A Kleene Theorem for Languages of Words Indexed by Linear Orderings

Alexis Bès[1] and Olivier Carton[2]

[1] LACL, Université Paris XII-Val de Marne,
Email: `bes@univ-paris12.fr`,
Url: `http://www.univ-paris12.fr/lacl/bes/`
[2] LIAFA, Université Paris 7
Email: `Olivier.Carton@liafa.jussieu.fr`,
Url: `http://www.liafa.jussieu.fr/~carton/`

Abstract. In a preceding paper, Bruyère and the second author introduced automata, as well as rational expressions, which allow to deal with words indexed by linear orderings. A Kleene-like theorem was proved for words indexed by countable scattered linear orderings. In this paper we extend this result to languages of words indexed by all linear orderings.

1 Introduction

One of the fundamental results in automata theory is Kleene's theorem [15] which asserts the equivalence between sets of words accepted by automata and set of words described by rational expressions. During the past fifty years Kleene's theorem has been extended to various notions of infinite words, as well as structures like trees, pictures, and traces.

In [3], Bruyère and Carton introduce automata and rational expressions for words on linear orderings. These notions unify naturally previously defined notions for finite words, left- and right-infinite words, bi-infinite words, and ordinal words. They also prove that a Kleene-like theorem holds when the orderings are restricted to countable scattered linear orderings; recall that a linear ordering is scattered if it does not contain any dense sub-ordering. This result extends Kleene's theorem for finite words [15], infinite words [5, 16], bi-infinite words [12, 17] and ordinal words [7, 9, 26]. Since [3], the study of automata on linear orderings was carried on in several papers, that address the emptiness problem and the containment problem for languages [25, 8], as well as the classification of rational languages with respect to the rational operations needed to describe them [4]. More recently, the second author and Rispal [20] proved that regular languages of words over countable scattered linear orderings are closed under complementation.

In this paper we come back to Kleene's theorem, and show that the assumption that the linear orderings are countable and scattered, is not necessary. When all linear orderings are considered instead of countable and scattered ones, no change has to be made to the notion of automata already introduced in [3].

However the set of operators for the rational expressions has to be extended in order to deal with words indexed by a dense linear ordering. To cope with this issue, we add a *shuffle* operator for languages, which is a variant of the classical shuffle operation on linear orderings. A similar (but not equivalent) notion of shuffle for languages was already considered in [10, 14, 23]. This operator allows to extend the definition of rational languages of words, and to prove a general Kleene-like theorem.

Words indexed by a countable linear ordering were first considered in [10], where they were introduced as frontiers of labeled binary trees. Some kind of rational expressions were studied in [10, 14, 23], which lead to a characterization of words which are frontiers of regular trees.

Other related works can be found in the area of specification and verification of real-time systems. Indeed, words indexed by $\mathbb{R}$ (or other linear orderings) appear as a simple and natural way to model the behavior of a finite state real-time system. For example, ordinal words (called Zeno words) were recently considered as modeling infinite sequences of actions which occur in a finite interval of time [13, 2]. While the intervals of time are finite, infinite sequences of actions can be concatenated. A Kleene's theorem already exists for standard timed automata (where infinite sequences of actions are supposed to generate divergent sequences of times) [1]. In [2], automata considered by Choueka and Wojciechowski are adapted to Zeno words. A kind of Kleene's theorem is proved, that is, the class of Zeno languages is the closure under an operation called refinement of the class of languages accepted by standard timed automata. More recently, Dima introduces a notion of real-time automata [11] that captures a class of timed languages which is closed under complementation and for which a Kleene's theorem is proved. Let us finally mention the paper [19] which introduces star-free expressions for words indexed by $\mathbb{R}$, and shows that star-free languages of words indexed by $\mathbb{R}$ coincide with languages definable in some first-order logic, extending McNaughton-Papert theorem.

The paper is organized as follows: we recall in Sect. 2 some useful definitions related to linear orderings. Sections 3 and 4 respectively introduce rational expressions and automata for words over linear orderings. Section 5 states the main theorem and provides a few examples.

2 Linear Orderings

In this section we recall useful definitions and results about linear orderings. A good reference on the subject is Rosenstein's book [21].

A *linear ordering* J is an ordering $<$ which is total, that is, for any $j \neq k$ in J, either $j < k$ or $k < j$ holds. Given a linear ordering J, we denote by $-J$ the *backwards* linear ordering obtained by reversing the ordering relation. For instance, $-\omega$ is the backwards linear ordering of ω which is used to index the so-called left-infinite words.

The sum of orderings is concatenation. let J and K_j for $j \in J$, be linear orderings. The linear ordering $\sum_{j\in J} K_j$ is obtained by juxtaposition of the orderings K_j with respect to J. More formally, the *sum* $\sum_{j\in J} K_j$ is the set L of all pairs (k,j) such that $k \in K_j$. The relation $(k_1,j_1) < (k_2,j_2)$ holds iff $j_1 < j_2$ or ($j_1 = j_2$ and $k_1 < k_2$ in K_{j_1}). The sum of two orderings K_1 and K_2 is denoted $K_1 + K_2$.

Two elements j and k of a linear ordering J are called *consecutive* if $j < k$ and if there is no element $i \in J$ such that $j < i < k$. An ordering is *dense* if it contains no pair of consecutive elements. More generally, a subset $K \subset J$ is *dense* in J if for any $j, j' \in J$ such that $j < j'$, there is $k \in K$ such that $j < k < j'$.

The notion of a cut is needed to define a path in an automaton. A *cut* of a linear ordering J is a pair (K,L) of intervals such that $J = K \cup L$ and such that for any $k \in K$ and $l \in L$, $k < l$. The set of all cuts of the ordering J is denoted by $\hat{J}$. This set $\hat{J}$ can be linearly ordered by the relation defined by $c_1 < c_2$ iff $K_1 \subsetneq K_2$ for any cuts $c_1 = (K_1,L_1)$ and $c_2 = (K_2,L_2)$. This linear ordering can be extended to $J \cup \hat{J}$ by setting $j < c_1$ whenever $j \in K_1$ for any $j \in J$.

The consecutive elements of $\hat{J}$ deserve some attention. For any element j of J, define two cuts c_j^- and c_j^+ by $c_j^- = (K, \{j\} \cup L)$ and $c_j^+ = (K \cup \{j\}, L)$ where $K = \{k \mid k < j\}$ and $L = \{k \mid j < k\}$. It can be easily checked that the pairs of consecutive elements of $\hat{J}$ are the pairs of the form (c_j^-, c_j^+).

An ordering J is *complete* if for any cut (K,L) such that $K \neq \varnothing$ and $L \neq \varnothing$, either K has a greatest element or L has a least element.

3 Words and rational expressions

Given a finite alphabet A, a *word* $(a_j)_{j\in J}$ is a function from J to A which maps any element j of J to a letter a_j of A. We say that J is the *length* $|x|$ of the word x. For instance, the *empty word* ε is indexed by the empty linear ordering $J = \varnothing$. Usual finite words are the words indexed by finite orderings $J = \{1, 2, \ldots, n\}$, $n \geq 0$. A word of length $J = \omega$ is usually called an ω-word or an infinite word. A word of length $\zeta = -\omega + \omega$ is a sequence $\ldots a_{-2}a_{-1}a_0a_1a_2\ldots$ of letters which is usually called a bi-infinite word.

The sum operation on linear orderings leads to a notion of product of words as follows. Let J and K_j for $j \in J$, be linear orderings. Let $x_j = (a_{k,j})_{k\in K_j}$ be a word of length K_j, for any $j \in J$. The *product* $\prod_{j\in J} x_j$ is the word z of length $L = \sum_{j\in J} K_j$ equal to $(a_{k,j})_{(k,j)\in L}$. For instance, the word $a^\zeta = b^{-\omega}a^\omega$ of length ζ is the product of the two words $b^{-\omega}$ and a^ω of length $-\omega$ and ω respectively.

We now recall the notion of rational set of words on linear orderings as defined in [3]. The rational operations include of course the usual Kleene operations for finite words which are the union $+$, the concatenation $\cdot$ and the star operation $*$. They also include the omega iteration ω usually used to construct ω-words and the ordinal iteration $\sharp$ introduced by Wojciechowski [26] for ordinal words. Four new operations are also needed: the backwards omega iteration $-\omega$, the backwards ordinal iteration $-\sharp$, a binary operation denoted $\diamond$ which is a kind of iteration for all orderings, and finally a shuffle operation which allows to deal with dense linear orderings.

Let us recall the already known rational operations. We respectively denote by $\mathcal{N}$, $\mathcal{O}$ and $\mathcal{L}$ the classes of finite orderings, the class of all ordinals and the class of all linear orderings. For an ordering J, we denote by $\hat{J}^*$ the set $\hat{J} \setminus \{(\varnothing, J), (J, \varnothing)\}$ where $(\varnothing, J)$ and $(J, \varnothing)\}$ are the first and last cut. Given two sets X and Y of words, define

$$\begin{aligned}
X + Y &= \{z \mid z \in X \cup Y\}, \\
X \cdot Y &= \{x \cdot y \mid x \in X, y \in Y\}, \\
X^* &= \{\textstyle\prod_{j \in \{1,\ldots,n\}} x_j \mid n \in \mathcal{N}, x_j \in X\}, \\
X^\omega &= \{\textstyle\prod_{j \in \omega} x_j \mid x_j \in X\}, \\
X^{-\omega} &= \{\textstyle\prod_{j \in -\omega} x_j \mid x_j \in X\}, \\
X^\sharp &= \{\textstyle\prod_{j \in \alpha} x_j \mid \alpha \in \mathcal{O}, x_j \in X\}, \\
X^{-\sharp} &= \{\textstyle\prod_{j \in -\alpha} x_j \mid \alpha \in \mathcal{O}, x_j \in X\}, \\
X \diamond Y &= \{\textstyle\prod_{j \in J \cup \hat{J}^*} z_j \mid J \in \mathcal{L}, z_j \in X \text{ if } j \in J \text{ and } z_j \in Y \text{ if } j \in \hat{J}^*\}.
\end{aligned}$$

We use the notation $X^\diamond$ as an abbreviation for $(X \diamond \varepsilon) + \varepsilon$.

We now define the new shuffle operation which is needed to deal with dense orderings.

Definition 1. *Let A be a finite alphabet, $n \geq 1$, and $L_1, \ldots, L_n \subseteq A^\diamond$. We define the shuffle of $L_1, \ldots, L_n$, and denote by $\mathrm{sh}(L_1, \ldots, L_n)$ the set of words $w \in A^\diamond$ that can be written as $w = \prod_{j \in J} w_j$ where*

- *J is a complete linear ordering without first and last element;*
- *there exists a partition $(J_1, \ldots, J_n)$ of J such that all J_i's are dense in J, and for every $j \in J$, if $j \in J_k$ then $w_j \in L_k$.*

Let us remark that our definition of shuffle slightly differs from others, e.g. from [14, 23], because in Definition 1 we assume that J is a *complete* dense ordering.

Only countable orderings are considered in [14, 23]. Recall that $\mathbb{Q}$ is the unique countable and dense ordering without first and last element. Their definition of the shuffle operation is based on a partition $(J_1, \ldots, J_n)$ of $\mathbb{Q}$ into dense subsets $J_1, \ldots, J_n$. Then points of each J_i are substituted by words from L_i as we do. Our definition is not a straightforward generalization of this shuffle because $\mathbb{Q}$ is of course not complete. Actually the assumption that J is complete yields a more general notion of shuffle. The completion of $\mathbb{Q}$ yields the ordering $\mathbb{R}$. If each point of $\mathbb{R} \setminus \mathbb{Q}$ is substituted by the empty word, one obtains a shuffle in the sense of [14, 23]. This shows that our rational expression $\mathrm{sh}(L_1, \ldots, L_n, \varepsilon)$ corresponds to the shuffle of languages $L_1, L_2, \ldots, L_n$ in the sense of [14, 23].

An abstract *rational expression* is a well-formed term of the free algebra over $\{\varnothing\} \cup A$ with the symbols denoting the rational operations as function symbols. Each rational expression denotes a set of words which is inductively defined by the above definitions of the rational operations. A set of words is *rational* if it can be denoted by a rational expression. As usual, the dot denoting concatenation is omitted in rational expressions.

Example 1. Consider the word $w = (w_r)_{r \in \mathbb{R}}$ of length $\mathbb{R}$ over the alphabet $A = \{a, b\}$, defined by $w_r = a$ if $r \in \mathbb{Q}$, and $w_r = b$ otherwise. Then it is not difficult to check that $w \in \mathrm{sh}(a, b)$. Consider now the word $w' = (w'_q)_{q \in \mathbb{Q}}$ of length $\mathbb{Q}$ over the alphabet A, defined by $w'_q = a$ if $q \in \{m/2^n \mid m \in \mathbb{Z}, n \in \mathbb{N}\}$, and $w'_q = b$ otherwise. It can be checked that $w' \in \mathrm{sh}(a, b, \varepsilon)$ but that $w' \notin \mathrm{sh}(a, b)$ because $\mathbb{Q}$ is not complete.

Example 2. The rational expression $a^*(\varepsilon + \mathrm{sh}(a^*, \varepsilon))a^*$ denotes the set of words (over the unary alphabet $\{a\}$) whose length is an ordering containing no infinite sequence of consecutive elements. It is clear that the length of any word denoted by this expression cannot contain an infinite sequence of consecutive elements. Conversely, let J be such an ordering. Define the equivalence relation $\sim$ on J by $x \sim y$ iff there are finitely many elements between x and y. The classes of $\sim$ are then finite intervals. Furthermore the ordering of these intervals must be a dense ordering with possibly a first and a last element. This completes the converse.

Example 3. The rational expression $(\varepsilon + \mathrm{sh}(a)) \diamond a$ denotes the set of words (over the unary alphabet $\{a\}$) whose length is a complete ordering. The shuffle operator is defined using complete orderings and the ordering $\hat{J}$ is always complete. It follows from these two facts that the length of any word denoted by this expression is complete. Conversely, let J be a complete orderings. Define the equivalence relation $\sim$ on J by $x \sim y$ iff there is an open dense interval containing both x and y. Each class of $\sim$ is either a singleton or an open dense interval. Let K be the ordering of the singleton classes and let L_0 be the ordering of the dense classes. Let L_1 be the ordering of pairs of consecutive elements in K and let L be $L_0 \cup L_1$ equipped with the natural ordering. It can be shown that $K = \hat{L}$. This gives the expression $(\varepsilon + \mathrm{sh}(a)) \diamond a$ where ε is due to L_1, $\mathrm{sh}(a)$ to L_0 and a to K. This completes the converse.

4 Automata

In this section, we recall the definition given in [3] for automata accepting words on linear orderings. As already noted in [3], this definition is actually suitable for all linear orderings.

Automata accepting words on linear orderings are classical finite automata equipped with limit transitions. They are defined as $\mathcal{A} = (Q, A, E, I, F)$, where Q denotes the finite set of states, A is a finite alphabet, and I, F denote the set of initial and final states, respectively. The set E consists in three types of transitions: the usual *successor* transitions in $Q \times A \times Q$, the *left limit* transitions which belong to $2^Q \times Q$ and the *right limit* transitions which belong to $Q \times 2^Q$. A left (respectively right) limit transition $(P, q) \in 2^Q \times Q$ (respectively, $(q, P) \in Q \times 2^Q$) will usually be denoted by $P \to q$ (respectively $q \to P$).

We sometimes write that an automaton $\mathcal{A}$ has transitions $P_1, \ldots, P_m \to q_1, \ldots, q_n$ when $\mathcal{A}$ has all left limit transitions $P_i \to q_j$ for $1 \leq i \leq m$ and $1 \leq j \leq n$. Analogously we shall use the notation $q_1, \ldots, q_n \to P_1, \ldots, P_m$ for right limit transitions.

A word $x = (a_j)_{j \in J}$ of length J is accepted by $\mathcal{A}$ if it is the label of a successful path. A *path* γ is a sequence of states $\gamma = (q_c)_{c \in \hat{J}}$ of length $\hat{J}$ verifying the following conditions. For two consecutive states in γ, there must be a successor transition labeled by the letter in between. For a state $q \in \gamma$ which has no predecessor on γ, there must be a left limit transition $P \to q$ where P is the limit set of γ on the left of q. Right limit transitions are used similarly when q has no successor on γ.

Observe that the ordering $\hat{J}$ always has a first element and a last element, namely the cuts $c_{\min} = (\varnothing, J)$ and $c_{\max} = (J, \varnothing)$. For any cut $c \in \hat{J}$, define the sets $\lim_{c^-} \gamma$ and $\lim_{c^+} \gamma$ as follows:

$$\lim_{c^-} \gamma = \{q \in Q \mid \forall c' < c \;\; \exists k \quad c' < k < c \text{ and } q = q_k\},$$
$$\lim_{c^+} \gamma = \{q \in Q \mid \forall c < c' \;\; \exists k \quad c < k < c' \text{ and } q = q_k\}.$$

A sequence $\gamma = (q_c)_{c \in \hat{J}}$ of states is an accepting path for the word $x = (a_j)_{j \in J}$ if the following conditions are fulfilled. For any pair (c_j^-, c_j^+) of consecutive cuts of J, the automaton must have the successor transition $q_{c_j^-} \xrightarrow{a_j} q_{c_j^+}$. For any cut $c \neq c_{\min}$ which has no predecessor in $\hat{J}$, $\lim_{c^-} \gamma \;\to\; q_c$ must be a left limit transition. For any cut $c \neq c_{\max}$ in $\hat{J}$ which has no successor, $q_c \to \lim_{c^+} \gamma$ must be a right limit transition. A path is *successful* if its first state $q_{c_{\min}}$ is initial and its last state $q_{c_{\max}}$ is final.

Example 4. Let $A = \{a, b\}$. The automata $\mathcal{A}$ pictured in Fig. 1 has two successor transitions, three left limit transitions and three right limit transitions. State 0 is the only initial state, and state 5 is the only final state.

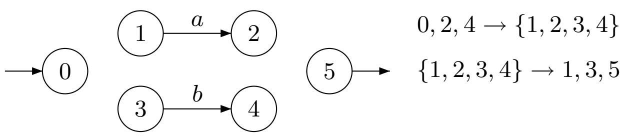


Fig. 1. Automaton accepting $\mathrm{sh}(a, b)$

Let us show that this automata accepts words in $\mathrm{sh}(a, b)$. Consider indeed a word $w = (w_j)_{j \in J}$, and assume first that w is accepted by $\mathcal{A}$. Let $\gamma = (q_c)_{c \in \hat{J}}$ be a successful path labeled by w. The ordering J must be dense since there are no consecutive transitions in $\mathcal{A}$. It must also be complete since there is no state with incoming left limit transitions and leaving right limit transitions. Occurrences of both a and b must be dense in J since all limit transitions involve the four states $\{1, 2, 3, 4\}$. Finally, J cannot have a first or a last element. Indeed, the only transition leaving state 0 is a limit one, and similarly for the only transition entering state 5.

Conversely, let $w = (w_j)_{j \in J}$ be a word indexed by a complete ordering J such that occurrences of both a and b are dense in J. Since J is complete, any cut of J (apart from $c_{\min}$ and $c_{\max}$) are either preceded or followed by a letter. Then the sequence $\gamma = (q_c)_{c \in \hat{J}}$ defined as follows is a successful path labeled by w.

- $q_{c_{\min}} = 0$,
- $q_{c_{\max}} = 5$,
- $q_c = 1$ if c is followed by an a and $q_c = 2$ if c is preceded by an a,
- $q_c = 3$ if c is followed by a b and $q_c = 4$ if c is preceded by a b.

The construction of an automaton accepting $\mathrm{sh}(L_1, \ldots, L_n)$ from automata accepting $L_1, \ldots, L_n$ is a straightforward generalization of the automaton for $\mathrm{sh}(a, b)$ which is pictured in Fig. 1.

5 Rational expressions vs automata

In this section we state the main theorem of the paper, i.e. that automata and rational expressions define the same languages of words over linear orderings.

Theorem 1. *A set of words over linear orderings is rational iff it is recognizable.*

This result was proved in [3] for the restricted case of countable scattered linear orderings. Many arguments still hold in the general case. As in [3], the *only if* part of the proof relies upon an induction on the rational expression. The only modification with respect to the proof of [3] is that we have to show that the shuffle of rational languages is rational. For the *if* part of the proof, we use as in [3] Yamada's classical technique, i.e. an induction on the set of states visited by a successful path of the automaton. A key ingredient is the use of successive condensations of linear orderings.

We illustrate the theorem with a few examples, over the alphabet $A = \{a, b\}$.

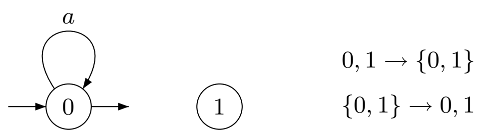
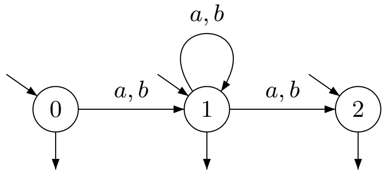


Fig. 2. Automaton accepting $a^*(\varepsilon + \mathrm{sh}(a^*, \varepsilon))a^*$

Example 5. The automaton pictured in Fig. 2 accepts the set $a^*(\varepsilon + \mathrm{sh}(a^*, \varepsilon))a^*$ of words already considered in Example 2.

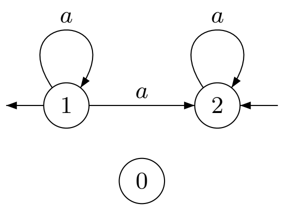


$2 \to \{1\}, \{0,1\}, \{1,2\}, \{0,1,2\}$

$\{1\}, \{0,1\}, \{1,2\}, \{0,1,2\} \to 0$

Fig. 3. Automaton accepting words with a complete length

Example 6. The automaton pictured in Fig. 3 accepts words over $\{a, b\}$ whose length is a complete ordering. Since all limit transitions enter state 0 and leave state 2, the length of an accepted word must be complete. Conversely, let x be a word of length J where J is a complete ordering. Define the path γ which maps any cut (K, L) of $\hat{J}$ to 0 if K has no greatest element, to 2 if L has no least element and to 1 otherwise. It is pure routine to check that this defines an accepting path for x.



$1 \to \{1\}$

$\{1\} \to 1$

$2 \to \{2\}$

$\{2\} \to 2$

$0, 2 \to \{1,2\}, \{0,1,2\}$

$\{1,2\}, \{0,1,2\} \to 0, 1$

Fig. 4. Automaton accepting words with a non scattered length

Example 7. The automaton pictured in Fig. 4 accepts words over $\{a\}$ whose length is a non scattered ordering. Let $w = (w_j)_{j \in J}$ be a word labeling a successful path γ in $\mathcal{A}$. Let K be the set of positions j such that w_j is read by the transition $1 \xrightarrow{a} 2$ in γ. It can be checked that K is a dense subordering of J. Therefore, the ordering J is not scattered.

For the converse, recall that each linear ordering J can be written $J = \sum_{k \in K} J_k$ where each ordering J_k is scattered and the ordering K is either the one-element ordering if J is scattered or a dense ordering [21, chap. 4]. From this decomposition, a word w whose length is not scattered is equal to a product $\prod_{k \in K} w_k$ where K is a dense ordering. Then a path γ_k from state 1 to state 2 labeled by w_k can be constructed as follows. Let J_k be the length of w_k and let z_k be an arbitrarily chosen element of J_k. Any cut of J_k before z_k is mapped to state 1 and any cut after z_k is mapped to state 2. A path labeled by w is finally constructed as follows. Any cut inside some w_k is mapped to the corresponding state in γ_k and any remaining gap is mapped to state 0.

6 Conclusion and open questions

We considered rational expressions and automata for words indexed by linear orderings, and prove that these two formalisms capture the same languages.

A natural question is whether the class of recognizable languages is closed under complementation. It has been proved in [20] that the answer is positive when one considers only words indexed by countable and scattered orderings (the proof relies upon semigroup theory). However the answer is negative in the general case: one can prove that the set of words indexed by a non scattered ordering is recognizable, while its complement is not.

The connections between automata over linear orderings and logic would be interesting to explore. In his seminal paper [5], Büchi proved that recognizable languages of finite words coincide with languages definable in the weak monadic second order theory of $(\omega, <)$, which allowed him to prove decidability of this theory. In [6] he proved that a similar equivalence holds between recognizable languages of infinite words of length ω and languages definable in the monadic second order theory of $(\omega, <)$. The result was then extended to languages of words indexed by a countable ordinal [7]. What can be said about monadic second order theories for linear orderings beyond ordinals ? Using the automata technique, Rabin proved in [18] decidability of the monadic second order theory of the binary tree, from which he deduces decidability of the monadic second order theory of $\mathbb{Q}$, which in turn implies decidability of the monadic second order theory of countable linear orderings. On the other hand, Shelah [22] improved model-theoretical techniques [24] that allow him to reprove almost all known decidability results about monadic second order theories, as well as new decidability results for the case of linear orderings. On the other hand he proved that the monadic second order theory of the real line is undecidable. Shelah's decidability method is model-theoretical, and up to now no corresponding automata techniques are known. This led Thomas to ask [24] whether there is an appropriate notion of automata for words indexed by linear orderings beyond the ordinals. As mentioned in [3], this question was an important motivation for the introduction of automata considered in the present paper. It would be interesting to provide a logical characterization of recognizable languages of words over linear orderings.

References

1. E. Asarin, P. Caspi, and O. Maler. A Kleene theorem for timed automata. In *Proceedings, Twelfth Annual IEEE Symposium on Logic in Computer Science*, pages 160–171, 1997.
2. B. Bérard and C. Picaronny. Accepting Zeno words without making time stand still. In *Mathematical Foundations of Computer Science 1997*, volume 1295 of *Lect. Notes in Comput. Sci.*, pages 149–158, 1997.

3. V. Bruyère and O. Carton. Automata on linear orderings. In J. Sgall, A. Pultr, and P. Kolman, editors, *MFCS'2001*, volume 2136 of *Lect. Notes in Comput. Sci.*, pages 236–247, 2001.
4. V. Bruyère and O. Carton. Hierarchy among automata on linear orderings. In R. Baeza-Yate, U. Montanari, and N. Santoro, editors, *Foundation of Information technology in the era of network and mobile computing*, pages 107–118. Kluwer Academic Publishers, 2002.
5. J. R. Büchi. Weak second-order arithmetic and finite automata. *Z. Math. Logik und grundl. Math.*, 6:66–92, 1960.
6. J. R. Büchi. On a decision method in the restricted second-order arithmetic. In *Proc. Int. Congress Logic, Methodology and Philosophy of science, Berkeley 1960*, pages 1–11. Stanford University Press, 1962.
7. J. R. Büchi. Transfinite automata recursions and weak second order theory of ordinals. In *Proc. Int. Congress Logic, Methodology, and Philosophy of Science, Jerusalem 1964*, pages 2–23. North Holland, 1965.
8. O. Carton. Accessibility in automata on scattered linear orderings. In K.Diks and W.Rytter, editors, *MFCS'2002*, volume 2420 of *Lect. Notes in Comput. Sci.*, pages 155–164, 2002.
9. Y. Choueka. Finite automata, definable sets, and regular expressions over ω^n-tapes. *J. Comput. System Sci.*, 17(1):81–97, 1978.
10. B. Courcelle. Frontiers of infinite trees. *RAIRO Theoretical informatics*, 12(4):319–337, 1978.
11. C. Dima. Real-time automata. *Journal of Automata, Languages and Combinatorics*, 6(1):3–24, 2001.
12. D. Girault-Beauquier. Bilimites de langages reconnaissables. *Theoret. Comput. Sci.*, 33(2–3):335–342, 1984.
13. M. R. Hansen, P. K. Pandya, and Z. Chaochen. Finite divergence. *Theoret. Comput. Sci.*, 138(1):113–139, 1995.
14. S. Heilbrunner. An algorithm for the solution of fixed-point equations for infinite words. *RAIRO Theoretical informatics*, 14(2):131–141, 1980.
15. S. C. Kleene. Representation of events in nerve nets and finite automata. In C.E. Shannon, editor, *Automata studies*, pages 3–41. Princeton university Press, Princeton, 1956.
16. D. E. Muller. Infinite sequences and finite machines. In *Switching Circuit Theory and Logical Design: Proc. Fourth Annual Symp.*, pages 3–16. I.E.E.E., New York, 1963.
17. M. Nivat and D. Perrin. Ensembles reconnaissables de mots bi-infinis. In *Proceedings of the Fourteenth Annual ACM Symposium on Theory of Computing*, pages 47–59, 1982.
18. M.O. Rabin. Decidability of second-order theories and automata on infinite trees. *Transactions of the American Mathematical Society*, 141:1–35, 1969.
19. A. Rabinovich. Star free expressions over the reals. *Theoret. Comput. Sci.*, 233(1–2):233–245, 2000.
20. C. Rispal and O. Carton. Complementation of rational sets on countable scattered linear orderings. In C. S. Calude, E. Calude, and M. J. Dinneen, editors, *DLT'2004*, volume 3340 of *Lect. Notes in Comput. Sci.*, pages 381–392, 2004.
21. J. G. Rosenstein. *Linear orderings.* Academic Press, New York, 1982.
22. S. Shelah. The monadic theory of order. *Annals of Mathematics*, 102:379–419, 1975.
23. W. Thomas. On frontiers of regular sets. *RAIRO Theoretical informatics*, 20:371–381, 1986.

24. W. Thomas. Ehrenfeucht games, the composition method, and the monadic theory of ordinal words. In *Structures in Logic and Computer Science, A Selection of Essays in Honor of A. Ehrenfeucht*, number 1261 in Lect. Notes in Comput. Sci., pages 118–143. Springer-Verlag, 1997.
25. O. Carton V. Bruyère and G. Sénizergues. Tree automata and automata on linear orderings. In T. Harju and J. Karhumäki, editors, *WORDS'2003*, pages 222–231. Turku Center for Computer Science, 2003.
26. J. Wojciechowski. Finite automata on transfinite sequences and regular expressions. *Fundamenta informaticæ*, 8(3-4):379–396, 1985.