad \text{where } a, b \in \Sigma,\ a \neq b.$$

We will use the properties that for each α the shortest period of $\langle \alpha \rangle$ is longer than half of its length and $\langle \alpha \rangle$ is primitive. We remark that now $\langle \alpha \rangle$ can occur in $\langle \alpha \rangle^2$ only as a prefix and a suffix. Let us denote $u_1 \cdots u_n = u$. With these observations we may deduce that

$$u_1 = v \text{ or } u_2 = v \text{ or } \cdots \text{ or } u_n = v \quad \Leftrightarrow \quad \exists Z, Z' : \ X = ZYZ' \ ,$$

where

$$Y = \langle u \rangle^2 v \langle u \rangle v \langle u \rangle^2$$

and

$$X = \langle u \rangle^2 u_1 \langle u \rangle u_1 \langle u \rangle^2 u_2 \langle u \rangle u_2 \langle u \rangle^2 \cdots \langle u \rangle^2 u_n \langle u \rangle u_n \langle u \rangle^2 \ .$$

The proof of the previous equivalence is based on the facts that the word $\langle u \rangle^2$ is a prefix and a suffix of Y and that it occurs in X in exactly $n+1$ places. We concentrate on the nontrivial part of the proof. Thus, if $X = ZYZ'$ holds there are essentially two possibilities for $v \langle u \rangle v$:

$$v \langle u \rangle v = u_i \langle u \rangle u_i \ , \quad \text{for some } i$$

or

$$\begin{aligned} v \langle u \rangle v = u_i \langle u \rangle u_i \langle u \rangle^2 u_{i+1} \langle u \rangle u_{i+1} \langle u \rangle^2 \cdots \\ u_{j-1} \langle u \rangle u_{j-1} \langle u \rangle^2 u_j \langle u \rangle u_j \ , \quad \text{for some } i \text{ and } j \text{ with } i < j. \end{aligned}$$

In the first case $v = u_i$ as required. In the second case we can use the positions of factors $\langle u \rangle$ and $\langle u \rangle^2$ to conclude that this case is not possible, which completes the proof. We separate the analyzis into two cases depending on whether $v \langle u \rangle v$ equals to an expression containing an odd number of factors $\langle u \rangle^2$ or an even number of those. The following two examples illustrate the argumentation in each case. We leave it to the reader to apply corresponding arguments for the other values of i and j.

Let $w = u_1 \langle u \rangle u_1 \langle u \rangle^2 u_2 \langle u \rangle u_2$ and assume $v \langle u \rangle v = w$. Now the factor $\langle u \rangle$ in the middle of $v \langle u \rangle v$ has to overlap with the factor $\langle u \rangle^2$ of w, otherwise one of the v's would contain a factor $\langle u \rangle^2$. In a general case the overlapping concerns the centermost occurrence of factors $\langle u \rangle^2$. Now the factor preceding (or succeeding) the mentioned $\langle u \rangle$ has the length at least $2|u_1| + 2|\langle u \rangle|$ (or $2|u_2| + 2|\langle u \rangle|$). We may assume $|v| \geq 2|u_1| + 2|\langle u \rangle|$, the other case being similar. Now $|v \langle u \rangle v| \geq 4|u_1| + 5|\langle u \rangle| > |w|$ because $|\langle u \rangle| > 2|u_2|$. This gives a contradiction.

Let $w' = u_1 \langle u \rangle u_1 \langle u \rangle^2 u_2 \langle u \rangle u_2 \langle u \rangle^2 u_3 \langle u \rangle u_3$ and assume $v \langle u \rangle v = w'$. Now the factor $\langle u \rangle^2$ has to be located in the same place on both occurrences of v in the word $v \langle u \rangle v$. This gives $v = u_1 \langle u \rangle u_1 \langle u \rangle^2 u_3 \langle u \rangle u_3$ and thus $|v \langle u \rangle v| = 9|\langle u \rangle| + 4|u_1| + 4|u_3| > |w'|$ giving a contradiction.

□

With a *positive* Boolean combination we refer to a Boolean combination that does not contain any negations, e.g. a Boolean combination of equations without inequalities. Now we can show that the conversion of a finite positive Boolean combination of equations over overlapping products into a single ordinary word equation requires only two extra unknowns.

Theorem 5. *For any finite positive Boolean combination of equations with overlapping products we can construct a single ordinary word equation with two additional unknowns such that the sets of solutions of the Boolean combination and the single equation are equal for some choice of these additional unknowns.*

Proof. For each equation over overlapping products we have a corresponding finite disjunction of ordinary equations based on reduction of Theorem 2. Thus, any finite positive Boolean combination of equations with overlapping products can be transformed into a finite positive Boolean combination of ordinary word equations. We may write the constructed Boolean combination in a disjunctive normal form and replace each conjuction of equations by a single equation. Thus, we have formed a finite disjunction of word equations without any additional unknowns. By Theorem 4 we can transform this disjunction into a single equation with two additional unknowns which proves the claim. □

The compactness theorem for words says that each system of equations over Σ^+ and with a finite number of unknowns is equivalent to some of its finite subsystems, see [1], [8] and also [9]. We remark that the analogical result concerning equations with overlapping products is not as obvious a consequence of the reduction as the satisfiability theorem analyzed in the end of this section. If we use the reduction on an infinite system of equations with overlapping products in order to be able to use the compactness theorem of ordinary word equations, we will end up with an infinite number of finite systems of disjunctions connected with conjunctions. Although, a finite positive Boolean combination of equations over overlapping products can be reduced into a single ordinary word equation with only two additional unknowns, a corresponding reduction of an infinite Boolean combination would require an infinite number of unknowns. Thus, we cannot use the original compactness theorem because of the infinite number of unknowns and the question about validity of the compactness theorem for equations over overlapping products remains open.

The decidability result for equations with overlapping products is instead obtained easily.

Theorem 6. *The satisfiability problem for a finite positive Boolean combination of equations with overlapping products is decidable.*

Proof. Theorem 3 shows that an equation with overlapping products can be reduced into a single equation without overlapping products. With Makanin's algorithm we can decide whether this equation without overlapping products has solutions or not and the existence of solutions is not affected by the additional unknowns in a sense that they would restrict the existence. Thus, we can straightforwardly decide the existence of solutions of the original equation with overlapping products, too. □

Acknowledgements. We would like to thank the anonymous referee for constructive comments, especially on Theorem 4.

References

1. Albert, M.H., Lawrence, J.: A proof of Ehrenfeucht's conjecture. Theoret. Comput. Sci. 41, 121–123 (1985)
2. Büchi, J.R., Senger, S.: Coding in the existential theory of concatenation. Arch. Math. Logik Grundlag. 26, 101–106 (1986/1987)
3. Choffrut, C., Karhumäki, J.: Combinatorics of words. In: Rozenberg, G., Salomaa, A. (eds.) Handbook of Formal Languages, vol. 1, pp. 329–438. Springer (1997)
4. Csuhaj-Varjú, E., Petre, I., Vaszil, G.: Self-assembly of strings and languages. Theoret. Comput. Sci. 374, 74–81 (2007)
5. Cărăuşu, A., Păun, G.: String intersection and short concatenation. Revue Roumaine de Mathématiques Pures et Appliquées 26, 713–726 (1981)
6. Domaratzki, M.: Semantic Shuffle on and Deletion Along Trajectories. In: Calude, C.S., Calude, E., Dinneen, M.J. (eds.) DLT 2004. LNCS, vol. 3340, pp. 163–174. Springer, Heidelberg (2004)
7. Domaratzki, M.: Minimality in Template-Guided Recombination. Information and Computation 207, 1209–1220 (2009)
8. Guba, V.S.: Equivalence of infinite systems of equations in free groups and semigroups to finite subsystems. Mat. Zametki 40, 321–324 (1986) (in Russian)
9. Harju, T., Karhumäki, J., Plandowski, W.: Independent systems of equations. In: Lothaire, M. (ed.) Algebraic Combinatorics on Words, ch. 13, pp. 443–472. Cambridge University Press (2002)
10. Holzer, M., Jakobi, S.: Chop Operations and Expressions: Descriptional Complexity Considerations. In: Mauri, G., Leporati, A. (eds.) DLT 2011. LNCS, vol. 6795, pp. 264–275. Springer, Heidelberg (2011)
11. Ito, M., Lischke, G.: Generalized periodicity and primitivity. Mathematical Logic Quarterly 53, 91–106 (2007)
12. Karhumäki, J., Mignosi, F., Plandowski, W.: The expressibility of languages and relations by word equations. J. ACM 47, 483–505 (2000)
13. Lothaire, M.: Combinatorics on words. Addison-Wesley (1983)
14. Lyndon, R.C., Schützenberger, M.P.: The equation $a^M = b^N c^P$ in a free group. Michigan Math. J. 9, 289–298 (1962)

15. Makanin, G.S.: The problem of the solvability of equations in a free semigroup. Mat. Sb. (N.S.) 103, 147–236 (1997) (in Russian), English translation in: Math. USSR-Sb. 32, 129–198 (1977)
16. Mateescu, A., Păun, G., Rozenberg, G., Salomaa, A.: Simple splicing systems. Discrete Applied Mathematics 84, 145–162 (1998)
17. Mateescu, A., Salomaa, A.: Parallel composition of words with re-entrant symbols. Analele Universităţii Bucureşti Mathematică-Informatică 45, 71–80 (1996)
18. Păun, G., Rozenberg, G., Salomaa, A.: DNA Computing, New Computing Paradigms. Springer (1998)
19. Plandowski, W.: Satisfiability of word equations with constants is in PSPACE. J. ACM 51, 483–496 (2004)