On Enumerating Subsemigroups of the Full Transformation Semigroup

Attila Egri-Nagy

joint work with James East (Univ. of Western Sydney) and James D. Mitchell (University of St. Andrews, Scotland)

2013.06.08. Novi Sad Algebra Conference

Motivation

Practical need: to have a library of small transformation semigroups.

Personally, I am looking for interesting holonomy decompositions.

Semigroup enumeration and classification

Problems:

- There are lots of semigroups.
- Most of them are 3-nilpotent, i.e. they satisfy the $xyz = 0$ identity.

> *"So, whereas groups are gems, all of them precious, the garden of semigroups is filled with weeds. One needs to yank out these weeds to find the interesting semigroups."*

Rhodes, J., Steinberg, B.: The q-theory of Finite Semigroups. Springer (2008)

So, it is useless and hopeless.

History of semigroup enumeration

1955 Forsythe, G. E., **SWAC computes 126 distinct semigroups of order 4**, Proc. Amer. Math. Soc., 6 (1955), 443–447.

Tetsuya, K., Hashimoto, T., Akazawa, T., Shibata, R., Inui, T. and Tamura, T., **All semigroups of order at most 5**, *J. Gakugei Tokushima Univ. Nat. Sci. Math.*, 6 (1955), 19–39.

1967 Plemmons, R. J., **There are 15973 semigroups of order 6**, *Math. Algorithms*, 2 (1967), 2–17.

1977 Jürgensen, H. and Wick, P., **Die Halbgruppen der Ordnungen $\leq$ 7**, *Semigroup Forum*, 14 (1) (1977), 69–79.

1994 Satoh, S., Yama, K. and Tokizawa, M., **Semigroups of order 8**, *Semigroup Forum*, 49 (1) (1994), 7–29.

Current state of semigroup enumeration

Inspired by the SmallGroups Library for GAP and Magma there is now a GAP package called Smallsemi.

Smallsemi provides a database of all the small semigroups up to order 8, tools for identifying semigroups and their properties (e.g. commutative, band, inverse, regular, etc., 16 of them in total).

The size of the compressed database is 22 Mbytes.

Andreas Distler, James D. Mitchell

`http://www-groups.mcs.st-andrews.ac.uk/~jamesm/smallsemi/`

Number of semigroups of order n

order	#groups	#semigroups	#3-nilpotent semigroups
1	1	1	0
2	1	4	0
3	1	18	1
4	2	126	8
5	1	1,160	84
6	2	15,973	2,660
7	1	836,021	609,797
8	5	1,843,120,128	1,831,687,022
9	2	52,989,400,714,478	52,966,239,062,973

The calculation was done by combining GAP and a Constraint Satisfaction Problem (CSP) solver Minion `minion.sf.net`.

Enumerating transformation semigroups

Idea: Find the subsemigroups of the full transformation semigroup.

Straightforward brute-force algorithm: enumerate all subsets of $\mathcal{T}_n$ and keep those that form a subsemigroup.

However, there are 2^{n^n} subsets of $\mathcal{T}_n$.

n	n^n	2^{n^n}
1	1	2
2	4	16
3	27	134217728
4	256	115792089237316195423570985008687907853269984665640564039457584007913129639936
5	3125	2^{3125}

We know lot more about permutation groups

Subgroups of $\mathcal{S}_n$

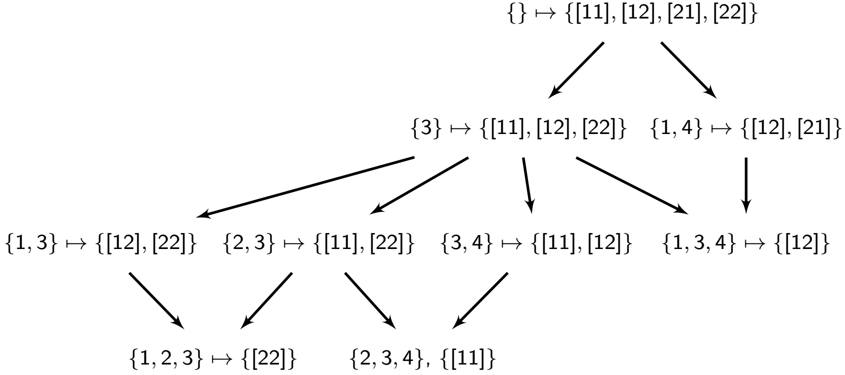
n	#distinct subgroups	#conjugacy classes
1	1	1
2	2	2
3	6	4
4	30	11
5	156	19
6	1455	56
7	11300	96
8	151221	296
9	1694723	554
10	29594446	1593
11	404126228	3094
12	10594925360	10723
13	175238308453	20832

A000638 and A005432 on `oeis.org`.

All subsemigroups of $\mathcal{T}_2$

$1 \mapsto [1,1]$, $2 \mapsto [1,2]$, $3 \mapsto [2,1]$, $4 \mapsto [2,2]$

1144	1144	1144	1144	1144	1144	1144	1144	1144
1234	1234	1234	1234	1234	1234	1234	1234	1234
1324,	1324,	1324,	1324,	1324,	1324,	1324,	1324,	1324
1414	1414	1414	1414	1414	1414	1414	1414	1414



Idea: systematic reduction of multiplication tables

Let S be a semigroup, $n = |S|$. We fix an order on the semigroup elements, $s_1, \ldots, s_n$, thus we can easily refer to the elements by their indices.

Definition

Then the *multiplication table* of S is a $n \times n$ matrix M with entries from $\{1, .., n\}$ such that $M_{i,j} = k$ if $s_i s_j = s_k$. This table is often called the *Cayley-table* of the semigroup.

Definition (cut, closed cut)

A *cut* is a subset of the semigroup, $K \subseteq S$ a set elements that we cut from the M. A cut is *closed* if the table spanned by $S \setminus K$ is a multiplication table, i.e. it is closed under multiplication.

Forbidden Elements

Definition (Forbidden Elements)

$$F(K) = \{i \in S \setminus K \mid \; \exists j \in S \setminus K \text{ such that } M_{i,j} \in K \text{ or } M_{j,i} \in K\}$$

i.e. those elements not in the cut, whose column or row contains an element in the cut.

Example: $\mathcal{S}_3$

Consider $\mathcal{S}_3$ with the ordering: (), (2,3), (1,2), (1,2,3), (1,3,2), (1,3).
The cut $K = \{2\}$ (i.e. removing (2,3)) is not a closed one.
$F(K) = \{3, 4, 5, 6\}$
K extended by the forbidden elements $K \cup F(K) = \{2, 3, 4, 5, 6\}$ is closed.

1	2	3	4	5	6
2	1	4	3	6	5
3	5	1	6	2	4
4	6	2	5	1	3
5	3	6	1	4	2
6	4	5	2	3	1

Problem

The closed cut $K \cup F(K)$ corresponds to the trivial subgroup. However there are more closed cuts including K: $\{2,3,6\}$, $\{2,3,4,5\}$,$\{2,4,5,6\}$.

123456	123456	123456
214365	214365	214365
351624	351624	351624
462513	462513	462513
536142	536142	536142
645231	645231	645231

This means that we have to extend the cut one by one with the elements from the completion. Therefore we are back to the brute-force algorithm (actually even less efficient).

Heuristics

1. Diagonal closure.
2. "Rescuing"
3. Conjugacy.
4. Dynamic programming.

Definition (**diagonal completion of a cut**)

$$D(K) = \{i \in S \setminus K \mid \; M_{i,i} \in K\}$$

i.e. those elements not in the cut, whose diagonal contains an element in the cut.

The diagonal closure of a cut

Iterating

$$\Delta(K) := K \cup D(K)$$

Since cutting an element from a diagonal can be done only one way, we can extend the cut by its diagonal completion.

Algorithm 1: Calculating the diagonal closure of a cut.

```
input  : M multiplication table, K a cut
output: K extended to Δ(K)
repeat
    finished ← true;
    for i ∈ S \ K do
        if M_{i,i} ∈ K then
            K ← K ∪ {i};
            finished ← false;
until finished;
```

Again using the multiplication table of $\mathcal{S}_3$ if we cut by $K = \{5\}$ we get the following table:

1	2	3	4	5	6
2	1	4	3	6	5
3	5	1	6	2	4
4	6	2	5	1	3
5	3	6	1	4	2
6	4	5	2	3	1

5 appears in the diagonal for element 4, so $\Delta(\{5\}) = \{4, 5\}$. In this particular case $\Delta(\{4\})$ is also $\{4, 5\}$, but having the same closure is not a symmetric relation. For instance, $\Delta(\{1\}) = \{1, 2, 3, 6\}$ but $\Delta(\{6\}) = \{6\}$.

Exploiting symmetries

We use the most traditional approach to conjugacy for semigroups and define *G-conjugacy*. Elements $s, t \in S$ are G-conjugate, denoted by

$$s \sim_G t, \text{ if } s = g^{-1}tg \text{ for some } g \in G.$$

Here we act on the transformation representation.

Ways to use conjugacy:

- Whenever we find a subsemigroup we take the orbit under conjugation.
- For a non-semigroup subset we can also use the conjugacy class to prune the underlying search tree.
- We start cutting only from conjugacy class representatives.

…and of course we get the conjugacy classes as well.

Conjugacy classes of subsemigroups of $\mathcal{T}_2$

1144
1234
1324
1414

1144
1234
1324
1414

1144
1234
1324
1414

1144 1144
1234 1234
1324, 1324
1414 1414

1144
1234
1324
1414

1144 1144 1144
1234 1234 1234
1324 1324, 1324,
1414 1414 1414

"Rescuing elements"

Observation: There is a problem with trying to cut the identity from groups. After the diagonal closure the algorithm reverts back to full enumeration of the subsets of $S \setminus \Delta(K)$.

The "rescue" set of s relative to cut K:

$$R(K, F(K), s) := \{i \in S \setminus K \mid M_{s,i} \in F(K) \text{ or } M_{i,s} \in F(K)\}$$

What shall I remove if I want to keep s?

How to measure complexity/efficiency?

The number of visited cuts - the space complexity.

The number of visited cuts and the number of revisits.

$\mathcal{S}_3$	#Cuts	#Dups
basic	63,63	103,41
R	36,36	46,25
Δ	17,17	31,17
ΔR	14,14	19,13

$\mathcal{T}_2$	#Cuts	#Dups
basic	13,13	11,9
R	13,13	11,9
Δ	11,11	11,9
ΔR	11,11	11,9

Sing_3	#Cuts	#Dups
basic	?	?
Δ	88555,88555	691298,116767
R	6782,6782	20608,3672
ΔR	3764,3764	11764,2166

$\mathcal{T}_3$	#Cuts	#Dups
basic	?	?
Δ	1505328,1505328	15670601,2629323
R	44291,44291	206865,35713
ΔR	15664,15664	65104,11724

Easy test cases: Cyclic Groups

Cyclic groups - the number of subgroups is the number of divisors.

Cyclic groups of prime order - just 2 subgroups, but there is a bit of surprise.

n	2	3	5	7	11	13	17	19	23	29	31	37
#cuts	2	3	3	7	3	3	7	3	7	3	48	3

n	41	43	47	53	59	61	67	71	73	79	83	89
#cuts	7	9	7	3	3	3	3	7	83	7	3	51

n	97	101	103	107	109	113	127	131	137
#cuts	7	3	7	3	9	11	786	3	7

$\mathcal{T}_3$ data, the sizes of subsemigroups

Order	#occurences
1	3
2	10
3	19
4	28
5	38
6	42
7	38
8	30
9	25
10	14
11	12

Order	#occurences
12	7
13	3
14	1
15	3
16	2
17	2
21	1
22	1
23	1
24	1
27	1

Summary of Results

n	0	1	2	3	4
S_n	-	1,1	2,2	6,4	30,11
T_n	1,1	2,2	10,8	1299,283	
$T_n \setminus S_n$	1,1	1,1	4,3	600,123	

A215650, A215651 `http://oeis.org`

Progress with $\mathcal{T}_4$

$K_{4,2}$

3788251 ($\approx$ 3.8 million) subsemigroups in 162331 in conjugacy classes.

213268743 ($\approx$ 213 million) cuts checked, more than 10GB data, 80323087 ($\approx$ 80 million) revisits.

This data will be used to build the subsemigroup lattice from the bottom.

Also, once we have the subsemigroups of Sing_4, we can just them together with the subgroups and see what they generate.

Also, we can start from maximal subgroups.

Distribution of elements in multiplication tables

$\mathcal{T}_1$

Frequency	#elements
1	1

$\mathcal{T}_2$

Frequency	#elements
2	2
6	2

$\mathcal{T}_3$

Frequency	#elements
6	6
24	18
87	3

$\mathcal{T}_4$

Frequency	#elements
24	24
120	144
408	36
504	48
2200	4

	Frequency	#elements
$\mathcal{T}_5$	120	120
	720	1200
	2820	900
	3420	600
	11020	200
	16720	100
	84245	5

$\mathcal{T}_6$	720	720
	5040	10800
	22320	16200
	26640	7200
	78480	1800
	95760	7200
	143280	1800
	363600	300
	445680	450
	795600	180
	4492656	6

Non-synchronising transformation semigroups

	#subsemigroups	#conjugacy classes
$\mathcal{T}_2$	2	2
$\mathcal{T}_3$	64	20
$\mathcal{T}_4$	58610	3085

What to expect?

A software tool for finding subsemigroups of any transformation semigroup with less than $\approx$100 elements.

A database of all transformation semigroups on n points.

- $n \leq 3$ we have the data, included in the GAP package SEMIGROUPS.
- $n = 4$ It seems to be within reach with the same heuristics, just a bit more data juggling.
- $n = 5$ Probably the same idea may work with more new heuristics and solving big data handling difficulties.
- $n = 6$ Not with this idea.

Thank You!

Transformation (and other type) semigroups software

SEMIGROUPS

http://www-circa.mcs.st-and.ac.uk/~jamesm/citrus.php

Group & semigroup decomposition software:

SGPDEC http://sgpdec.sf.net

On computational semigroup theory:

http://compsemi.wordpress.com