Computing with Semigroup Congruences

Michael Torpey

University of St Andrews

2016-03-17

Congruences

Definition

A **congruence** on a semigroup S is an equivalence relation $\rho \subseteq S \times S$ such that

$$(x, y) \in \rho \quad \Rightarrow \quad (ax, ay), (xa, ya) \in \rho,$$

or equivalently,

$$(x, y), (s, t) \in \rho \quad \Rightarrow \quad (xs, yt) \in \rho,$$

for all $x, y, a, s, t \in S$.

(we may write $x \ \rho \ y$ for $(x, y) \in \rho$)

Simple ways to represent congruences

- List of pairs: $\{(x_1, x_3), (x_1, x_9), (x_{42}, x_{11}), \dots\}$
- Partition: $\{\{x_1, x_3, x_9, x_{14}\}, \{x_2\}, \{x_4, x_5, x_8\}, \dots\}$
- ID list: $(1, 2, 1, 3, 3, 4, 5, 3, 1, \dots)$

Generating pairs

- Let $\mathbf{R} \subseteq S \times S$ be a set of pairs.
- Let ρ be the least congruence on S containing all the pairs in $\mathbf{R}$.
- We call $\mathbf{R}$ a **generating set** for ρ.
- Two elements a and b are ρ-related if and only if there exists a sequence

  $$a = a_1 \to a_2 \to \cdots \to a_n = b$$

  such that for each i there exist x, y, z, t such that

  $$a_i = xzy, \quad a_{i+1} = xty,$$

  and either (z, t) or (t, z) is in ρ.
- Finding whether two elements are ρ-related has worst-case complexity $O(|S|^2)$.

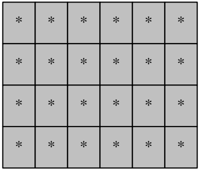
Simple and 0-simple semigroups

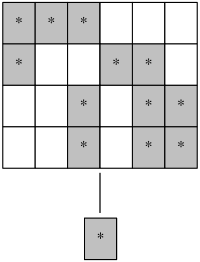
Definition

A semigroup S without zero is **simple** if it has no proper ideals.

Definition

A semigroup S with zero is **0-simple** if its only ideals are $\{0\}$ and S.





The Rees Theorem

Theorem (Rees)

Every completely 0-simple semigroup is isomorphic to a Rees 0-matrix semigroup

$$\mathcal{M}^0[G; I, \Lambda; P],$$

where G is a group and P is regular. Conversely, every such Rees 0-matrix semigroup is completely 0-simple.

Linked triples

Definition

For a finite 0-simple Rees 0-matrix semigroup $\mathcal{M}^0[G; I, \Lambda; P]$, a **linked triple** is a triple

$$(N, \mathcal{S}, \mathcal{T})$$

consisting of a normal subgroup $N \trianglelefteq G$, an equivalence relation $\mathcal{S}$ on I and an equivalence relation $\mathcal{T}$ on Λ, such that the following are satisfied:

1. $\mathcal{S}$ only relates columns which have zeroes in the same places,
2. $\mathcal{T}$ only relates rows which have zeroes in the same places,
3. For all $i, j \in I$ and $\lambda, \mu \in \Lambda$ such that $p_{\lambda i}, p_{\lambda j}, p_{\mu i}, p_{\mu j} \neq 0$ and either $(i, j) \in \mathcal{S}$ or $(\lambda, \mu) \in \mathcal{T}$, we have that $q_{\lambda\mu ij} \in N$, where

$$q_{\lambda\mu ij} = p_{\lambda i} p_{\mu i}^{-1} p_{\mu j} p_{\lambda j}^{-1}.$$

Linked triples

A finite 0-simple semigroup S has a bijection Γ between its *non-universal* congruences and its linked triples,

$$\Gamma : \rho \mapsto (N, \mathcal{S}, \mathcal{T})$$

We may write ρ as $[N, \mathcal{S}, \mathcal{T}]$.
Two non-zero elements (i, a, λ) and (j, b, μ) are ρ-related if and only if

1. $(i, j) \in \mathcal{S}$;
2. $(\lambda, \mu) \in \mathcal{T}$;
3. $(p_{\xi i} a p_{\lambda x})(p_{\xi j} b p_{\mu x})^{-1} \in N$ for some $x \in I, \xi \in \Lambda$ such that $p_{\xi i}, p_{\xi j}, p_{\lambda x}, p_{\mu x} \neq 0$.

This can be determined in constant time.
(i, a, λ) is related to 0 only in the universal congruence $S \times S$.

Finding a congruence's linked triple

- Clearly, if we have a congruence's linked triple, we should use it for all calculations. But what if we do not?
- We want an algorithm to find a linked triple $(N, \mathcal{S}, \mathcal{T})$ from a set of generating pairs $\mathbf{R}$.
- First observe the following:

Lemma

If $(N_1, \mathcal{S}_1, \mathcal{T}_1)$ *and* $(N_2, \mathcal{S}_2, \mathcal{T}_2)$ *are linked triples such that*

$$N_1 \leq N_2, \quad \mathcal{S}_1 \subseteq \mathcal{S}_2, \quad \mathcal{T}_1 \subseteq \mathcal{T}_2,$$

then $[N_1, \mathcal{S}_1, \mathcal{T}_1] \subseteq [N_2, \mathcal{S}_2, \mathcal{T}_2]$.

Finding a congruence's linked triple

Our strategy:

- Find any elements that must be in N because of $\mathbf{R}$.
- Find any pairs of columns that must be in $\mathcal{S}$ because of $\mathbf{R}$.
- Find and pairs of rows that must be in $\mathcal{T}$ because of $\mathbf{R}$.
- Add any elements and pairs necessary for $(N, \mathcal{S}, \mathcal{T})$ to be *linked*.
- Watch out for anything that would force this to be the universal congruence $S \times S$.

The algorithm

Require: $S = \mathcal{M}^0[G; I, \Lambda; P]$ is a finite 0-simple Rees 0-matrix semigroup

procedure $\textsc{LinkedTriple}(\mathbf{R})$

$N := \varnothing$

$\mathcal{S} := \Delta_I$

$\mathcal{T} := \Delta_\Lambda$

for $(x, y) \in \mathbf{R}$ **do**

if $x = y$ **then**

Skip this pair

else if $x = 0$ **or** $y = 0$ **then**

return "Universal Congruence" (no linked triple)

end if

Let $x = (i, a, \lambda)$

Let $y = (j, b, \mu)$

if $(i, j) \notin \varepsilon_I$ **or** $(\lambda, \mu) \notin \varepsilon_\Lambda$ **then**

return "Universal Congruence" (no linked triple)

end if

...

The algorithm

Require: $S = \mathcal{M}^0[G; I, \Lambda; P]$ is a finite 0-simple Rees 0-matrix semigroup
procedure $\textsc{LinkedTriple}(\mathbf{R})$
...
for $(x, y) \in \mathbf{R}$ **do**
...
▷ *Combine row and column classes*
$\textsc{Union}(\mathcal{S}, i, j)$
$\textsc{Union}(\mathcal{T}, \lambda, \mu)$

▷ *Add generators for normal subgroup*
Choose $\nu \in \Lambda$ such that $p_{\nu i} \neq 0$
Choose $k \in I$ such that $p_{\lambda k} \neq 0$
Add $(p_{\nu i} a p_{\lambda k})(p_{\nu j} b p_{\mu k})^{-1}$ to N
...

Require: $S = \mathcal{M}^0[G; I, \Lambda; P]$ is a finite 0-simple Rees 0-matrix semigroup

procedure $\textsc{LinkedTriple}(\mathbf{R})$

. . .

for $(x, y) \in \mathbf{R}$ **do**

. . .

▷ *Add more generators for normal subgroup*

for $\xi \in \Lambda \setminus \{\nu\}$ such that $p_{\xi i} \neq 0$ **do**

Add $q_{\nu\xi ij}$ to N

end for

for $x \in I \setminus \{k\}$ such that $p_{\lambda x} \neq 0$ **do**

Add $q_{\lambda\mu kx}$ to N

end for

end for

$N := \langle\langle N \rangle\rangle$

return $(N, \mathcal{S}, \mathcal{T})$

end procedure

The algorithm

- Finding the linked triple is fast.
- Doesn't require enumerating S.
- Transforms an $O(|S|^2)$ time problem into $O(1)$.
- Other information can be found from $(N, \mathcal{S}, \mathcal{T})$: number of congruence classes, size of congruence classes, etc.
- A list of all congruences on S can be found.

Semigroups package for GAP

- Generic semigroups: **generating pairs**.
- Simple & 0-simple semigroups: **linked triples**.
- Inverse semigroups: **kernel and trace**.
- Rees congruences are also implemented.