Ordered monoids and $\mathcal{J}$-trivial monoids

Karsten Henckell* and Jean-Eric Pin†

July 19, 1999

Abstract

In this paper we give a new proof of the following result of Straubing and Thérien: every $\mathcal{J}$-trivial monoid is a quotient of an ordered monoid satisfying the identity $x \leq 1$.

We will assume in this paper that the reader has a basic background in finite semigroup theory (in particular, Green's relations and identities defining varieties) and in computer science (languages, trees, heaps). All semigroups except free monoids and free semigroups are assumed finite. As a consequence, the term *variety* always means *variety of finite semigroups* (or *pseudo-variety*) and the term *identity* refers to *pseudo-identity*, in the terminology of Almeida [2].

A relation $\leq$ on a semigroup S is *stable* if, for every $x, y, z \in S$, $x \leq y$ implies $xz \leq yz$ and $zx \leq zy$. An *ordered semigroup* is a semigroup S equipped with a stable partial order $\leq$ on S. Ordered monoids are defined analogously.

Let A^* be a free monoid. Given a subset P of A^*, the relation $\preceq_P$ defined on A^* by setting $u \preceq_P v$ if and only if, for every $x, y \in A^*$,

$$xvy \in P \Rightarrow xuy \in P,$$

is a stable partial preorder. The equivalence relation $\sim_P$ associated with $\preceq_P$ is defined, for every $x, y \in A^*$, by

$$xuy \in P \iff xvy \in P$$

The monoid $M(P) = A^*/{\sim_P}$, ordered with the order relation induced by $\preceq_P$, is called the *ordered syntactic monoid* of P.

*New College of the University of South Florida, Sarasota, FL 34243, USA, henckell@virtu.sar.usf.edu

†LIAFA, Université Paris VII et CNRS, Tour 55-56, 2 Place Jussieu, 75251 Paris Cedex 05, FRANCE, Jean-Eric.Pin@liafa.jussieu.fr

1. Introduction

The aim of this paper is to give a new proof of the following result of Straubing and Thérien [20]:

Theorem 1.1. *Every $\mathcal{J}$-trivial monoid is a quotient of an ordered monoid satisfying the identity $x \leq 1$.*

There are several reasons to consider Theorem 1.1 as an important result in the theory of finite semigroups. The first reason is its close connection with a celebrated result of Simon in language theory [16]. Recall that a language of A^* is *piecewise testable* if it is a boolean combination of languages of the form $A^*a_1A^*a_2A^* \cdots A^*a_nA^*$. Simon's theorem can be stated as follows.

Theorem 1.2. *A recognizable language is piecewise testable if and only if its syntactic monoid is $\mathcal{J}$-trivial.*

It is not very difficult to establish the equivalence of Theorems 1.1 and 1.2. In one direction, it suffices to observe that the ordered syntactic monoid of a language of the form $A^*a_1A^*a_2A^* \cdots A^*a_nA^*$ satisfies the identity $x \leq 1$. In the opposite direction, it is easy to establish that any language recognized by an ordered monoid satisfying the identity $x \leq 1$ is piecewise testable. There are actually several known proofs of Theorem 1.2, but the proof of Straubing and Thérien was the first to proceed by induction on the size of the monoid.

The second reason of the importance of Theorem 1.1 lies in the role played by ordered monoids in this statement. A systematic use of ordered monoids in language theory was initiated in [11] and developed for instance in [13, 14, 15]. This approach, combined with some deep results obtained in the recent years [3, 4, 5, 6, 7, 8, 9, 10] gives evidence that Theorem 1.1, far from being an isolated result, is actually the prototype of a new kind of covering theorems. We list a few of these results from [13] below, to give the flavor to the reader. Recall that a block group is a monoid in which every regular $\mathcal{R}$-class and $\mathcal{L}$-class contains a unique idempotent. Many equivalent definitions can be found for instance in [12].

Theorem 1.3. *Every block group monoid is the quotient of an ordered monoid satisfying the identity $x^\omega \leq 1$.*

A similar result holds for monoids with commuting idempotents.

Theorem 1.4. *Every monoid with commuting idempotents is the quotient of an ordered monoid with commuting idempotents satisfying the identity* $x^\omega \leq 1$.

Our last example comes from language theory. The languages of level 1 in the so called *dot-depth* hierarchy were characterized by Knast [9, 10]. Their syntactic semigroup satisfies Knast identity:

$$(x^\omega py^\omega qx^\omega)^\omega py^\omega s(x^\omega ry^\omega sx^\omega)^\omega = (x^\omega py^\omega qx^\omega)^\omega(x^\omega ry^\omega sx^\omega)^\omega \qquad \text{(K)}$$

Theorem 1.5. *Every semigroup satisfying (K) is the quotient of an ordered semigroup satisfying the identity* $x^\omega yx^\omega \leq x^\omega$.

All these statements follow the same pattern, stated here in the monoid case:

Every monoid M *of a certain variety of monoids* $\mathbf{V}$ *is the quotient of an ordered monoid* $\hat{M}$ *of a certain variety of ordered monoids* $\mathbf{V}'$.

Unfortunately, no direct proof of Theorems 1.3, 1.4 or 1.5 is known and it is a challenging problem to find such a direct proof. One of the difficulties is that the covering monoid $\hat{M}$ is usually not constructed directly. Even in the proof of Straubing and Thérien [20], the cover is built by induction on the size of M, and thus its construction requires several steps.

In this paper, we give a proof of Theorem 1.1 which provides a direct construction of the covering monoid. Although our construction is rather abstract, we hope it will be easier to adapt to the cases noted above than the indirect proofs.

Factorization forests play an important role in this proof. This gives some evidence that this concept introduced by Simon in 1989 [17, 18, 19] to supersede the Ramsey-type arguments used in semigroup theory, is indeed a fundamental tool. Another illustration of the power of factorization forests can be found in [13].

Technically speaking, our proof is a *global* version of that of Straubing and Thérien. We keep the idea of using 2-factorizations, but since we want a direct construction, the induction used in [20] has to be done in one single step. This leads naturally to factorization trees, which exactly encode iterated factorizations. The level of induction of the Straubing-Thérien proof is now controlled by the label of the nodes in the factorization tree. This is the reason why our factorizations trees form a heap with respect to the $\leq_{\mathcal{J}}$-order.

The resulting proof is certainly much longer than the original proof. The main reason is that dealing with trees requires an important amount of notation. Nevertheless, our construction is not *conceptually* difficult. Given a set of generators A of M, it consists in defining a preorder on A^* in the following way:

(1) Every word admits a factorization tree.
(2) The $\leq_{\mathcal{J}}$-order on M induces a natural preorder $\leq$ on factorization trees.
(3) The preorder on trees extends naturally to a preorder on words: if u and v are words, $u \leq v$ if, for every factorization tree s of u, there is a factorization tree t of v such that $s \leq t$.

Now the equivalence $\sim$ associated with $\leq$ is a congruence of finite index, and the required cover is the monoid $A^*/{\sim}$, equipped with the partial order inherited from $\leq$.

2. Ordered monoids and $\mathcal{J}$-trivial monoids

In the sequel, we fix a $\mathcal{J}$-trivial monoid M. Then there exists a finite alphabet A and a surjective monoid morphism $\pi : A^* \to M$ such that, for each $a \in A$, $\mu(a) \neq 1$. One can take for instance $A = M \setminus \{1\}$ and set $\mu(a) = a$ for each $a \in A$.

The set of idempotents of M is denoted by $E(M)$ and the set of words of A^* with an idempotent image in M is denoted by R. In particular, $R = \pi^{-1}(E(M))$. An element which is not idempotent is called *null*. The set of null elements of M is denoted by $\mathrm{Null}(M)$. Finally, we denote by $B = A \setminus R$ the set of letters with a null image in M. We first recall some elementary facts about $\mathcal{J}$-trivial monoids.

Proposition 2.1. *Let $a, b, c \in M$ and $e \in E(M)$.*

(1) *If $a \leq_{\mathcal{J}} b$ and if $ac \in E(M)$, then $ac \leq_{\mathcal{L}} bc$ (and thus $ac \leq_{\mathcal{J}} bc$).*
(2) *If $e \leq_{\mathcal{J}} a$, then $e = ea = ae$.*
(3) *If $e \leq_{\mathcal{J}} a$ and $e \leq_{\mathcal{J}} b$, then $e \leq_{\mathcal{J}} ab$.*
(4) *If $e \leq_{\mathcal{J}} abc$, then $e \leq_{\mathcal{J}} ac$.*

Proof. (1) If $a \leq_{\mathcal{J}} b$, then $a = xby$ for some $x, y \in M$. Thus $xbyc = ac = (ac)(ac) = (xbyc)(xbyc)$. Since M is $\mathcal{J}$-trivial, it follows that $xbyc$ is fixed under right multiplication by x, b, y and c. Therefore $ac = xbycxbc = (xbycx)bc$. Thus $ac \leq_{\mathcal{L}} bc$.

(2) If $e \leq_{\mathcal{J}} a$, $e = xay$ for some $x, y \in M$. Thus $xay = (xay)(xay)$ and xay is fixed under right multiplication by a, whence $e = ea$. A dual argument would show that $e = ae$.

(3) If $e \leq_{\mathcal{J}} a$ and $e \leq_{\mathcal{J}} b$, then $ea = e = eb$ by (2) and thus $e = eab$, whence $e \leq_{\mathcal{J}} ab$.

(4) If $e \leq_{\mathcal{J}} abc$, then $e \leq_{\mathcal{J}} a$ since $abc \leq_{\mathcal{J}} a$. Similarly, $e \leq_{\mathcal{J}} c$, and by (3), $e \leq_{\mathcal{J}} ac$. □

We now introduce one of the key concept of this article. A *good factorization* of a word $u \in A^*$ is a triple of one of the following types:

(1) (u_0, a, u_1) with $u_0, u_1 \in A^*$, $a \in B$, $u = u_0 a u_1$, $\pi(u) <_{\mathcal{J}} \pi(u_0)$ and $\pi(u) <_{\mathcal{J}} \pi(u_1)$.

(2) $(u_0, 1, u_1)$ with $u_0, u_1 \in A^*$, $u = u_0 u_1$, $\pi(u) <_{\mathcal{J}} \pi(u_0)$ and $\pi(u) <_{\mathcal{J}} \pi(u_1)$.

Proposition 2.2. *Every word of $A^* \setminus R$ has a good factorization.*

Proof. Let $u \in A^* \setminus R$. Since $\pi(u)$ is not idempotent, u is not the empty word, and $\pi(u) <_{\mathcal{J}} \pi(1)$. Therefore, u has a maximal prefix u_0 such that $\pi(u) <_{\mathcal{J}} \pi(u_0)$. Setting $u = u_0 a u_1$, with $a \in A$, we have $\pi(u_0 a) = \pi(u)$ by the maximality of u_0. It follows that $\pi(u_1) \neq \pi(u)$, for otherwise, $\pi(u)$ would be idempotent. Thus $\pi(u) <_{\mathcal{J}} \pi(u_1)$. If $a \in B$, (u_0, a, u_1) is a good factorization of u. Otherwise $\pi(a)$ is idempotent, and we claim that $(u_0, 1, a u_1)$ is a good factorization. We already know that $\pi(u) <_{\mathcal{J}} \pi(u_0)$ and it suffices to verify that $\pi(u) <_{\mathcal{J}} \pi(a u_1)$. Assume by contradiction that $\pi(u) = \pi(a u_1)$. Then, the sequence of equalities

$$\pi(u) = \pi(u_0)\pi(a)\pi(u_1) = \pi(u_0)\pi(a)\pi(a)\pi(u_1) = \pi(u_0 a)\pi(a u_1) = \pi(u)\pi(u)$$

shows that $\pi(u)$ is idempotent, a contradiction. □

3. Binary trees

Binary trees play a crucial role in this article and we need a rigorous definition of these objects. In general, given a set S of symbols and a set V of variables, the set $T(S, V)$ of binary trees over S and V is the smallest set T such that

(1) for each $v \in V$, $(v) \in T$,

(2) if $t_0, t_1 \in T$ and $s \in S$, then $(t_0, s, t_1) \in T$.

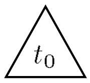
A set of trees is called a *forest*. The *root*, the *leaves* and the *internal nodes* of a tree t are defined by structural induction as follows.

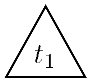
(1) if $t = (v)$, then v is the root and the unique leaf of t. In this case, t does not have any internal node.

(2) if $t = (t_0, s, t_1)$, then s is the root of t. The leaves of t are those of t_0 and of t_1. The internal nodes of t are s and those of t_0 and of t_1.

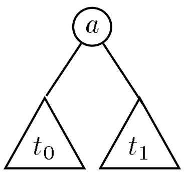
It is often convenient to use a graphical representation for trees, according to the following rules:

(1) the tree (v) is represented by v,

(2) if t_0 and t_1 are represented respectively by 

and 

, then (t_0, a, t_1) is represented by



In this representation, the leaves are variables and the internal nodes are symbols. For instance, the tree $((u, a, v), 1, ((v, a, v), b, (u, 1, w)))$ is represented by

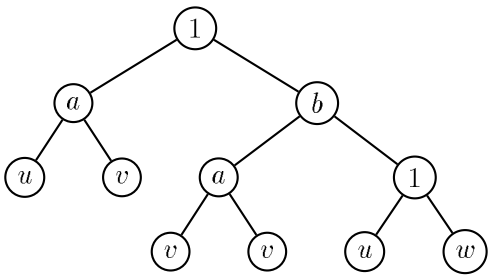


Figure 3.1. A tree

An occurrence of a node within a tree may be given by its *position*, which is a word of $\{0, 1\}^*$ describing the path from the root of the tree to the node, with the convention that 0 means left and 1 means right. For instance, in the tree represented in Figure 3., there are two occurrences of the node u, the nodes in position 00 and 110. Given a position p in a tree t, the subtree of t rooted at p is denoted $t_{|p}$. The tree obtained by replacing this subtree $t_{|p}$ by a tree s is denoted by $t[s]_p$. Here is the formal definition:

(1) If p is the empty word, $t_{|p} = t$ and $t[s]_p = s$.

(2) If $p = 0q$ for some $q \in \{0,1\}^*$, and if $t = (t_0, x, t_1)$, then $t_{|p} = t_{0|q}$ and $t[s]_p = (t_0[s]_q, x, t_1)$.

(3) If $p = 1q$ for some $q \in \{0,1\}^*$, and if $t = (t_0, x, t_1)$, then $t_{|p} = t_{1|q}$ and $t[s]_p = (t_0, x, t_1[s]_q)$.

A position q is said to be *on the right of* p (or that p is *on the left of* q), if q is visited after p in an inorder walk of the tree. Formally, one of the following cases hold:

(1) $p = r0p'$ and $q = r1q'$, for some $r, p', q' \in \{0,1\}^*$,

(2) $q = p1q'$, for some $q' \in \{0,1\}^*$,

(3) $p = q0p'$, for some $p' \in \{0,1\}^*$.

By extension, we say that a position q is *on the right of the subtree rooted at* p if q is on the right of every position in $p\{0,1\}^*$, which amounts to saying that we are in case (1) or in case (3).

4. Factorization trees

Formally, a *factorization tree* is a tree with symbols in $(B \cup \{1\}) \times \mathrm{Null}(M)$ and variables in $R \times E(M)$ satisfying certain conditions. It means that the internal nodes are pairs of the form (a, n), with $a \in B \cup \{1\}$ and $n \in \mathrm{Null}(M)$. The leaves are pairs of the form (u, e), where $u \in R$ and $e \in E(M)$. The first component of a node, which is always an element of A^*, is called the *yield* of the node. The second component, which is always an element of M, is called its *label*. The label of a tree t is the label of its root. It is denoted by $\lambda(t)$.

Factorization trees are more conveniently defined by structural induction as follows:

(1) If $u \in R$ and $e \in E(M)$, then (u, e) is a factorization tree if and only if $e \leq_{\mathcal{J}} \pi(u)$.

(2) If t_0 and t_1 are factorization trees, if $a \in B \cup \{1\}$ and $n \in \mathrm{Null}(M)$ satisfies $n <_{\mathcal{J}} \lambda(t_0)$, $n <_{\mathcal{J}} \lambda(t_1)$ and $n \leq_{\mathcal{J}} \pi(a)$, then $(t_0, (a, n), t_1)$ is a factorization tree.

The second condition states that the labels of a factorization tree form a heap, with respect to the $>_{\mathcal{J}}$ order: the label of each descendant of a given node is $\mathcal{J}$-above the label of that node.

We use a graphical representation in which a node (a, n) is represented by a circled a labelled by n.

Example 4.1. For instance, the tree

$$(((aa, e_0), (1, n_1), (b, e_1)), (a, n_0), (((ba, e_2), (1, n_3), (bb, e_3)), (b, n_2), (ab, e_4)))$$

is represented below

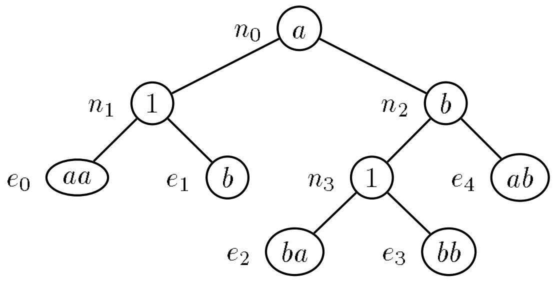


Figure 4.1. A factorization tree

This tree represents the iterated factorization

$$((aa, 1, b), a, ((ba, 1, bb), b, ab))$$

The *yield* of a tree t is the word $\mu(t)$ spelled during an inorder walk of t. It can be formally defined by structural induction as follows:

(1) if $t = (u, e)$ then $\mu(t) = u$,

(2) if $t = (t_0, (a, n), t_1)$, then $\mu(t) = \mu(t_0)a\mu(t_1)$.

For instance, if t is the tree represented in Figure 4.1, an inorder walk of t prints, in this order, the words aa, 1, b, a, ba, 1, bb, b, ab. Therefore, the yield of this tree is $aababababbbab$.

The *value* of a tree t is the value of its yield in M, denoted by $\nu(t)$. In other words, $\nu = \pi \circ \mu$. We remind the reader that the leaves of our trees are elements of $R \times E(M)$. It follows in particular that if t has a single node, then $\nu(t)$ is necessarily an idempotent of M.

If the yield of a tree is u, the tree is called a factorization tree of u. The set $F(u)$ of all the factorization trees of u is called the *factorization forest* of u. The next proposition shows in particular that $F(u)$ is never empty.

Proposition 4.1. *Each word u admits a factorization tree of label $\pi(u)$.*

Proof. If $u \in R$, then $(u, \pi(u))$ is by definition a factorization tree of u. This covers in particular the case $u = 1$. If $u \notin R$, then by Proposition 2.2, u admits a good factorization (u_0, a, u_1), with $a \in B \cup \{1\}$. Arguing by induction on the length of u, we may assume that u_0 and u_1 admit factorization trees t_0 and t_1, of label $\pi(u_0)$ and $\pi(u_1)$, respectively. We claim that $t = (t_0, (a, \pi(u)), t_1)$ is a factorization tree of u. First the yield of t is u. Next, since (u_0, a, u_1) is a good factorization of u, $\pi(u) <_{\mathcal{J}} \pi(u_0) = \lambda(t_0)$, $\pi(u) <_{\mathcal{J}} \pi(u_1) = \lambda(t_1)$ and $\pi(u) \leq_{\mathcal{J}} \pi(a)$, proving the claim. □

Example 4.2. Let $A = \{a, b, c\}$ and let M be the syntactic monoid of the language a^*bc. Thus M is the monoid with zero presented by $(\{a, b, c\} \mid a^2 = a, ab = b, ac = ba = bb = ca = cb = cc = 0)$. Thus $M = \{1, a, b, c, bc, 0\}$. The $\mathcal{J}$-class structure of M is represented in Figure 4.2

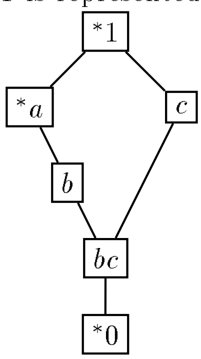
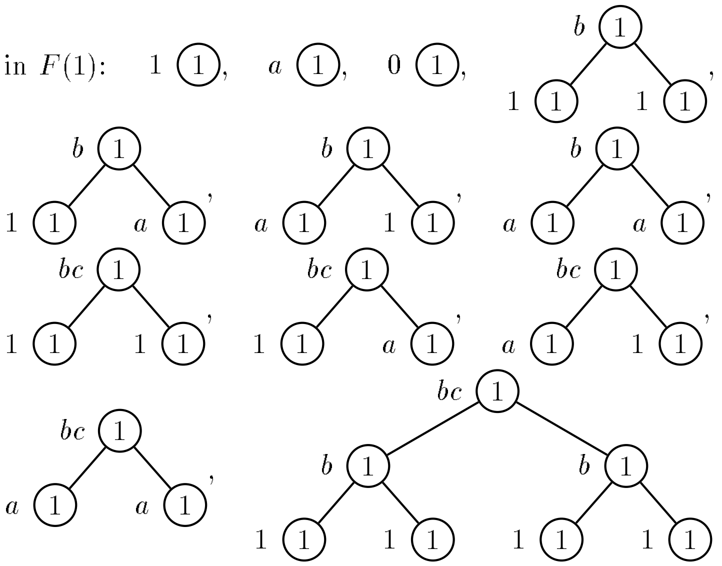


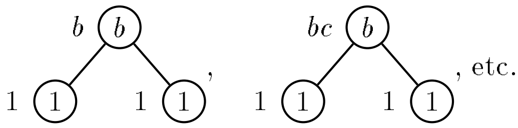
Figure 4.2. The $\mathcal{J}$-class structure of M

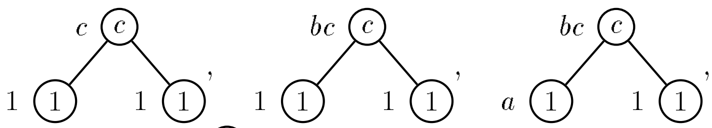
Some factorization trees are listed below.

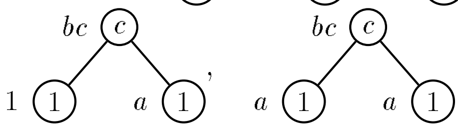
in $F(1)$:

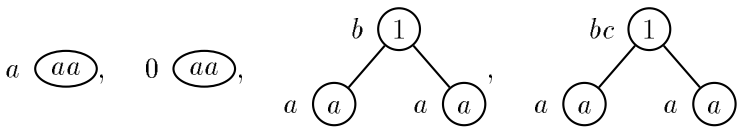


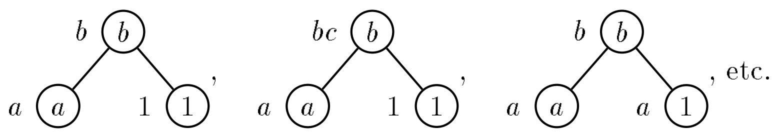
in $F(a)$: a (a), 0 (a), etc.

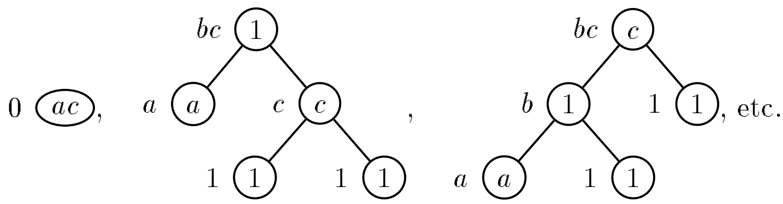
in $F(b)$: 

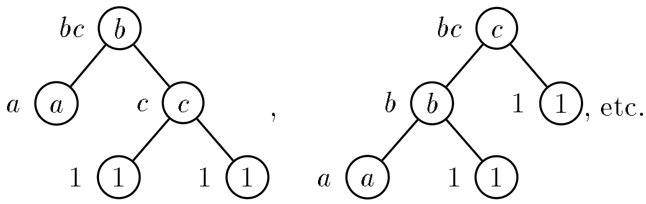
in $F(c)$: 



in $F(aa)$: 

in $F(ab)$: 

in $F(ac)$: 

in $F(abc)$: 

Under certain conditions, replacing a factorization tree in a factorization tree gives a factorization tree.

Proposition 4.2. *Let s and s' be two factorization trees and let p be a position in s such that $\lambda(s_{|p}) \leq_{\mathcal{J}} \lambda(s')$. Then $s[s']_{|p}$ is a factorization tree.*

Proof. The result is quite intuitive, but we give nevertheless a formal proof by structural induction on s. If p is the empty word, then $s[s']_{|p} = s'$ and the result is trivial. Assume that $p = 0q$ for some $q \in \{0,1\}^*$ (the case $p = 1q$ would be similar). Then, if $s = (s_0, (a, n), s_1)$, $s[s']_{|p} = (s_0[s']_{|q}, (a, n), s_1)$. Now, s_1 is a factorization tree by definition and since $\lambda(s_{0|q}) = \lambda(s_{|p}) \leq \lambda(s')$, $s_0[s']_{|q}$ is a factorization tree by the induction hypothesis. Furthermore, since s is a factorization tree, $n \leq_{\mathcal{J}} \pi(a)$, $n <_{\mathcal{J}} \lambda(s_0)$ and $n <_{\mathcal{J}} \lambda(s_1)$. Therefore, it suffices to verify that

$$n <_{\mathcal{J}} \lambda(s_0[s']_{|q}) \tag{1}$$

If q is the empty word, $s_0[s']_{|q} = s'$ and $s_{|p} = s_0$. Therefore,

$$n <_{\mathcal{J}} \lambda(s_0) = \lambda(s_{|p}) \leq_{\mathcal{J}} \lambda(s') = \lambda(s_0[s']_{|q})$$

which proves (1). If q is not empty, then $\lambda(s_0[s']_{|q}) = \lambda(s_0)$, which again gives (1). □

5. A preorder on factorization trees

We define a relation $\leq$ on factorization trees by structural induction:

(1) If s has a single node, then $s \leq t$ if and only if $\nu(s) \leq_{\mathcal{J}} \nu(t)$ and $\lambda(s) \leq_{\mathcal{J}} \lambda(t)$,

(2) If $s = (s_0, (a, n), s_1)$, then $s \leq t$ if and only if $t = (t_0, (b, n), t_1)$ with $s_0 \leq t_0$, $s_1 \leq t_1$ and $b \in \{1, a\}$.

Some of the properties of this relation are summarized in the next proposition.

Proposition 5.1.

(1) *If $s \leq t$, then $\lambda(s) \leq_{\mathcal{J}} \lambda(t)$.*

(2) *If $e \leq_{\mathcal{J}} \nu(s)$ for some idempotent e and if $s \leq t$, then $e \leq_{\mathcal{J}} \nu(t)$.*

(3) *The relation $\leq$ is a preorder on factorization trees.*

Proof. (1) If s has a single node, then $\lambda(s) \leq_{\mathcal{J}} \lambda(t)$ by definition. If $s = (s_0, (a, n), s_1)$, then $\lambda(s) = \lambda(t) = n$.

(2) By structural induction on s. If s has a single node, then $\nu(s) \leq_{\mathcal{J}} \nu(t)$ and thus $e \leq_{\mathcal{J}} \nu(s) \leq_{\mathcal{J}} \nu(t)$. Otherwise $s = (s_0, (a, n), s_1)$ and since $e \leq_{\mathcal{J}} \nu(s) = \nu(s_0)\pi(a)\nu(s_1)$, $e \leq_{\mathcal{J}} \nu(s_0)$, $e \leq_{\mathcal{J}} \pi(a)$ and $e \leq_{\mathcal{J}} \nu(s_1)$. Now, since $s \leq t$, $t = (t_0, (b, n), t_1)$ with $b \in \{1, a\}$, $s_0 \leq t_0$ and $s_1 \leq t_1$. Then $e \leq_{\mathcal{J}} \pi(a) \leq_{\mathcal{J}} \pi(b)$ since $b \in \{1, a\}$. Applying the induction hypothesis to s_0 and s_1, we get $e \leq_{\mathcal{J}} \nu(t_0)$ and $e \leq_{\mathcal{J}} \nu(t_1)$ and by Proposition 2.1 (4), $e \leq_{\mathcal{J}} \nu(t_0)\pi(b)\nu(t_1) = \nu(t)$.

(3) The relation is clearly reflexive. Suppose that $r \leq s$ and $s \leq t$. We prove that $r \leq t$ by structural induction on r. First, $\lambda(r) \leq_{\mathcal{J}} \lambda(s) \leq_{\mathcal{J}} \lambda(t)$ by (1). If r has a single node, then $\nu(r) \leq_{\mathcal{J}} \nu(s)$. Since $\nu(r)$ is idempotent, we have by (2), $\nu(r) \leq_{\mathcal{J}} \nu(t)$, whence $r \leq t$. If $r = (r_0, (a, n), r_1)$, then $s = (s_0, (b, n), s_1)$, with $r_0 \leq s_0$, $r_1 \leq s_1$ and $b \in \{1, a\}$ and $t = (t_0, (c, n), t_1)$, with $s_0 \leq t_0$, $s_1 \leq t_1$ and $c \in \{1, b\}$. Then $c \in \{1, a\}$ since $b \in \{1, a\}$, and by induction $r_0 \leq t_0$ and $r_1 \leq t_1$. Therefore $r \leq t$. □

Corollary 5.2. *Let s and t be trees such that $s \leq t$.*

(1) *If $\nu(s)$ is idempotent, then $\nu(s) \leq_{\mathcal{J}} \nu(t)$.*

(2) *If the yield of the root of s is 1, then the yield of the root of t is also 1.*

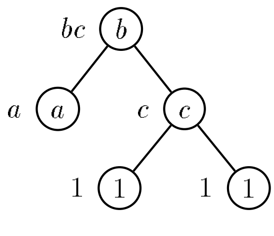
Proof. (1) is a direct consequence of Proposition 5.1 (2).

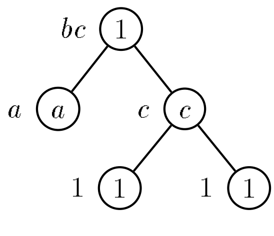
(2) By structural induction on s. Suppose first that s has a single node. Then the yield of s is the empty word, whence $\nu(s) = 1$, and by (1), $1 \leq_{\mathcal{J}} \nu(t)$. Since M is $\mathcal{J}$-trivial, it follows that $\nu(t) = 1$, and thus the yield of t is the empty word, since we assume that $\pi(a) \neq 1$ for every $a \in A$.

If s has more than one node, then necessarily $s = (s_0, (1, n), s_1)$, and since $s \leq t$, $t = (t_0, (1, n), t_1)$ with $s_0 \leq t_0$ and $s_1 \leq t_1$. Therefore, the yield of the root of t is 1. □

However, Property 5.2 (1) fails if $\nu(s)$ is not idempotent, as shown in the next example.

Example 5.1. We take the same morphism π as in Example 4.2. Let

$s =$ 

and $t =$ 

Then $s \leq t$ but $\nu(s) = bc \not\leq_{\mathcal{J}} \nu(t) = 0$.

We come back to a more positive result with the next theorem. Denote by $\sim$ the equivalence associated with $\leq$, defined by $s \sim t$ if and only if $s \leq t$ and $t \leq s$.

Theorem 5.3. *If $s \sim t$, then $\nu(s) = \nu(t)$.*

Proof. By structural induction on s. First assume that s has a single node. Then, since $t \leq s$, t also has a single node. Furthermore $\nu(s) \leq_{\mathcal{J}} \nu(t)$ and $\nu(t) \leq_{\mathcal{J}} \nu(s)$, whence $\nu(s) = \nu(t)$. Suppose now that $s = (s_0, (a, n), s_1)$ for some $a \in B \cup \{1\}$. As $s \leq t$, $t = (t_0, (b, n), t_1)$, with $s_0 \leq t_0$, $s_1 \leq t_1$, $b \in \{a, 1\}$, and as $t \leq s$, $t_0 \leq s_0$, $t_1 \leq s_1$ and $a \in \{b, 1\}$. It follows that $a = b$, $s_0 \sim t_0$ and $s_1 \sim t_1$, and, by induction, $\nu(s_0) = \nu(t_0)$ and $\nu(s_1) = \nu(t_1)$. Therefore $\nu(s) = \nu(s_0)\pi(a)\nu(s_1) = \nu(t_0)\pi(b)\nu(t_1) = \nu(t)$. □

The next proposition shows that the preorder is inherited by subtrees.

Proposition 5.4. *Let s and t be trees such that $s \leq t$.*

(1) *If p is a position in s, then p is also a position in t and $s_{|p} \leq t_{|p}$. In particular, $\lambda(s_{|p}) \leq_{\mathcal{J}} \lambda(t_{|p})$.*
(2) *If p is an internal position in s, then p is also an internal position in t. Furthermore, if the node in position p in s is (a, n), then the node in position p in t is equal to (b, n), for some $b \in \{a, 1\}$.*

Proof. The result is clear if p is the empty word, since then, $s_{|p} = s$ and $t_{|p} = t$. This covers the case where s has a single node. Arguing by induction on the structural complexity of s, assume that $s = (s_0, (a, n), s_1)$. Then since $s \leq t$, $t = (t_0, (b, n), t_1)$ with $s_0 \leq t_0$, $s_1 \leq t_1$ and $b \in \{a, 1\}$. If $p = 0q$ for some $q \in \{0, 1\}^*$, then $s_{|p} = s_{0|q}$, $t_{|p} = t_{0|q}$, and by induction $s_{0|q} \leq t_{0|q}$. Therefore $s_{|p} \leq t_{|p}$.

Furthermore, if p is an internal node of s, then q is an internal node of s_0. By induction, q is an internal node of t_0 and hence p is an internal node of t. Finally, if the node in position p in s is (a, n), the node in position q in s_0 is also (a, n), and by induction, the node in position q in t_0 is equal to (b, n), for some $b \in \{a, 1\}$. But this node is also the node in position q of t.

The case $p = 1q$ is handled similarly. □

We conclude this section by a result which states that, under certain conditions, the preorder $\leq$ is stable under replacement.

Proposition 5.5. *Let s and t be trees such that $s \leq t$ and let p be a position in s. Then if s' and t' are factorization trees such that $s' \leq t'$, $\lambda(s_{|p}) \leq_{\mathcal{J}} \lambda(s')$ and $\lambda(t_{|p}) \leq_{\mathcal{J}} \lambda(t')$, then $s[s']_{|p}$ and $t[t']_{|p}$ are factorization trees and $s[s']_{|p} \leq t[t']_{|p}$.*

Proof. The fact that $s[s']_{|p}$ and $t[t']_{|p}$ are factorization trees is a consequence of Proposition 4.2.

If p is the empty word, the relation $s[s']_{|p} \leq t[t']_{|p}$ reduces to $s' \leq t'$. This covers the case where s has a single node. Arguing by induction on the structural complexity of s, assume that $s = (s_0, (a, n), s_1)$. Then since $s \leq t$, $t = (t_0, (b, n), t_1)$ with $s_0 \leq t_0$, $s_1 \leq t_1$ and $b \in \{a, 1\}$. Suppose that $p = 0q$ for some $q \in \{0, 1\}^*$ (the case $p = 1q$ would be similar). Then

$$s[s']_{|p} = (s_0[s']_{|q}, (a, n), s_1), \quad t[t']_{|p} = (t_0[t']_{|q}, (b, n), t_1)$$

and by induction, $s_0[s']_{|q} \leq t_0[t']_{|q}$. It follows that $s[s']_{|p} \leq t[t']_{|p}$. □

It is convenient to state separately a consequence of Proposition 5.5 which will be used in the proof of Theorem 6.4. Let (a, n) be an internal node in position p of a tree s. For each $b \in B \cup \{1\}$ such that $n \leq_{\mathcal{J}} \pi(b)$, we shall denote by $s[a \to b]_{|p}$ the tree obtained by replacing this node by (b, n).

Corollary 5.6. *Let s and t be trees such that $s \leq t$ and assume that s and t have the same internal node (a, n) in position p. Then, for each $b \in B \cup \{1\}$, $s[a \to b]_{|p} \leq t[a \to b]_{|p}$.*

Proof. By Proposition 5.4 (1), $s_{|p} \leq t_{|p}$. Since p is an internal position, $s_{|p} = (s_0, (a, n), s_1)$ and thus $t_{|p} = (t_0, (a, n), t_1)$ with $s_0 \leq t_0$ and $s_1 \leq t_1$. Let $s' = (s_0, (b, n), s_1)$ and $t' = (t_0, (b, n), t_1)$. By construction, $s' \leq t'$. Furthermore, $\lambda(s_{|p}) = n\lambda(s')$ and, similarly, $\lambda(t_{|p}) = \lambda(t')$. The corollary now follows from Proposition 5.5. □

6. A preorder on words

We define a preorder on A^* by setting $u \leq v$ if and only if, for each tree $s \in F(u)$, there exists a tree $t \in F(v)$ such that $s \leq t$.

Proposition 6.1.

(1) *If $e \leq_{\mathcal{J}} \pi(u)$ for some idempotent e and if $u \leq v$, then $e \leq_{\mathcal{J}} \pi(v)$.*

(2) *The relation $\leq$ is a preorder on A^*.*

(3) *The relation $u \leq 1$ holds for every $u \in A^*$.*

Proof. The first two properties follow directly from Proposition 5.1. For the last property, it suffices to verify that, for each tree s, there exists a tree $t \in F(1)$ such that $s \leq t$ and $\lambda(s) = \lambda(t)$. If s has a single node, say $s = (u, e)$, one can take $t = (1, e)$. If $s = (s_0, (a, n), s_1)$, we may assume by induction that there exist two trees $t_0, t_1 \in F(1)$ such that $s_0 \leq t_0$, $\lambda(s_0) = \lambda(t_0)$, $s_1 \leq t_1$ and $\lambda(s_1) = \lambda(t_1)$. Then $t = (t_0, (1, n), t_1)$ is a factorization tree, since $n <_{\mathcal{J}} \lambda(s_0) = \lambda(t_0)$, $n <_{\mathcal{J}} \lambda(s_1) = \lambda(t_1)$ and $n \leq_{\mathcal{J}} \pi(1)$. Furthermore $\nu(t) = \nu(t_0)\pi(1)\nu(t_1) = 1$ and thus $t \in F(1)$. Finally $s \leq t$ and $\lambda(s) = \lambda(t)$ by construction, and this completes the induction step. $\square$

Theorem 5.3 also has its counterpart on words. As for trees, we denote by $\sim$ the equivalence associated with the preorder $\leq$, defined on A^*.

The next theorem requires an auxiliary definition. Let t be a factorization tree. If we replace each leaf (u, e) of t by $(\pi(u), e)$, we obtain a new tree with variables in $E(M) \times E(M)$, called "t modulo π".

Theorem 6.2. *The relation $\sim$ has finite index.*

Proof. We first observe that two factorization trees with the same internal nodes and the same leaves modulo π are equivalent. It follows that if two words have the same factorization forests modulo π, they are equivalent. Now each factorization tree modulo π is a binary tree satisfying the following conditions:

(1) its depth is bounded by the $\mathcal{J}$-depth of M,
(2) its internal nodes are in $(B \cup \{1\}) \times \mathrm{Null}(M)$,
(3) its leaves are in $E(M) \times E(M)$

Since there are finitely many such trees, $\sim$ has finite index. $\square$

Theorem 6.3. *If $u \sim v$, then $\pi(u) = \pi(v)$.*

Proof. Let s be a maximal element of $F(u)$. Since $u \leq v$, there exists $t \in F(v)$ such that $s \leq t$, and since $v \leq u$, there exists $s' \in F(u)$ such that $t \leq s'$. Thus $s \leq t \leq s'$ and since s is maximal, it follows $s \sim s'$, whence $s \sim t \sim s'$. Therefore $\pi(u) = \nu(s) = \nu(t) = \pi(v)$ by Theorem 5.3. $\square$

We are now ready to prove the main property of the preorder.

Theorem 6.4. *The preorder $\leq$ is stable on A^*.*

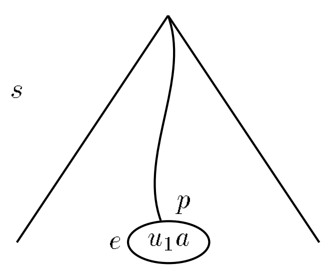
Proof. Let $u, v \in A^*$ with $u \leq v$, and let $a \in A$. By symmetry, it suffices to show that $ua \leq va$. Let $s \in F(ua)$. Let p be the right most position such that the yield of the node N in position p in s is not the empty word. In particular, the yield of every node on the right of N is 1.

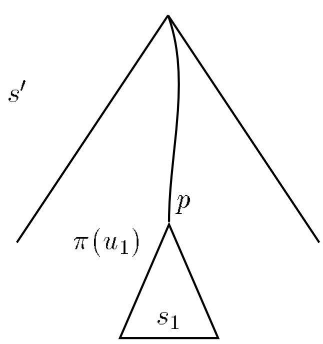
If N is an internal node, it is of the form (a, n) for some $n \in \mathrm{Null}(M)$, and necessarily, $a \in B$. Then $s' = s[a \to 1]_{|p}$ is a factorization tree for u, and since $u \leq v$, there exists a tree $t' \in F(v)$ such that $s' \leq t'$. By construction, the yield of the root of $s'_{|p}$ is 1 and by Corollary 5.2 (2), the yield of the root of $t'_{|p}$ is also 1. Setting $t = t'[1 \to a]_{|p}$ and observing that $s = s'[1 \to a]_{|p}$, we have $s \leq t$ by Corollary 5.6. Furthermore, if q is a position at the right of p in s, and hence in s', the yield of the corresponding node is 1. By Proposition 5.4, $s'_{|q} \leq t'_{|q}$ and by Corollary 5.2 (2), the yield of the node in position q in t', and hence in t, is also 1. It follows that the yield of t is va, and hence $t \in F(va)$.

Assume now that N is a leaf. By the choice of p, the yield of N is a suffix of ua and thus $N = (u_1a, e)$ for some suffix u_1 of u. By Proposition 4.1, u_1 has a factorization tree s_1 of label $\pi(u_1)$. Since (u_1a, e) is a leaf, $\pi(u_1a)$ and e are idempotent and

$$e \leq_{\mathcal{J}} \pi(u_1a) \leq_{\mathcal{J}} \pi(u_1) = \lambda(s_1) = \nu(s_1)$$

It follows that $\lambda(s_{|p}) = e \leq_{\mathcal{J}} \lambda(s_1)$ and, by Proposition 4.2, the tree $s' = s[s_1]_{|p}$ is a factorization tree of u. Note that, if q is a position of s to the right of p, or equivalently, if q is a position of s' to the right of s_1, then $s'_{|q} = s_{|q}$. It follows that the yield of the node in position q in s' is 1.





Now since $u \leq v$, there exists a tree $t' \in F(v)$ such that $s' \leq t'$. Let $t_1 = t'_{|p}$. Observing that $s_1 = s'_{|p}$, we have $s_1 \leq t_1$ by Proposition 5.4. Let q be a position of s' to the right of s_1. We have seen that the yield of the corresponding node is 1. By Proposition 5.4, $s'_{|q} \leq t'_{|q}$ and by Corollary 5.2 (2), the yield of the node in position q in t' is also 1. It follows that the yield v_1 of t_1 is a suffix of v.

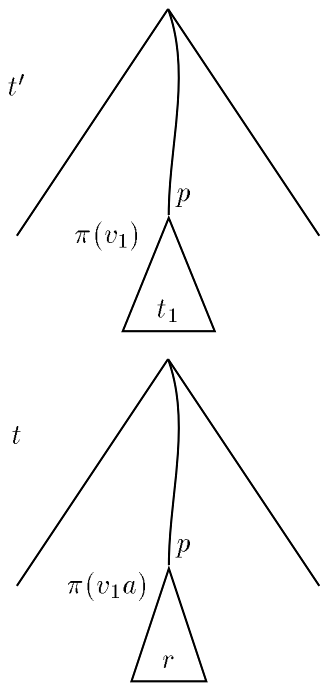
Since $\pi(u_1 a) \leq_{\mathcal{J}} \nu(s_1)$ and $s_1 \leq t_1$, Proposition 5.1 shows that

$$e \leq_{\mathcal{J}} \pi(u_1 a) \leq_{\mathcal{J}} \nu(t_1) = \pi(v_1)$$

Furthermore $e \leq_{\mathcal{J}} \pi(u_1 a) \leq_{\mathcal{J}} \pi(a)$ and by Proposition 2.1

$$e \leq_{\mathcal{J}} \pi(u_1 a) \leq_{\mathcal{J}} \pi(v_1)\pi(a) = \pi(v_1 a) \tag{2}$$

By Proposition 4.1, $v_1 a$ has a factorization tree r of label $\pi(v_1 a)$. Now $\lambda(s_{|p}) = e \leq_{\mathcal{J}} \pi(v_1 a) = \lambda(r)$ and by Proposition 4.2, the tree $t = s[r]_{|p}$ is a factorization tree of va.



We claim that $(u_1 a, e) \leq r$. Indeed, by (2)

$$\nu(u_1 a, e) = \pi(u_1 a) \leq_{\mathcal{J}} \pi(v_1 a) = \nu(r)$$

and

$$\lambda(u_1a, e) = e \leq_{\mathcal{J}} \pi(v_1a) = \lambda(r)$$

proving the claim. Thus by Proposition 5.5, $s = s[(u_1a, e)]_{|p} \leq s[r]_{|p} = t$. $\square$

7. Conclusion

We are now ready to prove Theorem 1.1.

Proof. Theorem 6.2 shows that the monoid $N = A^*/{\sim}$ is finite. Furthermore, the preorder $\leq$ defined on A^* induces a partial order, also denoted by $\leq$, on N. This order can be formally defined by $x \leq y$ if $u \leq v$ for some $u \in \pi^{-1}(x)$ and $v \in \pi^{-1}(y)$. Now, by Proposition 6.1 and Theorem 6.4, $(N, \leq)$ is an ordered monoid satisfying the identity $x \leq 1$. Furthermore, Theorem 6.3 shows that M is a quotient of N. $\square$

References

[1] J. Almeida, Implicit operations on finite $\mathcal{J}$-trivial semigroups and a conjecture of I. Simon, *J. Pure Appl. Algebra* **69**, (1990), 205–218.

[2] J. Almeida, *Finite semigroups and universal algebra*, Series in Algebra Vol 3, World Scientific, Singapore, 1994.

[3] C. J. Ash, Finite semigroups with commuting idempotents, in *J. Austral. Math. Soc. Ser. A* **43**, (1987), 81–90.

[4] C. J. Ash, Inevitable sequences and a proof of the type II conjecture, in *Proceedings of the Monash Conference on Semigroup Theory*, World Scientific, Singapore, (1991), 31–42.

[5] C. J. Ash, Inevitable Graphs: A proof of the type II conjecture and some related decision procedures, *Int. Jour. Alg. and Comp.* **1**, (1991), 127–146.

[6] K. Henckell, Blockgroups = Power Groups, in *Proceedings of the Monash Conference on Semigroup Theory*, World Scientific, Singapore, (1991) 117–134.

[7] K. Henckell, S. W. Margolis, J.-E. Pin and J. Rhodes, Ash's Type II Theorem, Profinite Topology and Malcev Products, *Int. Jour. Alg. and Comp.* **1**, (1991), 411–436.

[8] K. Henckell and J. Rhodes, The theorem of Knast, the $PG = BG$ and Type II Conjectures, in *Monoids and Semigroups with Applications, J. Rhodes (ed.)*, Word Scientific, (1991), 453–463.

[9] R. Knast, A semigroup characterization of dot-depth one languages, *RAIRO Inform. Théor.* **17**, (1983), 321–330.

[10] R. Knast, Some theorems on graph congruences, *RAIRO Inform. Théor.* **17**, (1983), 331–342.

[11] J.-E. Pin, A variety theorem without complementation, *Izvestiya VUZ Matematika* **39** (1995) 80–90. English version, *Russian Mathem. (Iz. VUZ)* **39** (1995) 74–83.

[12] J.-E. Pin, $\mathbf{BG} = \mathbf{PG}$, a success story, in NATO Advanced Study Institute *Semigroups, Formal Languages and Groups, J. Fountain (ed.)*, Kluwer academic publishers, (1995), 33–47.

[13] J.-E. Pin and P. Weil, Polynomial closure and unambiguous product, *Theory Comput. Systems* **30**, (1997), 1–39.

[14] J.-E. Pin and P. Weil, Semidirect product of ordered monoids, in preparation.

[15] J.-E. Pin and P. Weil, The wreath product principle for ordered semigroups, in preparation.

[16] I. Simon, Piecewise testable events, *Proc. 2nd GI Conf., Lect. Notes in Comp. Sci.* **33**, Springer Verlag, Berlin, Heidelberg, New York, (1975), 214–222.

[17] I. Simon, Properties of factorization forests, in *Formal properties of finite automata and applications, J.-E. Pin (ed.), Lect. Notes in Comp. Sci.* **386**, Springer Verlag, Berlin, Heidelberg, New York, (1989), 34–55.

[18] I. Simon, Factorization forests of finite height, *Theor. Comp. Sc.* **72**, (1990), 65–94.

[19] I. Simon, A short proof of the factorization forest theorem, in *Tree Automata and Languages, M. Nivat and A. Podelski eds.*, Elsevier Science Publ., Amsterdam, (1992) 433–438.

[20] H. Straubing and D. Thérien, Partially ordered finite monoids and a theorem of I. Simon, *J. of Algebra* **119**, (1985), 393–399.