TECHNISCHE UNIVERSITÄT WIEN

JOHANNES KEPLER UNIVERSITÄT LINZ

Counting and enumerating transformation monoids

Mike Behrisch[×][*][1]

[×]Institute of Discrete Mathematics and Geometry, Algebra Group,
TU Wien

[*]Institute for Algebra
JKU Linz

11th June 2023 • Tartu, Estonia

[1]Supported by Austrian Science Fund (FWF) grant P 33878.

Background story

- 2023, shortly before Easter:
  Colleague: 'How many transformation monoids on $\{0, 1, 2\}$?'
  Me: 'Don't know, but Sloane's `OEIS` should, right?'

Background story

- 2023, shortly before Easter:
  Colleague: 'How many transformation monoids on $\{0, 1, 2\}$?'
  Me: 'Don't know, but Sloane's `OEIS` should, right?'
- Sloane's *On-line Encyclopedia of Integer Sequences* (`https://oeis.org`): ?
  `transformation monoid`, `transformation semigroup`, `full semigroup` ... $\rightsquigarrow$ no result

Background story

- 2023, shortly before Easter:
  Colleague: 'How many transformation monoids on $\{0,1,2\}$?'
  Me: 'Don't know, but Sloane's `OEIS` should, right?'
- Sloane's *On-line Encyclopedia of Integer Sequences* (`https://oeis.org`): ?
  `transformation monoid, transformation semigroup, full semigroup ...` $\rightsquigarrow$ no result
- NextClosure algorithm: enumerate those monoids:
  $\rightsquigarrow$ 699 monoids colleague satisfied

Background story

- 2023, shortly before Easter:
  Colleague: 'How many transformation monoids on $\{0, 1, 2\}$?'
  Me: 'Don't know, but Sloane's `OEIS` should, right?'
- Sloane's *On-line Encyclopedia of Integer Sequences* (`https://oeis.org`): ?
  `transformation monoid, transformation semigroup, full semigroup ...` $\rightsquigarrow$ no result
- NextClosure algorithm: enumerate those monoids:
  $\rightsquigarrow$ 699 monoids colleague satisfied
- $1, 6, 699 \rightsquigarrow$ `https://oeis.org/search?q=1%2C6%2C699`
  `A343140`: 'Number of submonoids of the monoid of maps from an *n*-element set to itself.'
  (reason why we both didn't find it)

A343140

1, 6, 699

Links	Jannik Hess, Automorphism groups of monoids acting on number fields, Bachelor Thesis, 2019.
Keyword	`bref`, `hard`, `nonn`, `more`
Author	Max Alekseyev, Jan 27 2022

Keywords

`bref`	Sequence is too short to do any analysis with
`hard`	Next term is not known and may be hard to find. Would someone please extend this sequence?
`nonn`	A sequence of nonnegative numbers
`more`	More terms are needed! Would someone please extend this sequence? We need enough terms to fill about three lines on the screen.

A343140

1, 6, 699

Links	Jannik Hess, Automorphism groups of monoids acting on number fields, Bachelor Thesis, 2019.
Keyword	`bref`, `hard`, `nonn`, `more`
Author	Max Alekseyev, Jan 27 2022

Keywords

`bref`	Sequence is too short to do any analysis with
`hard`	Next term is not known and may be hard to find. Would someone please extend this sequence? Let's try!
`nonn`	A sequence of nonnegative numbers
`more`	More terms are needed! Would someone please extend this sequence? We need enough terms to fill about three lines on the screen.

A343140

1, 6, 699

Links: Jannik Hess, Automorphism groups of monoids acting on number fields, Bachelor Thesis, 2019.

Keyword: `bref`, `hard`, `nonn`, `more`

Author: Max Alekseyev, Jan 27 2022

Keywords

`bref` Sequence is too short to do any analysis with

`hard` Next term is not known and may be hard to find. Would someone please extend this sequence? Let's try!

`nonn` A sequence of nonnegative numbers

`more` More terms are needed! Would someone please extend this sequence? We need enough terms to fill about three lines on the screen. Let's not be too ambitious

Bernhard Ganter's NextClosure algorithm

A general purpose tool to enumerate all closed sets of a closure operator on a finite set...

Setting

- finite $M = \{m_1 < m_2 < m_3 < \cdots < m_k\}$ linearly ordered
- $\langle\,\rangle\colon 2^M \longrightarrow 2^M$ a closure operator, $\mathcal{F} = \{\, A \in 2^M \mid \langle A \rangle = A \}$
- lexicographic order of 2^M w.r.t. $(M, <)$: for $A, B \in 2^M$:

  $$A <_m B \iff m \in B \setminus A \wedge A \cap \{x \in M \mid x < m\} = B \cap \{x \in M \mid x < m\}$$

  $$A <_{\text{le}} B \iff \exists m \in M\colon A <_m B.$$

  linear order on 2^M.

NextClosure

enumerates closure system $\mathcal{F}$ in lexicographic order beginning with $\langle \emptyset \rangle$

Bernhard Ganter's NextClosure algorithm

A general purpose tool to enumerate all closed sets of a closure operator on a finite set...

Setting

- finite $M = \{m_1 < m_2 < m_3 < \cdots < m_k\}$ linearly ordered
- $\langle\,\rangle\colon 2^M \longrightarrow 2^M$ a closure operator, $\mathcal{F} = \{\, A \in 2^M \mid \langle A\rangle = A\}$
- lexicographic order of 2^M w.r.t. $(M, >)$: for $A, B \in 2^M$:

$$A <_m B \iff m \in B \setminus A \wedge \\ A \cap \{x \in M \mid x > m\} = B \cap \{x \in M \mid x > m\}$$

$$A <_{\mathrm{le}} B \iff \exists m \in M\colon A <_m B.$$

linear order on 2^M.

NextClosure

enumerates closure system $\mathcal{F}$ in lexicographic order beginning with $\langle\emptyset\rangle$

Bernhard Ganter's NextClosure algorithm

A general purpose tool to enumerate all closed sets of a closure operator on a finite set…

Setting

- finite $M = \{m_1 < m_2 < m_3 < \cdots < m_k\}$ linearly ordered
- $\langle\,\rangle\colon 2^M \longrightarrow 2^M$ a closure operator, $\mathcal{F} = \{\, A \in 2^M \mid \langle A\rangle = A \}$
- lexicographic order of 2^M w.r.t. $(M, >)$: for $A, B \in 2^M$:

  $A <_m B \iff A(m) = 0 < 1 = B(m) \wedge \forall x > m\colon A(x) = B(x)$

  $A <_{\text{le}} B \iff \exists m \in M\colon A <_m B.$

  linear order on 2^M.

NextClosure

enumerates closure system $\mathcal{F}$ in lexicographic order beginning with $\langle\emptyset\rangle$

NextClosure theory

Given $A \in 2^M$ such that $\exists B \in \mathcal{F}\colon A <_{\text{le}} B$ $\qquad$ $<_{\text{le}}$ w.r.t. $(M, >)$

Theorem (Ganter)

The $<_{\text{le}}$-smallest $B \in \mathcal{F}$ with $A <_{\text{le}} B$ is

$$A \oplus m := \langle (A \cap \{x \in M \mid x > m\}) \cup \{m\} \rangle$$

where $m \in M$ is $<$-least (i.e., $>$-largest) w.r.t. $A <_m A \oplus m$ $\quad (\implies m \notin A)$

If no $m \in M$ with $A <_m A \oplus m$ exists,
there is no $B \in \mathcal{F}$ with $A <_{\text{le}} B$.

NextClosure code

Algorithm 1: NextClosure(A)

Data: $A \in 2^M$

Result: the $<_{\text{le}}$-next $B \in 2^M$, $A <_{\text{le}} B$ if it exists, else `Error`

```
for m ∈ M in <-ascending order do
    if m ∈ A then                                   // A <_m A ⊕ m impossible
        A := A \ {m}                                // remove m
    else
        B := ⟨A ∪ {m}⟩                              // compute A ⊕ m
        if B \ A contains no element x > m then     // A <_m A ⊕ m
            return B
        else
            ignore B completely and continue with the next m
return Error                      // no further closure beyond A exists
```

NextClosure code

Algorithm 2: NextClosure(A) with characteristic vectors

Data: $A \in 2^M$

Result: the $<_{\mathrm{le}}$-next $B \in 2^M$, $A <_{\mathrm{le}} B$ if it exists, else `Error`

for $m \in M$ *in $<$-ascending order* **do**

- **if** $A(m) = 1$ **then** // $A <_m A \oplus m$ impossible
  - $A(m) := 0$ // remove m
- **else**
  - $B := A;\ B(m) := 1;\ B := \langle B \rangle$ // copy A, set m, close
  - **if** $\neg \exists x > m \colon B(x) > A(x)$ **then** // $A <_m A \oplus m$
    - **return** B
  - **else**
    - ignore B completely and continue with the next m

return `Error` // no further closure beyond A exists

Enumerating all closures

Algorithm 3: AllClosures

Data: closure operator $\langle\,\rangle$

Output: listing of all closed sets $A = \langle A \rangle$ in lexicographic order

```
A := ⟨∅⟩
while A ≠ Error do          // there is a next closure after A
    List A                  // print to screen or store otherwise
    A := NextClosure(A)
```

Both algorithms in one

Algorithm 4: AllClosures with NextClosure inlined

Data: closure operator $\langle\,\rangle$

Output: listing of all closed sets $A = \langle A \rangle$ in lexicographic order

```
List A := ⟨∅⟩
repeat
    stop := true              // assume A is the last closure, but revert if necessary
    for m ∈ M in <-ascending order do
        if A(m) = 1 then A(m) := 0     // if A <_m A ⊕ m impossible, remove m
        else
            B := A;  B(m) := 1;  B := ⟨B⟩              // B := A ⊕ m
            if ¬∃ x > m: B(x) > A(x) then              // A <_m A ⊕ m
                List A := B          // copy next closure B onto A
                stop := false        // found next closure, hence need to repeat
                break for-loop over m ∈ M
until stop                    // there is no next closure after A
```

Speeding up the core part

Part of AllClosures: (with NextClosure inlined)

$B := A;\quad B(m) := 1$

$B := \langle B \rangle$ // $B := A \oplus m$

if $\neg\exists x > m \colon B(x) > A(x)$ **then** // $A <_m A \oplus m$

  List $A := B$ // copy next closure B onto A

  stop := **false** // found next closure, hence need to repeat

  break for-loop over $m \in M$

After $B := \langle B \rangle$: Question $\{x > m \mid x \in B \setminus A\} \neq \emptyset$?

Speeding up the core part

Part of AllClosures: (with NextClosure inlined)

$B := A$; $B(m) := 1$

$B := \langle B \rangle$ // $B := A \oplus m$

if $\neg\exists x > m\colon B(x) > A(x)$ **then** // $A <_m A \oplus m$

  List $A := B$ // copy next closure B onto A

  stop := **false** // found next closure, hence need to repeat

  break for-loop over $m \in M$

After $B := \langle B \rangle$: Question $\{x > m \mid x \in B \setminus A\} \neq \emptyset$?

$\Longleftrightarrow$ Does application of $\langle B \rangle$ generate a new element $x > m$?

If yes, then computation of $\langle B \rangle$ does not need to be completed since B is discarded.

Speeding up the core part

Part of AllClosures: (with NextClosure inlined)

$B := A; \quad B(m) := 1$

$B := \langle B \rangle$ // $B := A \oplus m$

if $\neg\exists x > m \colon B(x) > A(x)$ **then** // $A <_m A \oplus m$

- List $A := B$ `// copy next closure` B `onto` A
- stop := **false** `// found next closure, hence need to repeat`
- **break** for-loop over $m \in M$

After $B := \langle B \rangle$: Question $\{x > m \mid x \in B \setminus A\} \neq \emptyset$?

$\Longleftrightarrow$ Does application of $\langle B \rangle$ generate a new element $x > m$?

If yes, then computation of $\langle B \rangle$ does not need to be completed since B is discarded.

For $M = T_n$ (full transformation monoid on n), we will shortcut generation $\langle B \rangle = \langle B \rangle_{T_n}$ appropriately

Shortcut code for generating the submonoid $\langle B\rangle_{T_n}$

$B := A$; $B(m) := 1$; $B := \langle B\rangle$ // $B := A \oplus m$

if $\neg\exists x > m\colon B(x) > A(x)$ **then** … // $A <_m A \oplus m$

Improved part of AllClosures: (with NextClosure inlined)

```
B := A;  B(m) := 1; B(id) = 1                 // copy A, add m and id
notgennew := true          // assume ⟨B⟩ does not generate x > m
repeat                                             // compute ⟨B⟩
    closed := true          // assume B is closed, revert if not
    for x, y ∈ B do          // i.e., x, y ∈ M with B(x) = B(y) = 1
        if B(x ∘ y) = 0 then
            B(x ∘ y) := 1; closed := false     // product is new, hence repeat
            if x ∘ y > m then
                notgennew := false        // B will be discarded, thus closing
                break repeat-until-loop         // does not have to be completed
until closed                                        // B is closed
if notgennew then …
```

Results

Number of all transformation monoids on n elements

n	1	2	3	4
$\lvert \mathrm{Sub}(T_n) \rvert$	1	6	699	?

Results

Number of all transformation monoids on n elements

n	1	2	3	4
$\lvert \mathrm{Sub}(T_n) \rvert$	1	6	699	?

Audience poll: $x = \log_{10}$?

- $x < 5$
- $5 \leq x < 6$
- $6 \leq x < 7$
- $7 \leq x < 9$
- $9 \leq x < 10$
- $x \geq 10$

Results

Number of all transformation monoids on n elements

n	1	2	3	4
$\lvert \mathrm{Sub}(T_n) \rvert$	1	6	699	$\approx 1.58 \cdot 10^9$

($\approx$ 1 day)

Results

Number of all transformation monoids on n elements

n	1	2	3	4	
$\lvert\mathrm{Sub}(T_n)\rvert$	1	6	699	$\approx 1.58 \cdot 10^9$	($\approx$ 1 day)

Number of transformation monoids with all constants (constantive)

n	1	2	3	4	
$\lvert\{ B \leq T_n \mid C_n \subseteq B\}\rvert$	1	2	342	$\approx 1.25 \cdot 10^9$	($\approx$ 1 day)
% of $\lvert\mathrm{Sub}(T_n)\rvert$	100	≈ 33	≈ 49	≈ 79	

Results

Number of all transformation monoids on n elements

n	1	2	3	4
$\lvert \mathrm{Sub}(T_n) \rvert$	1	6	699	$\approx 1.58 \cdot 10^9$

($\approx$ 1 day)

Number of transformation monoids with all constants (constantive)

n	1	2	3	4
$\lvert \{ B \leq T_n \mid C_n \subseteq B \} \rvert$	1	2	342	$\approx 1.25 \cdot 10^9$
% of $\lvert \mathrm{Sub}(T_n) \rvert$	100	≈ 33	≈ 49	≈ 79

($\approx$ 1 day)

Number of non-constantive transformation monoids

n	1	2	3	4
$\lvert \{ B \leq T_n \mid C_n \subseteq B \} \rvert$	0	4	357	$\approx 328 \cdot 10^6$
% of $\lvert \mathrm{Sub}(T_n) \rvert$	0	≈ 67	≈ 51	≈ 21

($\approx$ 3.25 h)

Results

Number of all transformation monoids on n elements

n	1	2	3	4
$\lvert \mathrm{Sub}(T_n) \rvert$	1	6	699	$\approx 1.58 \cdot 10^9$

($\approx$ 1 day)

Number of transformation monoids with all constants (constantive)

n	1	2	3	4
$\lvert \{ B \leq T_n \mid C_n \subseteq B \} \rvert$	1	2	342	$\approx 1.25 \cdot 10^9$
% of $\lvert \mathrm{Sub}(T_n) \rvert$	100	≈ 33	≈ 49	≈ 79

($\approx$ 1 day)

Number of transformation monoids without any constants (constant-free)

n	1	2	3	4	5
$\lvert \{ B \leq T_n \mid C_n \cap B = \emptyset \} \rvert$	0	2	39	30741	$> 46 \cdot 10^9$ *)
% of $\lvert \mathrm{Sub}(T_n) \rvert$	0	≈ 33	≈ 5.6	≈ 0.0019	?

*) value updated post-lecture (1 Aug 2023)

Future steps

1. wait for the constant-free monoids on $5 = \{0, 1, 2, 3, 4\}$ to finish
   (guess: about 5 months more)
2. try to enumerate representatives up to isomorphism

Future steps

1. wait for the constant-free monoids on $5 = \{0, 1, 2, 3, 4\}$ to finish (guess: about 5 months more)
2. try to enumerate representatives up to isomorphism

Questions/Remarks?