INTERNATIONAL JOURNAL OF FOUNDATIONS OF COMPUTER SCIENCE
Volume 22, Number 1, January 2011

CONTENTS

Special Issue — Natural Computing: Theory and Applications

This journal is covered in SciSearch® (also known as Science Citation Index - Expanded), ISI Alerting Services, CompuMath Citation Index®, Current Contents/Engineering, Computing & Technology, MathSciNet, Mathematical Reviews, DBLP Bibliography Server and Zentralblatt MATH.

International Journal of Foundations of Computer Science
Vol. 22, No. 1 (2011) 213–222

DOI: 10.1142/S0129054111007964

ORTHOGONAL SHUFFLE ON TRAJECTORIES

MARK DALEY, LILA KARI and SHINNOSUKE SEKI
Department of Computer Science, the University of Western Ontario
London, Ontario, N6A 5B7, Canada
{daley, lila, sseki}@csd.uwo.ca

PETR SOSÌK
Institute of Computer Science, Faculty of Philosophy and Science
Silesian University in Opava, 74601 Opava, Czech Republic
and
Departamento de Inteligencia Artificial, Facultad de Informática, Universidad Politécnica
de Madrid Campus de Montegancedo s/n, Boadilla del Monte, 28660 Madrid, Spain
petr.sosik@fpf.slu.cz

Received 10 June 2010
Accepted 15 September 2010
Communicated by Marian Gheorghe

A language L is called the orthogonal shuffle of the language L_1 with the language L_2, along the trajectory set T if every word in L is uniquely obtained as the shuffle between a word in L_1, a word in L_2 along a trajectory word in T. In this paper we investigate properties of the orthogonal shuffle on trajectories, as well as several types of language equations using this language operation. As a corollary, we obtain several properties of orthogonal catenation, orthogonal literal shuffle and orthogonal insertion.

Keywords: Shuffle on trajectories; Orthogonal operation; Language equation; Decidability.

1. Introduction

A language L is the orthogonal catenation of languages L_1 and L_2 if every word of L can be written in a unique way as a catenation of a word in L_1 and a word in L_2. In [3], Daley, Domaratzki, and Salomaa investigated the orthogonal catenation $\odot_\perp$. This notion can be generalized to other language operations, for example, to shuffle, or shuffle on trajectories. Shuffle on trajectories was introduced by Mateescu, Rozenberg, and Salomaa [9] in order to generalize several operations on words and languages, and was investigated in detail by Domaratzki, e.g., see [8]. In this paper, we generalize orthogonal catenation to orthogonal shuffle on trajectories, and investigate several problems related to this operation. The paper is organized as follows. Section 2 contains the formal definition of orthogonal operations, including orthogonal shuffle on trajectories, as well as some general properties of this operation. Section 3 addresses several decidability questions. For example, we give a proof of the fact that it is decidable, given a regular set of trajectories T, and two regular languages L_1 and L_2, whether or not their orthogonal catenation, or their orthogonal shuffle on T, is defined (Theorem 8). If one of the languages is linear and the other is regular, while the trajectory set is still regular, then the same problem becomes undecidable (Theorem 9). We also prove that when an equation of the form $L_1 \circ X = L$, where $\circ$ denotes the orthogonal shuffle on a complete set of trajectories T, has a solution, this solution is minimal and unique (Proposition 10, Theorem 12). Thus, a similar result holds if $\circ$ denotes orthogonal catenation, orthogonal literal shuffle, or orthogonal insertion (Corollary 13). Lastly, we prove that if the language L is linear, the language L_1 is regular, and the operation involved is orthogonal catenation, then it is undecidable whether or not such a language equation has a solution (Corollary 16). Section 4 presents several topics of future work. We conclude this introductory section with several notions and notation used in this paper. Let Σ be a finite alphabet that is totally ordered by the ordering $\prec$. A sequence of letters in Σ is called a *word over* Σ. The *length* of a word $w \in \Sigma^*$ is the number of letters occurring in it, and denoted by $|w|$. In particular, the *empty word*, denoted by λ, is the word of length zero. For a word $w \in \Sigma^*$ and a letter $a \in \Sigma$, $|w|_a$ denotes the number of occurrences of a in w. The set of all words (all non-empty words) is denoted by Σ^* (resp. Σ^+). Moreover, for $n \geq 0$, let $\Sigma^n = \{w \in \Sigma^* : |w| = n\}$. A subset of Σ^* is called a *language*. For a non-empty language L, let $\underline{L}$ be the set of all shortest words in L.

A word $u \in \Sigma^*$ is called a *prefix* of a word $v \in \Sigma^*$ if $v = ux$ for some $x \in \Sigma^*$ (in this case, the left quotient $u^{-1}v$ is defined as x). By $\mathrm{pref}_1(u)$, we denote the prefix of u of length 1, i.e., the first letter of u. Two words u and v have always a unique *maximal common prefix*, which is denoted by $u \wedge v$. Based on that, a *total ordering* of Σ^* called *lexicographic ordering* is introduced, and denoted by $\prec_{\mathrm{lex}}$. The order on Σ by $\prec$ is extended to Σ^* in the following way:

$$u \prec_{\mathrm{lex}} v \iff u^{-1}v \in \Sigma^+ \text{ or } \mathrm{pref}_1((u \wedge v)^{-1}u) \prec \mathrm{pref}_1((u \wedge v)^{-1}v).$$

A trajectory is a binary word over $\{0, 1\}$. Consider a trajectory t and two words $u = a_1 a_2 \cdots a_i$ and $v = b_1 b_2 \cdots b_j$ for some $i, j \geq 0$ and $a_1, \ldots, a_i, b_1, \ldots, b_j \in \Sigma$. The shuffle of u and v on the trajectory t, denoted by $u \sqcup\!\sqcup_t v$, is defined as follows: if $|u| \neq |t|_0$ or $|v| \neq |t|_1$, then $u \sqcup\!\sqcup_t v = \emptyset$; otherwise $u \sqcup\!\sqcup_t v = c_1 c_2 \cdots c_{i+j}$, where, for $1 \leq k \leq i + j$, if $|t_1 t_2 \cdots t_{k-1}|_0 = n_0$ and $|t_1 t_2 \cdots t_{k-1}|_1 = n_1$, then

$$c_k = \begin{cases} a_{n_0+1} & \text{if } t_k = 0 \\ b_{n_1+1} & \text{if } t_k = 1. \end{cases}$$

Shuffle on trajectories is extended to a set $T \subseteq \{0, 1\}^*$ of trajectories as follows:

$$u \sqcup\!\sqcup_T v = \bigcup_{t \in T} u \sqcup\!\sqcup_t v.$$

Further, for languages $L_1, L_2 \subseteq \Sigma^*$, we define

$$L_1 \sqcup\!\sqcup_T L_2 = \bigcup_{u \in L_1, v \in L_2} u \sqcup\!\sqcup_T v.$$

A set of trajectories T is said to be *complete* if $u \sqcup\!\sqcup_T v \neq \emptyset$ for all $u, v \in \Sigma^*$ [9]. In other words, T is complete if and only if for any $(i, j) \in \mathbb{N}^2$, there exists $t \in T$ which contains i 0's and j 1's.

Lemma 1. *Let $u, v, v' \in \Sigma^*$ and $t \in \{0, 1\}^*$ such that neither $u \sqcup\!\sqcup_t v$ nor $u \sqcup\!\sqcup_t v'$ is empty. If $v \prec_{\text{lex}} v'$, then $u \sqcup\!\sqcup_t v \prec_{\text{lex}} u \sqcup\!\sqcup_t v'$.*

A language $L \subseteq \Sigma^*$ is called a *uniform code* if for any $u, v \in L$, $|u| = |v|$.

Lemma 2. *For uniform codes $C, C_1, C_2 \subseteq \Sigma^*$ and a set $T \in \{0, 1\}^*$ of trajectories, if $C \sqcup\!\sqcup_T C_1 = C \sqcup\!\sqcup_T C_2 \neq \emptyset$, then $C_1 \cap C_2 \neq \emptyset$.*

Proof. The equality implies that the code lengths of C_1 and C_2 are the same. When Σ is unary, this lemma holds trivially because in this case, if C_1 and C_2 have the same code lengths, then they have to be the same. Let us consider the case when Σ is non-unary, and suppose that $C_1 \cap C_2 = \emptyset$. Let $w_{\min}$ be the smallest word in $C \sqcup\!\sqcup_T C_1$ with respect to $\prec_{\text{lex}}$. Due to the equality $C \sqcup\!\sqcup_T C_1 = C \sqcup\!\sqcup_T C_2$, there exist $u, u' \in C$, $v \in C_1$, $v' \in C_2$, and $t, t' \in T$ such that

$$w_{\min} = u \sqcup\!\sqcup_t v = u' \sqcup\!\sqcup_{t'} v'.$$

Since C_1 and C_2 are assumed to be disjoint, $v \neq v'$. As such, either $v \prec_{\text{lex}} v'$ or $v \succ_{\text{lex}} v'$ holds. However, in the former case, $u' \sqcup\!\sqcup_{t'} v \prec_{\text{lex}} u' \sqcup\!\sqcup_{t'} v'$, which contradicts the smallest property of $w_{\min}$ because $u' \sqcup\!\sqcup_{t'} v \in C \sqcup\!\sqcup_T C_1$. We have exactly the same contradiction even in the latter case. □

2. Orthogonal Operations on Languages

In the following, we investigate properties of a special case of operations on words and languages, termed *orthogonal operations*. Let $\circ$ be a binary operation on words, called *bin-op*. For two languages L_1, L_2, consider the following condition:

(OR1) $(\forall u, u' \in L_1, v, v' \in L_2)$ if $u \neq u'$ or $v \neq v'$, then $u \circ v \cap u' \circ v' = \emptyset$.

Then, we define the *orthogonal bin-op* of L_1 and L_2 as

$$L_1 \circ_\perp L_2 = \begin{cases} L_1 \circ L_2 & \text{if condition } \textbf{(OR1)} \text{ holds,} \\ \text{undefined} & \text{otherwise.} \end{cases}$$

If $L_1 \circ_\perp L_2$ is defined, we say that L_1 and L_2 are *bin-op*-orthogonal, or $\circ$-orthogonal. We say that a language L is an *orthogonal bin-op of* L_1 *and* L_2 if $L = L_1 \circ_\perp L_2$.

In this paper we focus on *orthogonal catenation*, denoted by $\odot_\perp$, and especially on *orthogonal shuffle on trajectories*, denoted by $\sqcup\!\sqcup_T^\perp$ for a set of trajectories T. In the case of orthogonal shuffle on trajectories, we redefine the notion of orthogonality by replacing **(OR1)** with the following equivalent condition:

(OR2) $(\forall u, u' \in L_1, v, v' \in L_2, t, t' \in T)$ if $u \neq u'$ or $v \neq v'$, then $u \sqcup\!\sqcup_t v \neq u' \sqcup\!\sqcup_{t'} v'$.

Recall that a language L is *k-thin* if $|L \cap \Sigma^n| \leq k$ for all $n \geq 0$ [10].

Proposition 3. *Let T be a 1-thin set of trajectories. Then for any languages L_1, L_2, $L_1 \sqcup\!\sqcup^{\perp}_{T} L_2$ is always defined and equal to $L_1 \sqcup\!\sqcup_{T} L_2$.*

Proof. Since T is 1-thin, any $w \in L_1 \sqcup\!\sqcup_{T} L_2$ admits a unique decomposition $w = u \sqcup\!\sqcup_{t} v$, where $u \in L_1$, $v \in L_2$, and $t \in T$. This means that (OR2) is satisfied. Hence, $L_1 \sqcup\!\sqcup^{\perp}_{T} L_2$ is defined and equal to $L_1 \sqcup\!\sqcup_{T} L_2$. □

Thus, the known results about shuffle on trajectories apply to orthogonal shuffle on trajectories when the set of trajectories is 1-thin. For example, Domaratzki and Salomaa proved that for given a regular language R and a 1-thin set T of trajectories, it is decidable whether there exist languages $L_1, L_2 \neq \{\lambda\}$ such that $R = L_1 \sqcup\!\sqcup_{T} L_2$ [4]. The problem obtained by replacing $\sqcup\!\sqcup_{T}$ by $\sqcup\!\sqcup^{\perp}_{T}$ in this problem is also decidable. We now prove several basic properties of orthogonal shuffle on trajectories. Assume that orthogonal shuffle of two languages is defined on a set of trajectories. Then on smaller sets of trajectories, shuffle of the languages remains orthogonal.

Proposition 4. *Let $L_1, L_2 \subseteq \Sigma^*$ and $T \subseteq \{0,1\}^*$. If $L_1 \sqcup\!\sqcup^{\perp}_{T} L_2$ is defined, then $L_1 \sqcup\!\sqcup^{\perp}_{T'} L_2$ is defined for any $T' \subseteq T$.*

An analogous result of Proposition 4 holds for the case when the two operands are replaced by their respective subsets.

Lemma 2 has an analogous result in relation to orthogonal shuffle on trajectories.

Lemma 5. *For uniform codes $C, C_1, C_2 \subseteq \Sigma^*$ and a set $T \subseteq \{0,1\}^*$ of trajectories, if $C \sqcup\!\sqcup^{\perp}_{T} C_1 = C \sqcup\!\sqcup^{\perp}_{T} C_2 \neq \emptyset$, then $C_1 = C_2$.*

Proof. As in the proof of Lemma 2, this lemma is trivial when Σ is unary. For the case when Σ is not unary, suppose $C_1 \neq C_2$, or we could suppose $C_1 - C_2 \neq \emptyset$ without loss of generality[a]. For any $u \in C$, $v \in C_1 - C_2$, and $t \in T$, there exist $u' \in C$, $v' \in C_2$, and $t' \in T$ such that

$$u \sqcup\!\sqcup_{t} v = u' \sqcup\!\sqcup_{t'} v'. \tag{7}$$

Since $v \notin C_2$, $v \neq v'$. Note that v' must not be in C_1 because otherwise Eq. (7) would violate the orthogonality of $C \sqcup\!\sqcup^{\perp}_{T} C_1$. Therefore, $C \sqcup\!\sqcup^{\perp}_{T} (C_1 - C_2) \subseteq C \sqcup\!\sqcup^{\perp}_{T} (C_2 - C_1)$. In the similar manner, its opposite inclusion relation can be proved. Hence, we would have $C \sqcup\!\sqcup^{\perp}_{T} (C_1 - C_2) = C \sqcup\!\sqcup^{\perp}_{T} (C_2 - C_1)$, but this contradicts Lemma 2. □

For a set $T \subseteq \{0,1\}^*$ of trajectories, a language $L \subseteq \Sigma^*$ is called a *T-code* if $(L \sqcup\!\sqcup_{T} \Sigma^+) \cap L = \emptyset$ [4].

Proposition 6. *Let $T \subseteq \{0,1\}^*$ that contains 0^*. For a language $L \subseteq \Sigma^*$, if $L \sqcup\!\sqcup^{\perp}_{T} \Sigma^*$ is defined, then L is a T-code.*

---

[a] $C_1 - C_2$ is defined as the set $\{w \in C_1 \mid w \notin C_2\}$.

Proof. Suppose that L were not T-code, i.e., there exist $u, v \in L$ such that $v \in u \sqcup\!\sqcup_T \Sigma^+$. Note that $|v| > |u|$ so that $v \neq u$. Since $0^* \subseteq T$, $v \in v \sqcup\!\sqcup_T \Sigma^*$. Thus, $u \sqcup\!\sqcup_T \Sigma^+ \cap v \sqcup\!\sqcup_T \Sigma^* \neq \emptyset$, and as a result $L \sqcup\!\sqcup_T^{\perp} \Sigma^*$ should not be defined. □

A set T of trajectories is said to be *associative* if $(u \sqcup\!\sqcup_T v) \sqcup\!\sqcup_T w = u \sqcup\!\sqcup_T (v \sqcup\!\sqcup_T w)$ for all $u, v, w \in \Sigma^*$ [9].

Lemma 7. *Let L_1, L_2, L_3 be non-empty languages, and T be an associative set of trajectories. If both $(L_1 \sqcup\!\sqcup_T^{\perp} L_2) \sqcup\!\sqcup_T^{\perp} L_3$ and $L_1 \sqcup\!\sqcup_T^{\perp} (L_2 \sqcup\!\sqcup_T^{\perp} L_3)$ are defined, then*

$$(L_1 \sqcup\!\sqcup_T^{\perp} L_2) \sqcup\!\sqcup_T^{\perp} L_3 = L_1 \sqcup\!\sqcup_T^{\perp} (L_2 \sqcup\!\sqcup_T^{\perp} L_3).$$

Proof. Let $w \in (L_1 \sqcup\!\sqcup_T^{\perp} L_2) \sqcup\!\sqcup_T^{\perp} L_3$. Since this set is defined, $w \in (L_1 \sqcup\!\sqcup_T L_2) \sqcup\!\sqcup_T L_3$. Due to the associativity of T, $w \in L_1 \sqcup\!\sqcup_T (L_2 \sqcup\!\sqcup_T L_3)$. Since $L_1 \sqcup\!\sqcup_T^{\perp} (L_2 \sqcup\!\sqcup_T^{\perp} L_3)$ is defined, it is equal to $L_1 \sqcup\!\sqcup_T (L_2 \sqcup\!\sqcup_T L_3)$, and hence, it contains w. Therefore, $(L_1 \sqcup\!\sqcup_T^{\perp} L_2) \sqcup\!\sqcup_T^{\perp} L_3 \subseteq L_1 \sqcup\!\sqcup_T^{\perp} (L_2 \sqcup\!\sqcup_T^{\perp} L_3)$. Analogously we can prove the opposite inclusion relation. □

3. Decidability and Language Equations

This section addresses several decidability questions related to the orthogonal shuffle on trajectories. Given regular languages L_1, L_2, and a trajectory set T, we first ask whether or not it is decidable if L_1 and L_2 are $\sqcup\!\sqcup_T$-orthogonal. Secondly, for non-empty languages L and L_1, and a complete set T of trajectories, we ask whether or not a solution to the equation $L_1 \sqcup\!\sqcup_T^{\perp} X = L$ is unique if any. Thirdly, for regular languages R_1, R and a regular set T of trajectories, we ask if it is decidable whether or not the equation $R_1 \sqcup\!\sqcup_T^{\perp} X = R$ has a solution.

Question 1. *For given languages L_1, L_2, and a trajectory set T, is it decidable whether or not L_1 and L_2 are $\sqcup\!\sqcup_T$-orthogonal?*

The following two results have given partial solutions to Question 1.

Theorem 8 ([3]) *Given regular languages $R_1, R_2 \subseteq \Sigma^*$ and a regular set of trajectories T, it is decidable whether R_1 and R_2 are (i) $\odot$-orthogonal, (ii) $\sqcup\!\sqcup_T$-orthogonal.*

Proof. (i) In [3], the authors state that this result is well-known, without mentioning specific references. A short proof is: since R_1 and R_2 are regular, so are the left quotient $R_1^{-1} R_1$ and the right quotient $R_2 R_2^{-1}$. Then R_1 and R_2 are $\odot$-orthogonal if and only if $R_1^{-1} R_1 \cap R_2 R_2^{-1} = \{\lambda\}$. The latter is decidable, and hence, so is the former. (ii) The statement is proven in [3], Theorem 5. □

This decidability result is complemented by the undecidability result obtained by expanding the language class which R_1 or R_2 belongs to up to the class of linear languages.

Theorem 9. *Given a linear language* $L \subseteq \Sigma^*$*, a regular language* $R \subseteq \Sigma^*$*, and a regular set* T *of trajectories, it is undecidable whether or not*

(1) L *and* R *are* $\sqcup\!\sqcup_T$*-orthogonal,*
(2) R *and* L *are* $\sqcup\!\sqcup_T$*-orthogonal.*

Proof. The instance of the first (second) problem with $T = 0^*1^*$ and $R = \Sigma^*$ is known to be equivalent to the problem of whether L is a prefix (resp. suffix) code [3]. It is undecidable whether a given linear language is a prefix- (a suffix-) code. As a result, these two problems have to be undecidable.

This result is also verified by the fact that it is undecidable whether for a linear language L and a regular language R, L and R are $\odot$-orthogonal [3]. □

Language equations involving shuffle on trajectories were intensively investigated in [11]. Here we address this question, but shift our focus to orthogonal shuffle on trajectories. First of all, let us recall the equation

$$L_1 \sqcup\!\sqcup_T X = L, \tag{9}$$

where L_1, L are languages over an alphabet Σ, T is a set of trajectories, and X is a variable. As done in [4], we define the right-useful solutions to Eq. (9) as

$$\mathrm{use}_T^{(r)}(X; L_1) = \{x \in X \mid L_1 \sqcup\!\sqcup_T x \neq \emptyset\}, \tag{10}$$

where X is any language. Since $L_1 \sqcup\!\sqcup_T (X - \mathrm{use}_T^{(r)}(X; L_1)) = \emptyset$, in the following we assume that any solution X of Eq. (9) satisfies $X = \mathrm{use}_T^{(r)}(X; L_1)$. Let us replace shuffle in Eq.(9) with orthogonal shuffle and consider an equation

$$L_1 \sqcup\!\sqcup_T^{\perp} X = L. \tag{11}$$

Then the following question arises:

Question 2. *Let* L *and* L_1 *be non-empty languages, and* T *be a set of trajectories. When Eq.* (11) *has a solution for the variable* X*, is this solution unique?*

By definition of orthogonal shuffle on trajectories, it is clear that a solution to Eq. (11) is a solution to Eq. (9). The next proposition strengthens this statement further.

Proposition 10. *For a set* T *of trajectories, if the language equation* $L_1 \sqcup\!\sqcup_T^{\perp} X = L$ *has a solution* L_2*, then* L_2 *is a minimal solution of* $L_1 \sqcup\!\sqcup_T Y = L$*.*

Proof. Suppose that there were a language L' which is a proper subset of L_2 and satisfies $L_1 \sqcup\!\sqcup_T L' = L$. Because of the assumption that $L_2 = \mathrm{use}_T^{(r)}(L_2, L_1)$, for any $v \in L_2 - L'$, $L_1 \sqcup\!\sqcup_T v \neq \emptyset$. With $L_1 \sqcup\!\sqcup_T v \subseteq L$, this implies that $L_1 \sqcup\!\sqcup_T v \cap L_1 \sqcup\!\sqcup_T L'$ is not empty. However, this breaks the orthogonality of $L_1 \sqcup\!\sqcup_T^{\perp} L_2$ because $L' \subseteq L_2$ and $v \in L_2 - L'$. □

This proposition means that, unlike the study on language equations based on shuffle on trajectories, we have to focus on the minimal solutions of $L_1 \sqcup\!\sqcup_T Y = L$ when considering the solutions of $L_1 \sqcup\!\sqcup_T^{\perp} X = L$. Compared to the maximal solution of language equations in general [11, 12], much less is known about minimal solutions of language equations. Let us imagine that for given languages L_1, L and a given set T of trajectories, the equation $L_1 \sqcup\!\sqcup_T^{\perp} X = L$ has a solution. Is this solution unique? In general, this uniqueness does not hold as shown in the next example.

Example 11. *Let* $L_1 = \{aa, aaa\}$, $T = \{001111, 000111\}$, *and* $L = \{aaabbb, aaaabb\}$. *Then both* $L_2 = \{aabb, bbb\}$ *and* $L_2' = \{abb, abbb\}$ *are solutions to* $L_1 \sqcup\!\sqcup_T^{\perp} X = L$.

On the other hand, it was proved in [3] that for the catenation $\odot$, if the equation $L_1 \odot_{\perp} X = L$ has a solution, then the solution is unique. Recall that catenation is a special case of shuffle on trajectories (with $T = 0^*1^*$, we have $\odot = \sqcup\!\sqcup_T$). Note that T in Example 11 is not complete, whereas $T = 0^*1^*$ is. We generalize this uniqueness result for orthogonal shuffle on any complete sets of trajectories.

Theorem 12. *Let* L *and* L_1 *be non-empty languages, and* T *be a complete set of trajectories. If an equation* $L_1 \sqcup\!\sqcup_T^{\perp} X = L$ *has a solution for the variable* X, *the solution is unique.*

Proof. Suppose that there were two distinct languages $L_2, L_2' \subseteq \Sigma^*$ such that

$$L_1 \sqcup\!\sqcup_T^{\perp} L_2 = L_1 \sqcup\!\sqcup_T^{\perp} L_2' = L. \qquad (12)$$

Let n be the length of a shortest word in L_1, and let m be the length of a shortest word in the symmetric difference between L_2 and L_2'. Note that we can assume without loss of generality that $L_2 - L_2'$ contains such a word of length m because of the symmetric roles of L_2 and L_2' in Eq. (12). Thus, $\underline{L_1} = L_1 \cap \Sigma^n$ and $\underline{L_2 - L_2'} = (L_2 - L_2') \cap \Sigma^m$. Choose an arbitrary $\underline{u} \in \underline{L_1}$, $\underline{v} \in \underline{L_2 - L_2'}$, and $t \in T$ for which $\underline{u} \sqcup\!\sqcup_t \underline{v}$ is not the empty set. The existence of such t is guaranteed by the completeness of T. Since $\underline{u} \sqcup\!\sqcup_t \underline{v} \in L_1 \sqcup\!\sqcup_T^{\perp} L_2 = L$, Eq. (12) implies that there exist $u \in L_1$, $v' \in L_2'$, and $t' \in T$ such that

$$\underline{u} \sqcup\!\sqcup_t \underline{v} = u \sqcup\!\sqcup_{t'} v'. \qquad (13)$$

Since $\underline{v} \notin L_2'$, we have $\underline{v} \neq v'$. If $v' \in L_2$, then Eq. (13) violates the orthogonality of $L_1 \sqcup\!\sqcup_T^{\perp} L_2$. Thus, $v' \in L_2' - L_2$, and hence, we have $|\underline{v}| \leq |v'|$ by the assumption that $\underline{v}$ be shortest among words in $(L_2 - L_2') \cup (L_2' - L_2)$. Provided this inequality holds strictly, then Eq. (13) implies that $|\underline{u}| > |u|$, contradicting the definition of u being shortest in L_1. Summarizing what we have obtained so far,

(1) for any $u \in \underline{L_1}$, $v \in \underline{L_2 - L_2'}$, and $t \in T$, there exist $u' \in \underline{L_1}$, $v' \in \underline{L_2' - L_2}$, and $t' \in T$ such that $u \sqcup\!\sqcup_t v = u' \sqcup\!\sqcup_{t'} v'$;
(2) $L_2' - L_2$ also contains a word of length m;

(3) for any $u \in \underline{L_1}$, $v' \in \underline{L_2' - L_2}$, and $t \in T$, there exist $u' \in \underline{L_1}$, $v \in \underline{L_2 - L_2'}$, and $t' \in T$ such that $u \sqcup\!\sqcup_t v' = u' \sqcup\!\sqcup_{t'} v$;

Although what we actually proved were the first and second statements, the third statement is the corollary of these and our previous argument. Thus,

$$\underline{L_1} \sqcup\!\sqcup_T^{\perp} \underline{L_2 - L_2'} = \underline{L_1} \sqcup\!\sqcup_T^{\perp} \underline{L_2' - L_2}. \tag{14}$$

In the light of Lemma 5, however, Eq. (14) cannot hold. □

As pointed out in [8,9], catenation, literal shuffle, and insertion are particular cases of the operation of shuffle on complete set of trajectories. Thus, this theorem has the following corollary.

Corollary 13. *Let $\circ$ be either catenation, literal shuffle, or insertion. For languages L_1, L, if the equation $L_1 \circ_\perp X = L$ has a solution for the variable X, then the solution is unique.*

So far we have been working on the assumption that a given equation has a solution. A more interesting topic is to consider a method of solving a given equation. Let us start our investigation along this line with one-variable equations. An algorithm is known for solving and constructing the (unique, regular) solution (if any) of an equation $R = R_1 \odot_\perp X$ for regular languages R, R_1 [1,3]. We consider the following question:

Question 3. *Let R_1, R be regular languages and T be a regular set of trajectories. Is it decidable whether the equation $R_1 \sqcup\!\sqcup_T^{\perp} X = R$ has a solution or not?*

If we limit our scope to the singleton solution, i.e., we consider the equation $R_1 \sqcup\!\sqcup_T^{\perp} \{x\} = R$, then this problem is decidable.

Proposition 14. *Given regular languages R_1, R and a regular set T of trajectories, the problem of whether there exists a word w satisfying $R_1 \sqcup\!\sqcup_T^{\perp} \{w\} = R$ is decidable.*

Proof. Let $n = \min\{|u| \mid u \in R\}$, which can be computed by breadth-first search on the minimum deterministic finite automaton accepting R. The solution to $R_1 \sqcup\!\sqcup_T^{\perp} \{w\} = R$ is of length at most n. Thus, we simply test for all words of length at most n whether the desired equality is satisfied with orthogonality using Theorem 8. □

The next proposition also gives a partial answer to Question 3.

Proposition 15. *Let $L_1, L \subseteq \Sigma^*$ be languages and $T \subseteq \{0,1\}^*$ be a set of trajectories. If L is context-free, then it is undecidable whether the equation $L_1 \sqcup\!\sqcup_T^{\perp} X = L$ has a solution for the variable X.*

Proof. We reduce the Post Correspondence Problem (PCP) to the problem of whether the equation $L_1 \sqcup\!\sqcup_T^{\perp} X = L$ has a solution or not for some specific L_1, T, L.

Let the PCP instance consist of lists $\alpha = (\alpha_1, \ldots, \alpha_n)$ and $\beta = (\beta_1, \ldots, \beta_n)$, where $\alpha_i, \beta_i \in \{0,1\}^+$. Consider the following two context-free languages:

$$L_\alpha = \{01^{2i_1+1} \cdots 01^{2i_k+1} 0^{2n+2} 1^{2n+2} \alpha_{i_k} \cdots \alpha_{i_1} \mid k \geq 1, 1 \leq i_p \leq n, 1 \leq p \leq k\},$$

and

$$L_\beta = \{01^{2j_1+1} \cdots 01^{2j_m+1} 0^{2n+2} 1^{2n+2} \beta_{j_m} \cdots \beta_{j_1} \mid m \geq 1, 1 \leq j_q \leq m, 1 \leq q \leq m\}.$$

Although in general CFG's are not closed under complementation, the complement of a so-called *List language* is known to be context-free [7]. L_α and L_β are a variant of List language, and hence, their complements L_α^c and L_β^c are context-free. Since CFG's are closed under union, $(L_\alpha \cap L_\beta)^c = L_\alpha^c \cup L_\beta^c$ is context-free. Note that this PCP instance has a solution if and only if $(L_\alpha \cap L_\beta)^c \neq \Sigma^*$. For $L_1 = \cup_{i \geq 0} \Sigma^{2i}$ and $T = 0^*1^*$, let us consider the language equation $L_1 \sqcup\!\sqcup_T^{\perp} X = (L_\alpha \cap L_\beta)^c$, where X is a variable. If the PCP instance does not have a solution, then this equation is $L_1 \sqcup\!\sqcup_T^{\perp} X = \Sigma^*$ and it has a solution $X = \{\lambda\} \cup \Sigma$. On the other hand, if PCP(α, β) has a solution, say $i_1, i_2, \ldots, i_s$, where $1 \leq s$, $1 \leq i_h \leq n$, for all $h, 1 \leq h \leq s$, then the word

$$w = 01^{2i_1+1} 01^{2i_2+1} \cdots 01^{2i_s+1} 0^{2n+2} 1^{2n+2} \alpha_{i_s} \cdots \alpha_{i_2} \alpha_{i_1}$$

is not in $(L_\alpha \cap L_\beta)^c$. By taking the first two letters (01) from w generates another word w' (i.e., $w = 01w'$). This word w' begins with 1 so that it cannot be in L_α or L_β. Thus, if we suppose that the equation $L_1 \sqcup\!\sqcup_T^{\perp} X = (L_\alpha \cap L_\beta)^c$ had a solution L_2, then there would exist $u \in L_1$ and $v \in L_2$ such that $w' = uv$ (recall that $T = 0^*1^*$, that is, $\sqcup\!\sqcup_T^{\perp}$ is equivalent to catenation). By definition of L_1, $01u \in L_1$ so that $w = 01uv$ would be in $L_1 \sqcup\!\sqcup_T^{\perp} L_2$, that is, $w \in (L_\alpha \cap L_\beta)^c$, a contradiction. □

Note that in the previous proof, L_1 is regular and the trajectory set employed is $T = 0^*1^*$, that is, $\sqcup\!\sqcup_T^{\perp} = \odot_\perp$. Hence, Proposition 15 augments the decidability result mentioned previously as:

Corollary 16. *For a linear language L, and a regular language R_1, it is undecidable whether the equation $R_1 \odot_\perp X = L$ has a solution for the variable X.*

4. Conclusions

This paper studied properties of the orthogonal shuffle on trajectories, i.e., a special case of shuffle on trajectories where every word in the result is the product of the unique combination of one word from each operand. Several topics of future work are of interest, for example Question 3 in its most general setting, and the following variant of Question 1:

Question 4. *For given regular languages R_1, R_2, is it decidable whether there exists a complete set T of trajectories such that R_1 and R_2 are $\sqcup\!\sqcup_T$-orthogonal?*

Remark that this question is meaningful because there exists languages R_1, R_2 such that $R_1 \sqcup\!\sqcup^{\perp}_T R_2$ is undefined for any complete set T of trajectories. A trivial example is $R_1 = a^*$ and $R_2 = \{\lambda, a\}$. A complete set T of trajectories have to contain t, t' such that $a^2 \sqcup\!\sqcup_t \lambda$ and $a \sqcup\!\sqcup_{t'} a$ are defined. However, $a^2 \sqcup\!\sqcup_t \lambda = a \sqcup\!\sqcup_{t'} a = a^2$, and hence, $R_1 \sqcup\!\sqcup^{\perp}_T R_2$ is undefined. An example over a binary alphabet is $R_1 = a^* \cup a^*ba^*$ and $R_2 = \{\lambda, b\}$. Note that for given languages L_1, L_2, it is not always the case that such a set T of trajectories that $L_1 \sqcup\!\sqcup^{\perp}_T L_2$ is defined is unique. Indeed, let $L_1 = a^*b$, $L_2 = \{ab\}$, $T_1 = 0^*101$, and $T_2 = 0^*110$. Then $L_1 \sqcup\!\sqcup^{\perp}_{T_1} L_2 = L_1 \sqcup\!\sqcup^{\perp}_{T_2} L_2 = a^*abb$.

Acknowledgments

This work was supported by The Natural Sciences and Engineering Council of Canada Discovery Grant R2824A01, and Canada Research Chair Award to L.K., and by the Ministerio de Ciencia e Innovación (MICINN), Spain, under project TIN2009–14421, by the Comunidad de Madrid (grant No. CCG06-UPM/TIC-0386 to the LIA research group), and by the Silesian University in Opava, grant No. SGS/4/2010 to P.S.

References

[1] M. Anselmo and A. Restivo. On Languages Factorizing the Free Monoid. *International Journal of Algebra and Computations*, 6:413–427, 1996.

[2] Jurek Czyzowicz, Wojciech Fraczak, Andrzej Pelc and Wojciech Rytter. Prime Decompositions of Regular Prefix Codes. In *In Proceedings CIAA 2002, LNCS, 2608*, pages 85–94. Springer, 2002.

[3] M. Daley, M. Domaratzki and K. Salomaa. On the Operational Orthogonality of Languages. In *Proceedings Theory and Applications of Language Equations, M. Kunc and A. Okhotin, eds., TUCS General Publication, 44*, pages 43–53, 2007.

[4] M. Domaratzki and K. Salomaa. Restricted Sets of Trajectories and Decidability of Shuffle Decompositions. *International Journal of Foundations of Computer Science*, 16(5):897–912, 2005.

[5] S. Eilenberg. *Automata, Languages, and Machines.* Academic Press, volume A, 1974.

[6] A. Mateescu, A. Salomaa and S. Yu. On the Decomposition of Finite Languages. *Technical report, Turku Center of Computer Science*, 1998.

[7] J. E. Hopcroft, R. Motwani and J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation.* Addison-Wesley, Boston, 2nd edition, 2001.

[8] M. Domaratzki. *Trajectory-Based Operations.* PhD Thesis, Queen's University, 2004.

[9] A. Mateescu, G. Rozenberg and A. Salomaa. Shuffle on Trajectories: Syntactic Constraints. *Theoretical Computer Science*, 197:1–56, 1998.

[10] Gh. Păun and A. Salomaa. Thin and Slender Languages. *Discrete Applied Mathematics*, 61:257–270, 1995.

[11] M. Domaratzki and K. Salomaa. Decidability of Trajectory-based Equations. *Theoretical Computer Science*, 345:304–330, 2005.

[12] L. Kari. On Language Equations with Invertible Operations. *Theoretical Computer Science*, 132:129–150, 1994.