On recognizable trace languages

Manfred Kufleitner, Pascal Weil

On recognizable trace languages

Manfred Kufleitner and Pascal Weil
LaBRI (Université de Bordeaux and CNRS)

Traces are a model for concurrency, whose formal definition is due to Mazurkiewicz [9]. In short, an alphabet A is fixed, with an independence relation on A. Traces can be defined as equivalence classes of words on A, under the relation that allows commutation of consecutive independent letters; and they are uniquely represented by certain A-labeled posets. A trace language is a set of traces on a fixed independence alphabet.

The notion of regularity for trace languages is derived naturally from the standard framework of formal language theory (the theory of languages of finite words): a trace language is regular if the set of linearizations of its elements is a regular (word) language. The robustness of this notion is established by two fundamental results: regular trace languages are exactly the sets of traces that are defined by monadic second-order sentences [19]; they are exactly the sets of traces that are accepted by the so-called Zielonka automata [20]. These automata capture, in their structure, the intrinsically distributed nature of the alphabet (with its independence relation) and of the language. A difference with classical language theory arises when rational expressions are considered: if letters a and b are independent, then $(ab)^*$ is not a regular trace language (the set of its linearizations is the archetypal context-free language of all words with an equal number of a's and b's), but a result by Ochmański [12] gives a restriction of rational expressions that describe exactly the regular trace languages.

On the basis of these encouraging results, it was natural to think that the algebraic point of view – that is so successful in automata theory – would also help classifying regular trace languages. Since the set of all traces is naturally endowed with a monoid structure, it makes sense to consider the recognizable trace languages, that is, those that are accepted by a morphism into a finite monoid, or equivalently, those whose syntactic monoid is finite. It turns out (and it is not difficult to establish) that those are exactly the regular trace languages. It was shown (Guaiana, Restivo, Salemi [6] and Ebinger, Muscholl [4]) that star-free expressions are as expressive as first-order formulas, and that the corresponding trace languages are exactly those whose syntactic monoid is aperiodic. The same statement also holds for word languages (Schützenberger [15] and McNaughton, Papert [11]) and it was one of the results that led to Eilenberg's intuition of the theory of varieties. This theory is based on the idea that certain combinatorial properties of recognizable languages are reflected in the algebraic properties of their syntactic monoids. More precisely, Eilenberg introduced varieties of languages: these are classes of recognizable languages closed under Boolean operations, left and right residuals and inverse morphisms. He also considered the classical notion of pseudovarieties of monoids (classes of finite monoids that are closed under taking submonoids, quotients and finite direct product) and he showed that (a) the languages whose syntactic monoid sits in a given pseudovariety $\mathbf{V}$ form a variety of languages $\mathcal{V}$, and (b) that the resulting correspondence $\mathbf{V} \to \mathcal{V}$ is one-to-one and onto between pseudovarieties of monoids and varieties of languages [5]. This gave a conceptual framework in which Schützenberger's theorem fits nicely [15], but also Simon's theorem on piecewise testable languages [16], Brzozowski, Simon and McNaughton's result on locally testable languages [2, 10], and a host of comparable results. Each of these results establishes that a natural class of recognizable languages is a variety, identifies the associated pseudovariety of monoids, and derives from it the decidability of the language class.

The theory was later extended by considering operations on varieties of languages and on pseudovarieties of monoids. For instance Straubing showed that if $\mathbf{V} \to \mathcal{V}$, then the closure of $\mathcal{V}$ under concatenation corresponds to the pseudovariety $\mathbf{A} \circledM \mathbf{V}$, where $\circledM$ designates the Mal'cev product and $\mathbf{A}$ is the pseudovariety of aperiodic monoids [17]. Variants of the theory also allow the consideration of classes of recognizable languages that do not satisfy all the closure properties of varieties. Thus Pin showed that positive varieties of languages (classes of languages closed under the same operations as varieties except for complementation) are in one-to-one correspondence with pseudovarieties of ordered monoids [13]. This allowed for instance the algebraic characterization of the operation of polynomial closure of a variety of languages (Pin, Weil [14]). In a different direction, Straubing explored the so-called $\mathcal{C}$-varieties, defined like varieties but restricting the closure under inverse morphisms to morphisms in a fixed class $\mathcal{C}$ [18].

There were few attempts to carry such results to trace languages. The most successful is due to Kufleitner [7, 8] who gave several characterizations, logical and combinatorial of the trace languages recognized by the pseudovariety $\mathbf{DA}$. These results generalize analogous statements on word languages, but in the latter framework, they follow from a set of more general results, notably on the polynomial closure of varieties of languages. In the case of trace languages, these more general results could not be extended in a satisfactory fashion, and there did not seem to be a clear route to achieve the characterization (and the decidability) of more natural classes of trace languages (e.g. those that are definable in the first-order theory of successor). In fact, no satisfactory theorem stating an analogue of Eilenberg's variety theorem was known so far, and all efforts to carry the spirit of this theory to trace languages proved ineffective.

One key reason for this, in our view, is that the mere consideration of finite monoids to characterize the properties of trace languages is not sufficient. In a general sense, this amounts to considering the trace monoid as *just a monoid*, whereas it also contains important information on dependence and independence; and at a very practical level (*i.e.*, in proofs), the trace monoid is not a free monoid, which leads to unsurpassable technical hurdles in proofs that are completely elementary in the word language case.

This leads us to propose a new algebraic framework for studying recognizable trace languages, which we call *independence monoids*, or *I-monoids*. The idea is to consider that, in the set of all traces, the notion of independence is as important a feature as the associative nature of concatenation, that is, as the monoid structure. In general, an I-monoid is a monoid equipped with a symmetric binary relation I, called the independence relation, subject to a few axioms and such that independent elements must commute (but no converse is required). A morphism between I-monoids must preserve the monoid structure, and it must also map independent elements to independent elements. Whether the converse holds or not defines what we call *strong* morphisms[1]. The notion of congruence and quotient is similarly adapted to I-monoids, and pseudovarieties of I-monoids can be naturally defined. The first important result, or observation, is that the class of trace languages that are algebraically recognized by finite I-monoids is exactly the traditional class of regular trace languages, so that it makes sense to study these languages by looking at the properties of their syntactic I-monoids. In fact, I-monoids already occurred implicitly in the literature on traces [3]. For instance, the transition monoid of a Zielonka automaton can be turned very naturally into an I-monoid recognizing L. Moreover, the usual construction (in proofs) of attaching the alphabet (that is, of considering images of traces in the direct product of a finite monoid and the alphabetic content monoid) yields an I-monoid. The advantage of the abstract notion of I-monoids is to avoid being restricted to a fixed independence alphabet, while still providing all the relevant information.

Our first results are foundational: we define varieties of trace languages and we establish the statement of an Eilenberg-like variety theorem. We also extend the equational approach to the description of pseudovarieties of I-monoids. We are then able to describe a few simple varieties of trace languages. An important remark is that the usual word languages can be considered as particular trace languages: on an alphabet with an empty independence relation. Similarly a monoid can be viewed as a particular I-monoid, with a trivial independence relation. In particular, the syntactic I-monoid of a word language essentially coincides with its usual syntactic monoid. In this sense, our result is a generalization of Eilenberg's theorem.

Our second set of results requires a bit more machinery. It is a generalization of Pin and Weil's results on the polynomial closure of varieties of languages, and of Kufleitner's results mentioned above. With the definition of I-monoids and of the corresponding varieties and pseudovarieties, the proofs from Pin and Weil now carry over without major difficulty. One application of this is to give an algebraic characterization of the class of trace languages defined by first-order formulas in Σ_n, that are similar to the word case. Of course, as in the word case, we do not know whether these classes are decidable.

We are currently working on a third set of results, involving the wreath product of I-monoids, and such applications as the characterization and decidability of the trace languages defined by formulas from the first-order theory of successor.

[1] The same notion of strong morphisms was considered for morphisms between trace monoids by Bruyère, de Felice and Guaiana [1] in the context of the study of trace coding.

References

[1] Véronique Bruyère, Clelia De Felice and Giovanna Guaiana. Coding with traces. In P. Enjalbert, E. W. Mayr and K. W. Wagner, eds, *Proc. STACS'94*, Lecture Notes in Computer Science **775**, pp. 353–364. Springer, 1994.

[2] Janusz A. Brzozowski and Imre Simon. Characterizations of locally testable events. *Discrete Math.*, 4:243–271, 1973.

[3] Volker Diekert. *The book of traces*. World Scientific, Singapore, 1994.

[4] Werner Ebinger and Anca Muscholl. Logical definability on infinite traces. *Theoretical Computer Science*, 154:67–84, 1996.

[5] Samuel Eilenberg. *Automata, Languages, and Machines*, volume B. Academic Press, New York and London, 1976.

[6] Giovanna Guaiana, Antonio Restivo, and Sergio Salemi. Star-free trace languages. *Theoretical Computer Science*, 97:301–311, 1992.

[7] Manfred Kufleitner. *Logical Fragments for Mazurkiewicz Traces: Expressive Power and Algebraic Characterizations*. Dissertation, Institut für Formale Methoden der Informatik, Universität Stuttgart, 2006.

[8] Manfred Kufleitner. Polynomials, fragments of temporal logic and the variety DA over traces. *Theoretical Computer Science*, 376:89–100, 2007. Special issue DLT 2006.

[9] Antoni Mazurkiewicz. Concurrent program schemes and their interpretations. DAIMI Rep. PB 78, Aarhus University, Aarhus, 1977.

[10] Robert McNaughton. Algebraic decision procedures for local testability. *Mathematical Systems Theory*, 8(1):60–76, 1974.

[11] Robert McNaughton and Seymour Papert. *Counter-Free Automata*. The MIT Press, Cambridge, Mass., 1971.

[12] Edward Ochmański. Regular behaviour of concurrent systems. *Bulletin of the European Association for Theoretical Computer Science*, 27:56–67, 1985.

[13] Jean-Éric Pin. A variety theorem without complementation. In *Russian Mathematics (Izvestija vuzov.Matematika)*, volume 39, pages 80–90, 1995.

[14] Jean-Éric Pin and Pascal Weil. Polynominal closure and unambiguous product. *Theory Comput. Syst*, 30(4):383–422, 1997.

[15] Marcel Paul Schützenberger. On finite monoids having only trivial subgroups. *Information and Control*, 8:190–194, 1965.

[16] Imre Simon. Piecewise testable events. In H. Barkhage, ed., *Automata Theory and Formal Languages, 2nd GI Conference*, Lecture Notes in Computer Science **33**, pp. 214–222. Springer-Verlag, 1975.

[17] Howard Straubing. Aperiodic homomorphisms and the concatenation product of recognizable sets. *Journal of Pure and Applied Algebra*, 15(3):319–327, 1979.

[18] Howard Straubing. On the logical descriptions of regular languages. In S. Rajsbaum, ed., *Proc. LATIN'02*, Lecture Notes in Computer Science **2286**, pp. 528–538. Springer, 2002.

[19] Wolfgang Thomas. On logical definability of trace languages. In V. Diekert, ed., *Proceedings of a workshop of the ESPRIT Basic Research Action No 3166: Algebraic and Syntactic Methods in Computer Science (ASMICS))*, Report TUM-I9002, Technical University of Munich, pages 172–182, 1990.

[20] Wiesław Zielonka. Notes on finite asynchronous automata. *R.A.I.R.O. — Informatique Théorique et Applications*, 21:99–135, 1987.