Computing Finite Commutative Semigroups

Pierre Antoine Grillet
Communicated by Boris Schein

Abstract
Precedence results are used to improve existing algorithms for the enumeration of finite commutative semigroups. As an application 11,545,843 distinct commutative semigroups of order 9 were found.

0. Introduction

Backtrack algorithms which list all distinct (not isomorphic or antiisomorphic) semigroups of fixed finite order \( n \) were developed in the 1950s [1], [5] and have determined the number of distinct semigroups of order \( n \) for \( n \leq 8 \). In the form perfected in [6] they are still in use today [10]. The general strategy is to list semigroup tables in lexicographic order. A new table is added to the list if it passes an associativity test (to insure that it is a semigroup) and a precedence test (to insure that it is not isomorphic or antiisomorphic to any of the previously computed semigroups). The simplest associativity test requires checking \( n^3 \) triples (and thus is of polynomial complexity), whereas the precedence test requires applying \( n! \) permutations to \( n^2 \) pairs (and is nonpolynomial).

Past improvements in these algorithms have sharpened the associativity test (see for instance [6], [10]); this saves computation time by limiting the number of choices that must be considered. In this article we improve the precedence test. Our strategy is to apply complete precedence tests only to full tables. Incomplete tables are tested with transpositions only, a much faster (and polynomial) test which catches most bad choices. A full table which survives this simpler test has a number of properties which can be used to omit most permutations from its precedence test. This makes the precedence test much faster and greatly improves computation time.

The relevant results are given in section 1 in the case of commutative semigroups. No doubt a similar set of results applies to arbitrary semigroups; however, enumerating all estimated 52 trillion semigroups of order 9 [7] would still require too much time, even on current supercomputers, whereas our improved algorithm has allowed the enumeration of all commutative semigroups of order 9 on a mere personal computer. The actual algorithm is described in section 2. On our comparatively slow machine (a standard 33MHz 486), the unimproved precedence test at order 9 takes several seconds. Our improvements cut total computation time down from over a year to about 130 hours.

Direct observation showed that the improved algorithm still spends most of its time administering precedence tests to satisfactory complete tables. This suggests that future improvements should concentrate on this part of the algorithm.

It is traditional to classify semigroups by number of idempotents and eggbox picture. For the 11,545,843 commutative semigroups of order 9 we used a more detailed classification, which is explained in section 3. Interestingly, 9 is the smallest order at which more than half of all commutative semigroups are nilsemigroups.
1. Precedence Results

1. When $S$ is a finite totally ordered set, binary operations on $S$ can be totally ordered lexicographically: when $m', m'' : S \times S \to S$, then $m' < m''$ in case there exists $a, b \in S$ such that $m'(x, y) = m''(x, y)$ whenever $x < a$, $m'(a, y) = m''(a, y)$ for all $y < b$, and $m'(a, b) < m''(a, b)$.

Now assume that $m : S \times S \to S$ is a commutative and associative operation on $S$. It is not assumed that $m$ and $<$ are compatible. For any permutation $\sigma$ of $S$ define $m^\sigma : S \times S \to S$ by

$$m^\sigma(x, y) = \sigma^{-1} m(\sigma x, \sigma y)$$

for all $x, y \in S$. We say that $S$ (totally ordered by $<$) has precedence in case $m^\sigma \geq m$ for all permutations $\sigma$ of $S$ [2].

Precedence occurs naturally in computer lists, where the elements of $S$ are in a fixed order $<$ and multiplication tables are generated in lexicographic order. A semigroup $(S, m)$ added to the list should not be isomorphic to any of the previous semigroups. This means no permutation $\sigma$ with $m^\sigma < m$; equivalently, $S$ has precedence.

The transposition test in our algorithm verifies $m^\tau \geq m$ only for all transpositions $\tau$ of $S$; then we say that $S$ has weak precedence.

In general there are two kinds of permutations on $S$: $\sigma$ affirms precedence if $m^\sigma \geq m$, and denies precedence to $S$ if $m^\sigma < m$. In this section we find permutations which must affirm precedence, and may therefore be omitted from precedence tests.

2. Assume that $S = (S, m)$ has weak precedence and let $e$ denote the least element of $S$. The following result is sharper than Lemma 1 of [2] but is proved similarly.

**Lemma 1.** When $S$ has weak precedence, $e$ is idempotent; $e x \leq x$ for all $x \in S$; and $x \leq y$ implies $ex \leq ey$.

**Proof.** If $e^2 \neq e$, then $S$ contains an idempotent $f \neq e$ and the transposition $\tau = (e f)$ satisfies

$$\tau^{-1} m(\tau e, \tau e) = \tau(ff) = \tau f = e < m(e, e).$$

Since $e$ is the least element this shows $m^\tau < m$, which contradicts weak precedence.

Next, assume that $ex \leq x$ does not hold for all $x \in S$. Then there is a least element $c$ such that $ce > c$. In particular $c > e$. Let $\tau = (c ec)$. If $c < e$, then $ex \leq x < c$ and $\tau^{-1} m(\tau e, \tau x) = \tau(e x) = \tau(e) = c < m(e, c)$. But

$$\tau^{-1} m(\tau e, \tau c) = \tau(\tau ec) = \tau(ec) = c < m(e, c).$$

This is the required contradiction.

Finally assume that $x \leq y$ does not imply $ex \leq ey$. Then there is a least $c$ such that $c < d$ and $ec > cd$ for some $d$. Again $c > e$. Let $\tau = (c d)$. If $c < e$, then $ex \leq x < c$ and $\tau^{-1} m(\tau e, \tau x) = \tau(ex) = m(e, x)$. On the other hand

$$\tau^{-1} m(\tau e, \tau c) = \tau(ed) = ed < m(e, c)$$

since $ed < ec \leq c$. Hence $m^\tau < m$. $\blacksquare$

If $S$ has precedence it follows from the main theorem of [2] that $e$ is in the kernel of $S$, and must be the least idempotent of $S$ in the Rees order. This provides a quick precedence test:

---

141