Theoretical Computer Science 610 (2016) 13–23

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

A survey on the local divisor technique

Volker Diekert, Manfred Kufleitner *,[1]

University of Stuttgart, FMI, Germany

ARTICLE INFO

Article history:
Received 1 November 2014
Received in revised form 30 June 2015
Accepted 4 July 2015
Available online 10 July 2015

Keywords:
Local divisors
Factorization forests
Bounded synchronization delay
Linear temporal logic
Kamp's theorem

ABSTRACT

Local divisors allow a powerful induction scheme on the size of a monoid. We survey this technique by giving several examples of this proof method. These applications include linear temporal logic, rational expressions with Kleene stars restricted to prefix codes with bounded synchronization delay, Church–Rosser congruential languages, and Simon's Factorization Forest Theorem. We also introduce the notion of a *localizable language class* as a new abstract concept which unifies some of the proofs for the results above.

1. Introduction

The notion of a *local divisor* refers to a construction for finite monoids. It appeared in this context first in [4] where it was used by the authors as a tool in the proof that local future temporal logic is expressively complete for Mazurkiewicz traces with respect to first-order logic. The definition of a local divisor is very simple: Let M be a finite monoid and $c \in M$. Then $cM \cap Mc$ is a semigroup, but it fails to be a submonoid unless c is invertible. If c is not invertible then $1 \notin cM \cap Mc$ and, as a consequence, $|cM \cap Mc| < |M|$. The idea is to turn $cM \cap Mc$ into a monoid by defining a new multiplication by $xc \circ cy = xcy$. This is well-defined and $M_c = (cM \cap Mc, \circ, c)$ becomes a monoid where c is the unit. Moreover, if c is not invertible then M_c is a smaller monoid than M; and this makes the construction attractive for induction. (The same idea works for $\{c\} \cup cMc$ and since $\{c\} \cup cMc \subseteq cM \cap Mc$ there is a choice here.) The original definition for a multiplication of type $xc \circ cy = xcy$ was given for associative algebras. It can be traced back to a technical report of Meyberg, [17]. He coined the notion of a *local algebra*. Just replace M above by a finite dimensional associative algebra (with a unit element) over a field k. For example, M is the algebra of $n \times n$ matrices over k. If $c \in M$ is not invertible then the vector space $cM \cap Mc$ has at least one dimension less and $(cM \cap Mc, +, \circ, c)$ is again an associative algebra with the unit element c. See also [11] for applications of Meyberg's construction.

Despite (or more accurately *thanks to*) its simplicity, the *local divisor technique* is quite powerful, see *e.g.* [6]. For example, it was used in a new and simplified proof for the Krohn–Rhodes Theorem [9]. Very recently, the construction of local divisors has also been an essential tool in Kuperberg's work on a linear temporal logic for regular cost functions [15]. In [7] we extended a classical result of Schützenberger from finite words to infinite words by showing that *ω-rational expressions with bounded synchronization delay* characterize star-free languages. In 2012 we presented a paper which solved a 25 years old conjecture in formal language theory [8]. We showed that regular languages are Church–Rosser congruential. We come

* Corresponding author.
E-mail addresses: diekert@fmi.uni-stuttgart.de (V. Diekert), kufleitner@fmi.uni-stuttgart.de (M. Kufleitner).

[1] The second author was supported by the German Research Foundation (DFG) grant DI 435/5-2.

http://dx.doi.org/10.1016/j.tcs.2015.07.008
0304-3975/

back to this result in more detail below. Our result was obtained in two steps. First, we had to show it for regular group languages, which is very difficult and technical. This part served as a base for induction. The second part uses induction using local divisors. This part is actually easy to explain, it will be done in Section 6.

The outline of the paper is as follows. In Section 3 we give a general framework for the local divisor technique in the context of aperiodic languages (*i.e.*, languages recognized by finite aperiodic monoids). We introduce the notion of *localizable language class* as a new abstract concept.

In the remaining sections we give four applications of the local divisor technique. In Section 4 we apply this technique to linear temporal logic, and in Section 5 it is used for a characterization of the aperiodic languages in terms of restricted rational expressions. In Section 6 we show how to apply the local divisor technique in the context of string rewrite systems. Finally, in Section 7 we give a proof of Simon's Factorization Forest Theorem; the proof is the archetype of how to apply the local divisor technique in arbitrary monoids.

2. Local divisors

We will apply the local divisor techniques mainly to monoids. However, it is instructive to place ourselves first in the slightly more general setting of semigroups. Let $S = (S, \cdot)$ be a finite semigroup. A *divisor* S' of S is a homomorphic image of a subsemigroup. Let $c \in S$ be any element and consider $cS \cap Sc$. We can turn the subset $cS \cap Sc$ into a semigroup by defining a new operation $\circ$ as follows:

$$xc \circ cy = xcy.$$

A direct calculation shows that the operation $\circ$ is well-defined and associative. Hence, $S_c = (cS \cap Sc, \circ)$ is a semigroup. In order to see that S_c is a divisor consider the following subsemigroup $S' = \{x \in S \mid cx \in Sc\}$ of S. Note that $c \in S'$. Define $\varphi : S' \to S_c$ by $\varphi(x) = cx$. It is surjective since $z \in cS \cap Sc$ implies that we can write $z = cx$ with $x \in S'$. Moreover, $cxy = cx \circ cy$ and S_c is the homomorphic image of S'. Therefore, S_c is a divisor. We call it the *local divisor at* c. We want to use S_c for induction. Therefore we characterize next when $|S_c| < |S|$. Recall that $e \in S$ is called an *idempotent* if $e^2 = e$. For every finite semigroup there is a natural number $\omega \in \mathbb{N}$ such that x^ω is idempotent for every $x \in S$, for instance $\omega = |S|!$. An element y is called a *unit* if it has a left- and right inverse, *i.e.*, if there is a neutral element $1 \in S$ and $xy = yx' = 1$ for some $x, x' \in S$ (and then we have $x = xyx' = x'$). Thus, if S contains a unit y, then it is a monoid with neutral element y^ω. We have the following result.

Proposition 2.1. *Let S be a semigroup and $S_c = (cS \cap Sc, \circ)$ be defined as above.*

(a) *If S is a monoid, then $S_c = (cS \cap Sc, \circ, c)$ is a monoid and S_c is a divisor in terms of monoids, i.e. a homomorphic image of a submonoid S' of S.*
(b) *If c is a unit of S, then $S = \{x \in S \mid cx \in Sc\}$ and $\varphi : S \to S_c$, $x \mapsto cx$ is an isomorphism of monoids.*
(c) *If S is finite and c is not a unit, then $|S_c| < |S|$.*
(d) *If $cxc = cyc$ is idempotent in S_c, then $cxcy$ and $xcyc$ are idempotent in S.*

Proof. (a): Since S is a monoid we have $1 \in S' = \{x \in S \mid cx \in Sc\}$ and S_c is the homomorphic image of the submonoid S'.
(b): Trivial.
(c): If $cS \cap Sc = S$, then we have $cS = S$ and $Sc = S$. This implies that c is a unit. Indeed, we have $c^\omega S = S = Sc^\omega$. For every element $c^\omega x \in S$ we have $c^\omega \cdot c^\omega x = c^\omega x$. Thus, c^ω is neutral and $c^{\omega-1}$ is the inverse of c, *i.e.*, c is a unit. Therefore, if c is not a unit, then $|S_c| < |S|$.
(d): We have $cxcy \cdot cxcy = \big((cxc) \circ (cyc) \circ (cxc)\big) \cdot y = cxc \cdot y$. The last equality uses the fact that $cxc = cyc$ is idempotent in S_c. The claim for $xcyc$ is symmetric. □

Remark 2.2. Note that $(\{cc\} \cup cSc, \circ)$ is a subsemigroup of $(cS \cap Sc, \circ)$. Moreover, if S is a monoid, then $(\{c\} \cup cSc, \circ, c)$ is a submonoid of $(cS \cap Sc, \circ, c)$. Hence by slight abuse of language, we might call $(\{cc\} \cup cSc, \circ)$ (resp. $(\{c\} \cup cSc, \circ, c)$) a local divisor of S, too. In addition, if $c \in S$ is idempotent, then $(cSc, \circ) = (cSc, \cdot)$ is the usual local monoid at c. The advantage is that $\{cc\} \cup cSc$ (resp. $\{c\} \cup cSc$) might be smaller than $cS \cap Sc$. However, in worst case estimations there is no difference.

3. Localizable language classes

A *language class* $\mathcal{V}$ assigns to every finite alphabet A a set of languages $\mathcal{V}(A^*) \subseteq 2^{A^*}$. A language class $\mathcal{V}$ is *left-localizable* if for all finite alphabets A and T the following properties hold:

(a) $\emptyset, A^* \in \mathcal{V}(A^*)$.
(b) If $K, L \in \mathcal{V}(A^*)$, then $K \cup L \in \mathcal{V}(A^*)$.
(c) For every $c \in A$, the alphabet $B = A \setminus \{c\}$ satisfies:
    1. If $K \in \mathcal{V}(B^*)$, then $K \in \mathcal{V}(A^*)$.
    2. If $K \in \mathcal{V}(A^*)$ and $L \in \mathcal{V}(B^*)$, then $KcL \in \mathcal{V}(A^*)$.