Rewriting Generators for (Sub)Semigroups

Nik Ruskuc

`nik@mcs.st-and.ac.uk`

School of Mathematics and Statistics, University of St Andrews

Porto, 6 July 2009

University
of
St Andrews

Setting the Scene

S – a semigroup/monoid

$S = \langle A \rangle$

$T \leq S$

Setting the Scene

S – a semigroup/monoid

$S = \langle A \rangle$

$T \leq S$

Find a generating set for T.

Setting the Scene

S – a semigroup/monoid

$S = \langle A \rangle$

$T \leq S$

Find a 'nice' generating set for T.

People

- Robert Gray, NR, Green index and finiteness conditions for semigroups, J. Algebra 320 (2008), 3145–3164.
- Alan J. Cain, Robert Gray, NR, Green index in semigroups: generators and related finiteness conditions, about to be submitted.
- Also: C.M. Campbell, M. Hoffmann, E.F. Robertson, R.M Thomas.
- Related: R. Gray, V. Maltcev, J.D. Mitchell, NR, Ideals, finiteness conditions and Green index for subsemigroups, about to be submitted.

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 \; a_4 a_5 \; a_6 \; a_7 a_8 a_9 \; \ldots$$

$\notin T$ $\in T$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 \; a_4 a_5 \; a_6 \; a_7 a_8 a_9 \; \dots$$

$\not\in T$ $\qquad\qquad$ $\in T$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 \; a_4 a_5 \; a_6 \; a_7 a_8 a_9 \; \ldots$$

$$\not\in T \qquad\qquad\qquad \in T$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 \; a_4 a_5 \; a_6 \; a_7 a_8 a_9 \; \dots$$

$$\notin T \qquad\qquad \in T$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 \mid a_4 a_5\ a_6\ a_7 a_8 a_9\ \dots$$

$$\not\in T \qquad\qquad \in T$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 \mid a_4 a_5 \; a_6 \; a_7 a_8 a_9 \; \ldots$$

$$\notin T \qquad\qquad \in T$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 \; a_6 \; a_7 a_8 a_9 \; \ldots$$

$$\not\in T \qquad\qquad \in T$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 \ a_7 a_8 a_9 \ \ldots$$

$$\not\in T \qquad\qquad \in T$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 \,|\, a_4 a_5 \,|\, a_6 \; a_7 a_8 a_9 \; \ldots$$

$$\notin T \qquad\qquad\qquad \in T$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 \quad \ldots$$

$$\notin T \qquad\qquad \in T$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 \ \ldots$$

$$\notin T \qquad \qquad \in T$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 \; \ldots$$

$$\notin T \qquad\qquad \in T$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 | \dots$$

$\notin T$ $\qquad\qquad\qquad\qquad$ $\in T$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 | \dots$$

$$\notin T \qquad \in T$$

Example

$S = \{a, b\}^*$

$T = \{\text{words of even length}\}$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 | \dots$$

$$\notin T \qquad\qquad \in T$$

Example

$S = \{a, b\}^*$

$T = \{\text{words of even length}\}$

$$ab\ aa\ ba\ bb$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 | \dots$$

$$\notin T \qquad\qquad \in T$$

Example

$S = \{a, b\}^*$

$T = \{\text{words of even length}\}$

$$ab\ aa\ ba\ bb$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 | \dots$$

$$\notin T \qquad\qquad \in T$$

Example

$S = \{a, b\}^*$

$T = \{\text{words of even length}\}$

$$ab\ aa\ ba\ bb$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 | \ldots$$

$\not\in T$ $\in T$

Example

$S = \{a, b\}^*$

$T = \{\text{words of even length}\}$

$$ab | aa\ ba\ bb$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 | \ldots$$

$\not\in T$ $\quad\quad\quad\quad$ $\in T$

Example

$S = \{a, b\}^*$

$T = \{\text{words of even length}\}$

$$ab|aa\ ba\ bb$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 | \ldots$$

$$\notin T \qquad\qquad \in T$$

Example

$S = \{a, b\}^*$

$T = \{\text{words of even length}\}$

$$ab | aa\ ba\ bb$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 | \dots$$

$$\not\in T \qquad \in T$$

Example

$S = \{a, b\}^*$

$T = \{\text{words of even length}\}$

$$ab | aa | ba\ bb$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 | \dots$$

$\notin T$ $\qquad\qquad\qquad$ $\in T$

Example

$S = \{a, b\}^*$

$T = \{\text{words of even length}\}$

$$ab | aa | ba\ bb$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 | \dots$$

$$\not\in T \qquad \in T$$

Example

$S = \{a, b\}^*$

$T = \{\text{words of even length}\}$

$$ab | aa | ba\ bb$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 | \ldots$$

$$\notin T \qquad\qquad \in T$$

Example

$S = \{a, b\}^*$

$T = \{\text{words of even length}\}$

$$ab|aa|ba|bb$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 | \dots$$

$$\not\in T \qquad \in T$$

Example

$S = \{a, b\}^*$

$T = \{\text{words of even length}\}$

$$ab|aa|ba|bb$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 | \ldots$$

$$\notin T \qquad\qquad \in T$$

Example

$S = \{a, b\}^*$

$T = \{\text{words of even length}\}$

$$ab|aa|ba|bb$$

A first naive approach

Given a 'long' word from A^* representing an element of T, 'decompose' it into 'shorter' members of T.

$$a_1 a_2 a_3 | a_4 a_5 | a_6 | a_7 a_8 a_9 | \ldots$$

$\notin T$ $\in T$

Example

$S = \{a, b\}^*$

$T = \{\text{words of even length}\}$

$$ab|aa|ba|bb$$

Hence $T = \langle a^2, ab, ba, b^2 \rangle$.

Variation: finite complement (Rees index)

Suppose $|S \setminus T| = r < \infty$

Variation: finite complement (Rees index)

Suppose $|S \setminus T| = r < \infty$

Consider a product $a_1 a_2 \dots a_{r+1}$. Then:

Variation: finite complement (Rees index)

Suppose $|S \setminus T| = r < \infty$

Consider a product $a_1 a_2 \dots a_{r+1}$. Then:

(i) $\exists k \ : \ a_1 \dots a_k \in T$, or

Variation: finite complement (Rees index)

Suppose $|S \setminus T| = r < \infty$

Consider a product $a_1 a_2 \dots a_{r+1}$. Then:

(i) $\exists k \ : \ a_1 \dots a_k \in T$, or

(ii) $\exists k, l \ : \ a_1 \dots a_k = a_1 \dots a_l$.

Variation: finite complement (Rees index)

Suppose $|S \setminus T| = r < \infty$

Consider a product $a_1 a_2 \dots a_{r+1}$. Then:

(i) $\exists k \ : \ a_1 \dots a_k \in T$, or

(ii) $\exists k, l \ : \ a_1 \dots a_k = a_1 \dots a_l$.

Theorem (Jura 1978)

If $S = \langle A \rangle$ *then*

$$T = \langle \{ uav \ : \ u, v \in (S \setminus T)^1, \ a \in A, \ ua, uav \in T \} \rangle.$$

Variation: finite complement (Rees index)

Suppose $|S \setminus T| = r < \infty$

Consider a product $a_1 a_2 \dots a_{r+1}$. Then:

(i) $\exists k \ : \ a_1 \dots a_k \in T$, or

(ii) $\exists k, l \ : \ a_1 \dots a_k = a_1 \dots a_l$.

Theorem (Jura 1978)

If $S = \langle A \rangle$ then

$$T = \langle \{ uav \ : \ u, v \in (S \setminus T)^1, \ a \in A, \ ua, uav \in T \} \rangle.$$

Corollary

S is finitely generated if and only if T is finitely generated.

Rewriting mapping

Rewriting mapping

Suppose $S = \langle A \rangle$, $T \leq S$, $T = \langle B \rangle$.

Rewriting mapping

Suppose $S = \langle A \rangle$, $T \leq S$, $T = \langle B \rangle$.

$\mathcal{L}(A, T) = \{$words from A^* representing elements of $T\}$

Rewriting mapping

Suppose $S = \langle A \rangle,\ T \leq S,\ T = \langle B \rangle$.

$\mathcal{L}(A, T) = \{$words from A^* representing elements of $T\}$

Then there exists a mapping

$$\phi\ :\ \mathcal{L}(A, T) \to B^*$$

such that $w \in \mathcal{L}(A, T)$ and $\phi(w) \in B^*$ represent the same element of T.

Rewriting mapping

Suppose $S = \langle A \rangle,\ T \leq S,\ T = \langle B \rangle$.

$\mathcal{L}(A, T) = \{$words from A^* representing elements of $T\}$

Then there exists a mapping

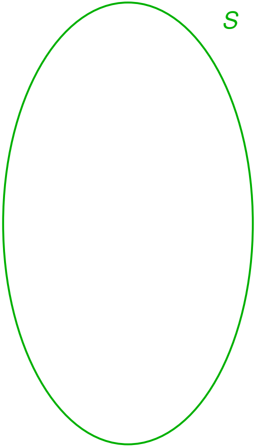
$$\phi\ :\ \mathcal{L}(A, T) \to B^*$$

such that $w \in \mathcal{L}(A, T)$ and $\phi(w) \in B^*$ represent the same element of T.

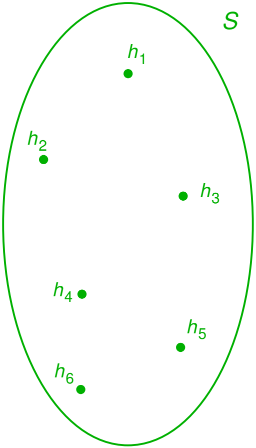
From the previous slides: For $w \in \mathcal{L}(A, T)$ let $w = uav$, where ua is the shortest prefix in $\mathcal{L}(A, T)$. Then

$$\phi(w) = \begin{cases} (ua) \cdot \phi(v) & \text{if } v \in \mathcal{L}(A, T) \\ w & \text{otherwise.} \end{cases}$$

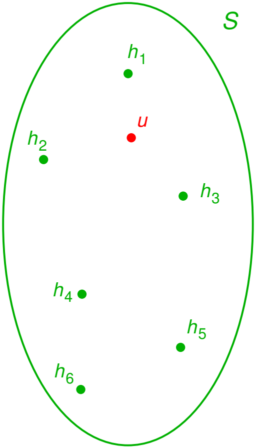
Another idea: ‘Coset representatives’



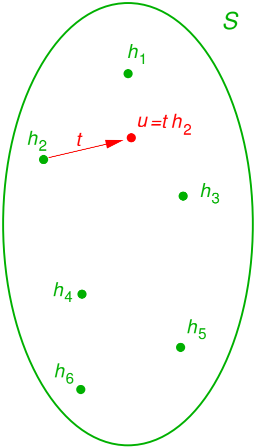
Another idea: 'Coset representatives'



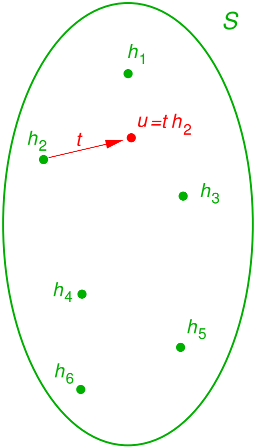
Another idea: 'Coset representatives'



Another idea: ‘Coset representatives’

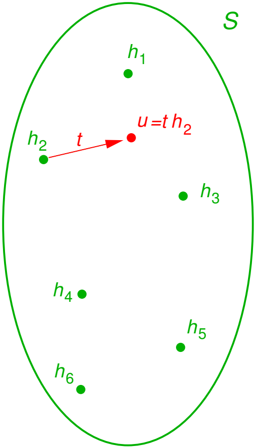


Another idea: 'Coset representatives'



$$\{h_i \;:\; i \in I\}$$

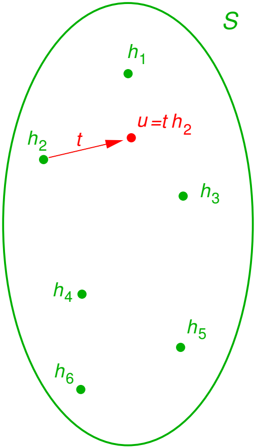
Another idea: 'Coset representatives'



$\{h_i \,:\, i \in I\}$

$h_i s = \tau(i, s) h_{\lambda(i,s)}$

Another idea: 'Coset representatives'



$\{h_i \,:\, i \in I\}$

$h_i s = \tau(i, s) h_{\lambda(i,s)}$

$\tau(i, s) \in T,\ \lambda(i, s) \in I$

$h_i s = \tau(i, s) h_{\lambda(i,s)}$

$h_i s = \tau(i, s) h_{\lambda(i,s)}$

$$a_1 a_2 a_3 \dots a_k$$

$h_i s = \tau(i, s) h_{\lambda(i,s)}$

$1 \; a_1 a_2 a_3 \ldots a_k$

$h_i s = \tau(i, s) h_{\lambda(i,s)}$

$$h_{i_1} a_1 a_2 a_3 \dots a_k$$

$h_i s = \tau(i,s) h_{\lambda(i,s)}$

$$
\begin{aligned}
& h_{i_1} a_1 a_2 a_3 \dots a_k \\
= \;& \tau(i_1, a_1) h_{\lambda(i_1,a_1)} a_2 a_3 \dots a_k
\end{aligned}
$$

$h_i s = \tau(i,s) h_{\lambda(i,s)}$

$$\begin{aligned} & h_{i_1} a_1 a_2 a_3 \dots a_k \\ = \;& \tau(i_1, a_1) \quad h_{i_2} \quad a_2 a_3 \dots a_k \end{aligned}$$

$h_i s = \tau(i, s) h_{\lambda(i,s)}$

$$
\begin{aligned}
& h_{i_1} a_1 a_2 a_3 \dots a_k \\
= \; & \tau(i_1, a_1) \quad h_{i_2} \quad a_2 a_3 \dots a_k \\
= \; & \dots
\end{aligned}
$$

$h_i s = \tau(i,s) h_{\lambda(i,s)}$

$$\begin{aligned} & h_{i_1} a_1 a_2 a_3 \dots a_k \\ = \; & \tau(i_1, a_1) \quad h_{i_2} \quad a_2 a_3 \dots a_k \\ = \; & \dots \\ = \; & \tau(i_1, a_1)\tau(i_2, a_2) \dots \tau(i_k, a_k) h_{i_{k+1}} \end{aligned}$$

$h_i s = \tau(i, s) h_{\lambda(i,s)}$

$$
\begin{aligned}
& h_{i_1} a_1 a_2 a_3 \dots a_k \\
= \;& \tau(i_1, a_1) \quad h_{i_2} \quad a_2 a_3 \dots a_k \\
= \;& \dots \\
= \;& \tau(i_1, a_1)\tau(i_2, a_2) \dots \tau(i_k, a_k) h_{i_{k+1}}
\end{aligned}
$$

'Problem': $h_{i_{k+1}}$ need not belong to T.

Groups: Schreier Theorem

$G = \langle A \rangle$ – a group

$H \leq G$

$\{h_i \, : \, i \in I\}$ – left coset representatives.

Groups: Schreier Theorem

$G = \langle A \rangle$ – a group

$H \leq G$

$\{h_i \ : \ i \in I\}$ – left coset representatives.

Theorem (Schreier 1927)

$H = \langle \{\tau(i, a) \ : \ i \in I, \ a \in A\} \rangle.$

Groups: Schreier Theorem

$G = \langle A \rangle$ – a group

$H \leq G$

$\{h_i \, : \, i \in I\}$ – left coset representatives.

Theorem (Schreier 1927)

$H = \langle \{\tau(i, a) \, : \, i \in I, \; a \in A\} \rangle.$

Corollary

G is finitely generated if and only if H is finitely generated.

Groups: Schreier Theorem

$G = \langle A \rangle$ – a group

$H \leq G$

$\{h_i \, : \, i \in I\}$ – left coset representatives.

Theorem (Schreier 1927)

$H = \langle \{\tau(i, a) \, : \, i \in I, \, a \in A\} \rangle.$

Corollary

G is finitely generated if and only if H is finitely generated.

Rewriting mapping:

$$\phi(i, aw) = \tau(i, a)\phi(\lambda(i, a), w)$$

Groups: Schreier Theorem

$G = \langle A \rangle$ – a group

$H \leq G$

$\{h_i \ : \ i \in I\}$ – left coset representatives.

Theorem (Schreier 1927)

$H = \langle \{\tau(i, a) \ : \ i \in I, \ a \in A\} \rangle.$

Corollary

G is finitely generated if and only if H is finitely generated.

Rewriting mapping:

$$\begin{aligned} \phi(i, aw) &= \tau(i, a)\phi(\lambda(i, a), w) \\ \phi(w) &= \phi(1, w). \end{aligned}$$

$\mathcal{H}$-classes

$\mathcal{H}$-classes

Exactly the same works in monoids for:

$\mathcal{H}$-classes

Exactly the same works in monoids for:

- maximal subgroups (group $\mathcal{H}$-classes);

$\mathcal{H}$-classes

Exactly the same works in monoids for:

- maximal subgroups (group $\mathcal{H}$-classes);
- arbitrary subgroups;

$\mathcal{H}$-classes

Exactly the same works in monoids for:

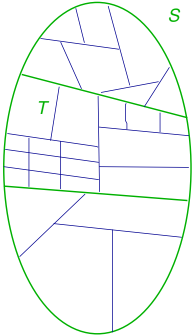
- maximal subgroups (group $\mathcal{H}$-classes);
- arbitrary subgroups;
- Schützenberger groups.

Green's relations

$$
\begin{aligned}
u\,\mathcal{L}\;\; v &\Leftrightarrow Su = Sv\\
u\,\mathcal{R}\;\; v &\Leftrightarrow uS = vS\\
u\,\mathcal{H}\;\; v &\Leftrightarrow u\,\mathcal{L}\;\; v \;\&\; u\,\mathcal{R}\;\; v
\end{aligned}
$$

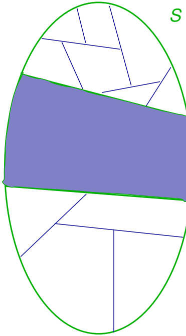
Relative Green's relations (Wallace 1963)

$$u\,\mathcal{L}^T v \;\Leftrightarrow\; Tu = Tv$$
$$u\,\mathcal{R}^T v \;\Leftrightarrow\; uT = vT$$
$$u\,\mathcal{H}^T v \;\Leftrightarrow\; u\,\mathcal{L}^T v \;\&\; u\,\mathcal{R}^T v$$
$$(T \leq S)$$



Relative Green's relations (Wallace 1963)

$$u\,\mathcal{L}^T v \;\Leftrightarrow\; Tu = Tv$$
$$u\,\mathcal{R}^T v \;\Leftrightarrow\; uT = vT$$
$$u\,\mathcal{H}^T v \;\Leftrightarrow\; u\,\mathcal{L}^T v \;\&\; u\,\mathcal{R}^T v$$
$$(T \leq S)$$



Definition

The Green index $[S:T]_G$ of T is S is the number of $\mathcal{H}^T$-classes in $S \setminus T$.

Green, Rees and group indices

Green, Rees and group indices

Proposition

(i) $|S \setminus T| < \infty \Rightarrow [S : T]_G < \infty$.

Green, Rees and group indices

Proposition

(i) $|S \setminus T| < \infty \Rightarrow [S : T]_G < \infty$.

(ii) $[G : H] < \infty \Leftrightarrow [G : H]_G < \infty$.

Green, Rees and group indices

Proposition

(i) $|S \setminus T| < \infty \Rightarrow [S : T]_G < \infty$.

(ii) $[G : H] < \infty \Leftrightarrow [G : H]_G < \infty$.

Proof

(ii)

$\mathcal{L}^H$-classes = right cosets

Green, Rees and group indices

Proposition

(i) $|S \setminus T| < \infty \Rightarrow [S : T]_G < \infty$.

(ii) $[G : H] < \infty \Leftrightarrow [G : H]_G < \infty$.

Proof

(ii)

$\mathcal{L}^H$-classes = right cosets

$\mathcal{R}^H$-classes = left cosets

$\mathcal{H}^H$-classes = intersections

Green rewriting

Green rewriting

$$h_i s = \tau(i, s) h_{\lambda(i,s)}$$

Green rewriting

$h_i s = \tau(i,s) h_{\lambda(i,s)}$

$\{h_i \,:\, i \in I\}$ – representatives of $\mathcal{L}^T$-classes

Green rewriting

$h_i s = \tau(i, s) h_{\lambda(i,s)}$

$\{h_i \, : \, i \in I\}$ – representatives of $\mathcal{L}^T$-classes, or $\mathcal{H}^T$-classes.

Green rewriting

$h_i s = \tau(i, s) h_{\lambda(i,s)}$

$\{h_i : i \in I\}$ – representatives of $\mathcal{L}^T$-classes, or $\mathcal{H}^T$-classes.

$$\begin{aligned} & h_{i_1} a_1 a_2 a_3 \dots a_k \\ = \; & \tau(i_1, a_1) h_{i_2} a_2 a_3 \dots a_k \\ = \; & \dots \\ = \; & \tau(i_1, a_1) \tau(i_2, a_2) \dots \tau(i_k, a_k) h_{i_{k+1}} \end{aligned}$$

Green rewriting

$h_i s = \tau(i,s) h_{\lambda(i,s)}$

$\{h_i \, : \, i \in I\}$ – representatives of $\mathcal{L}^T$-classes, or $\mathcal{H}^T$-classes.

$$\begin{aligned} & h_{i_1} a_1 a_2 a_3 \ldots a_k \\ = \; & \tau(i_1, a_1) h_{i_2} a_2 a_3 \ldots a_k \\ = \; & \ldots \\ = \; & \tau(i_1, a_1)\tau(i_2, a_2) \ldots \tau(i_k, a_k) h_{i_{k+1}} \end{aligned}$$

$$\lambda(i,s) : \begin{cases} h_i s \in H_{\lambda(i,s)} & \text{if } h_i s \in S \setminus T \\ 1 & \text{otherwise} \end{cases}$$

Example

Example

Set: $S = \mathbb{Z}_2 \times \mathbb{Z} \times \mathbb{Z}_2 \cup \{0\}$

Multiplication: $(a, b, c)(d, e, f) = \begin{cases} (a, b + e, f) & \text{if } c = d = 0 \\ 0 & \text{otherwise} \end{cases}$

Example

Set: $S = \mathbb{Z}_2 \times \mathbb{Z} \times \mathbb{Z}_2 \cup \{0\}$

Multiplication: $(a, b, c)(d, e, f) = \begin{cases} (a, b+e, f) & \text{if } c = d = 0 \\ 0 & \text{otherwise} \end{cases}$

$T = \{(a, b, c) \ : \ a \geq c\} \cup \{0\}$

Example

Set: $S = \mathbb{Z}_2 \times \mathbb{Z} \times \mathbb{Z}_2 \cup \{0\}$

Multiplication: $(a, b, c)(d, e, f) = \begin{cases} (a, b + e, f) & \text{if } c = d = 0 \\ 0 & \text{otherwise} \end{cases}$

$T = \{(a, b, c) \,:\, a \geq c\} \cup \{0\}$

$S \setminus T = \{(0, x, 1) \,:\, x \in \mathbb{Z}\}$

Example

Set: $S = \mathbb{Z}_2 \times \mathbb{Z} \times \mathbb{Z}_2 \cup \{0\}$

Multiplication: $(a,b,c)(d,e,f) = \begin{cases} (a, b+e, f) & \text{if } c = d = 0 \\ 0 & \text{otherwise} \end{cases}$

$T = \{(a,b,c) \ : \ a \geq c\} \cup \{0\}$

$S \setminus T = \{(0,x,1) \ : \ x \in \mathbb{Z}\}$

$(0, y-x, 0)(0,x,1) = (0,y,1)$

Example

Set: $S = \mathbb{Z}_2 \times \mathbb{Z} \times \mathbb{Z}_2 \cup \{0\}$

Multiplication: $(a, b, c)(d, e, f) = \begin{cases} (a, b+e, f) & \text{if } c = d = 0 \\ 0 & \text{otherwise} \end{cases}$

$T = \{(a, b, c) \ : \ a \geq c\} \cup \{0\}$

$S \setminus T = \{(0, x, 1) \ : \ x \in \mathbb{Z}\}$

$(0, y - x, 0)(0, x, 1) = (0, y, 1)$

Hence $S \setminus T$ is an $\mathcal{L}^T$-class.

Example

Set: $S = \mathbb{Z}_2 \times \mathbb{Z} \times \mathbb{Z}_2 \cup \{0\}$

Multiplication: $(a, b, c)(d, e, f) = \begin{cases} (a, b+e, f) & \text{if } c = d = 0 \\ 0 & \text{otherwise} \end{cases}$

$T = \{(a, b, c) \ : \ a \geq c\} \cup \{0\}$

$S \setminus T = \{(0, x, 1) \ : \ x \in \mathbb{Z}\}$

$(0, y - x, 0)(0, x, 1) = (0, y, 1)$

Hence $S \setminus T$ is an $\mathcal{L}^T$-class.

T is not finitely generated, as all $(1, x, 1)$ are indecomposable.

Green rerewriting

Green rerewriting

$$h_i s = \tau(i, s) h_{\lambda(i,s)}$$

$$a_1 a_2 a_3 \ldots a_k = \tau(i_1, a_1)\tau(i_2, a_2) \ldots \tau(i_k, a_k) h_{i_{k+1}}$$

Green rerewriting

$$h_i s = \tau(i,s) h_{\lambda(i,s)}$$
$$a_1 a_2 a_3 \dots a_k = \tau(i_1, a_1)\tau(i_2, a_2)\dots\tau(i_k, a_k) h_{i_{k+1}}$$
$$s h_i = h_{\rho(s,i)} \sigma(s,i)$$

Green rerewriting

$$h_i s = \tau(i,s) h_{\lambda(i,s)}$$

$$a_1 a_2 a_3 \ldots a_k = \tau(i_1,a_1)\tau(i_2,a_2)\ldots\tau(i_k,a_k)h_{i_{k+1}}$$

$$s h_i = h_{\rho(s,i)}\sigma(s,i)$$

$$\tau(i_1,a_1)\tau(i_2,a_2)\ldots\tau(i_k,a_k)h_{i_{k+1}}$$

Green rerewriting

$$h_i s = \tau(i,s) h_{\lambda(i,s)}$$
$$a_1 a_2 a_3 \dots a_k = \tau(i_1,a_1)\tau(i_2,a_2)\dots\tau(i_k,a_k) h_{i_{k+1}}$$
$$s h_i = h_{\rho(s,i)} \sigma(s,i)$$

$$\tau(i_1,a_1)\tau(i_2,a_2)\dots\tau(i_k,a_k)\ h_{j_k}$$

Green rerewriting

$$h_i s = \tau(i,s) h_{\lambda(i,s)}$$
$$a_1 a_2 a_3 \dots a_k = \tau(i_1, a_1)\tau(i_2, a_2) \dots \tau(i_k, a_k) h_{i_{k+1}}$$
$$s h_i = h_{\rho(s,i)} \sigma(s,i)$$

$$\begin{aligned} & \tau(i_1, a_1)\tau(i_2, a_2) \dots \tau(i_k, a_k)\ h_{j_k} \\ = \ & \tau(i_1, a_1) \dots \tau(i_{k-1}, a_{k-1})\, h_{\rho(\tau(i_k, a_k), j_k)}\, \sigma(\tau(i_k, a_k), j_k) \end{aligned}$$

Green rerewriting

$$h_i s = \tau(i,s) h_{\lambda(i,s)}$$
$$a_1 a_2 a_3 \dots a_k = \tau(i_1,a_1)\tau(i_2,a_2)\dots\tau(i_k,a_k) h_{i_{k+1}}$$
$$s h_i = h_{\rho(s,i)} \sigma(s,i)$$

$$\begin{aligned} & \tau(i_1,a_1)\tau(i_2,a_2)\dots\tau(i_k,a_k)\ h_{j_k} \\ = \ & \tau(i_1,a_1)\dots\tau(i_{k-1},a_{k-1}) \quad h_{j_{k-1}} \quad \sigma(\tau(i_k,a_k),j_k) \end{aligned}$$

Green rerewriting

$$h_i s = \tau(i,s) h_{\lambda(i,s)}$$

$$a_1 a_2 a_3 \dots a_k = \tau(i_1,a_1)\tau(i_2,a_2)\dots\tau(i_k,a_k) h_{i_{k+1}}$$

$$s h_i = h_{\rho(s,i)} \sigma(s,i)$$

$$\begin{aligned} & \tau(i_1,a_1)\tau(i_2,a_2)\dots\tau(i_k,a_k)\ h_{j_k} \\ = \ & \tau(i_1,a_1)\dots\tau(i_{k-1},a_{k-1}) \quad h_{j_{k-1}} \quad \sigma(\tau(i_k,a_k),j_k) \\ = \ & \dots \end{aligned}$$

Green rerewriting

$$h_i s = \tau(i,s) h_{\lambda(i,s)}$$
$$a_1 a_2 a_3 \dots a_k = \tau(i_1,a_1)\tau(i_2,a_2)\dots\tau(i_k,a_k) h_{i_{k+1}}$$
$$s h_i = h_{\rho(s,i)} \sigma(s,i)$$

$$\begin{aligned}
& \tau(i_1,a_1)\tau(i_2,a_2)\dots\tau(i_k,a_k)\ h_{j_k} \\
= \ & \tau(i_1,a_1)\dots\tau(i_{k-1},a_{k-1}) \quad h_{j_{k-1}} \quad \sigma(\tau(i_k,a_k),j_k) \\
= \ & \dots \\
= \ & h_{j_0}\sigma(\tau(i_1,a_1),j_1)\dots\sigma(\tau(i_{k-1},a_{k-1}),j_{k-1})\sigma(\tau(i_k,a_k),j_k)
\end{aligned}$$

Green rerewriting

$$h_i s = \tau(i,s) h_{\lambda(i,s)}$$
$$a_1 a_2 a_3 \dots a_k = \tau(i_1,a_1)\tau(i_2,a_2)\dots\tau(i_k,a_k) h_{i_{k+1}}$$
$$s h_i = h_{\rho(s,i)} \sigma(s,i)$$

$$\begin{aligned}
& \tau(i_1,a_1)\tau(i_2,a_2)\dots\tau(i_k,a_k)\ h_{j_k} \\
= & \ \tau(i_1,a_1)\dots\tau(i_{k-1},a_{k-1}) \quad h_{j_{k-1}} \quad \sigma(\tau(i_k,a_k),j_k) \\
= & \ \dots \\
= & \ h_{j_0}\sigma(\tau(i_1,a_1),j_1)\dots\sigma(\tau(i_{k-1},a_{k-1}),j_{k-1})\sigma(\tau(i_k,a_k),j_k)
\end{aligned}$$

Lemma

If $a_1 \dots a_k \in T$ then $h_{j_0} = 1$.

Generation theorems for Green index

Generation theorems for Green index

Theorem

If $S = \langle A \rangle$ and $\{h_i : i \in I\}$ are representatives of $\mathcal{H}^T$-classes in $S \setminus T$ then

$$T = \langle \{\sigma(\tau(i,a),j) : a \in A,\ i,j \in I\} \rangle.$$

Generation theorems for Green index

Theorem

If $S = \langle A \rangle$ *and* $\{h_i \,:\, i \in I\}$ *are representatives of* $\mathcal{H}^T$*-classes in* $S \setminus T$ *then*

$$T = \langle \{\sigma(\tau(i,a),j) \,:\, a \in A,\ i,j \in I\} \rangle.$$

Corollary

Suppose $[S:T]_G < \infty$*. Then* S *is finitely generated if and only if* T *is finitely generated.*

Generation theorems for Green index

Theorem

If $S = \langle A \rangle$ and $\{h_i : i \in I\}$ are representatives of $\mathcal{H}^T$-classes in $S \setminus T$ then

$$T = \langle \{\sigma(\tau(i,a),j) : a \in A,\ i,j \in I\} \rangle.$$

Corollary

Suppose $[S : T]_G < \infty$. Then S is finitely generated if and only if T is finitely generated.

Rewriting mapping:

$\phi(a_1 a_2 \dots a_k) = b_{i_1,a_1,j_1} b_{i_2,a_2,j_2} \dots b_{i_k,a_k,j_k}$

Generation theorems for Green index

Theorem

If $S = \langle A \rangle$ and $\{h_i \,:\, i \in I\}$ are representatives of $\mathcal{H}^T$-classes in $S \setminus T$ then

$$T = \langle \{\sigma(\tau(i,a),j) \,:\, a \in A,\ i,j \in I\} \rangle.$$

Corollary

Suppose $[S : T]_G < \infty$. Then S is finitely generated if and only if T is finitely generated.

Rewriting mapping:

$\phi(a_1 a_2 \dots a_k) = b_{i_1,a_1,j_1} b_{i_2,a_2,j_2} \dots b_{i_k,a_k,j_k}$

$+$ recursive formulas for i_l, j_l.

Do we need $\mathcal{H}^T$-classes?

Do we need $\mathcal{H}^T$-classes?

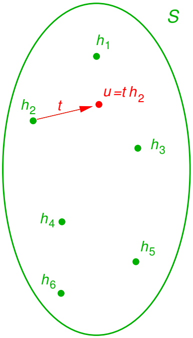
Or is sufficient to require:

$$\forall s \in S \ : \ \exists i \in I \ : \ \exists t, t' \in T \ : \ s = th_i = h_i t'?$$

Do we need $\mathcal{H}^T$-classes?

Or is sufficient to require:

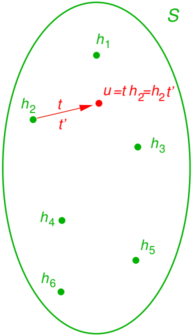
$$\forall s \in S \ : \ \exists i \in I \ : \ \exists t, t' \in T \ : \ s = th_i = h_i t'?$$



Do we need $\mathcal{H}^T$-classes?

Or is sufficient to require:

$$\forall s \in S \ : \ \exists i \in I \ : \ \exists t, t' \in T \ : \ s = th_i = h_i t'?$$



Do we need $\mathcal{H}^T$-classes?

Do we need $\mathcal{H}^T$-classes?

Example

M – a f.g. monoid

Do we need $\mathcal{H}^T$-classes?

Example

M – a f.g. monoid
J – a non f.g. ideal

Do we need $\mathcal{H}^T$-classes?

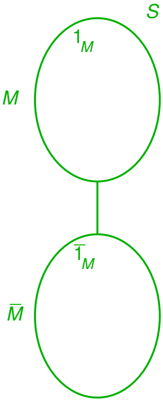
Example

M – a f.g. monoid
J – a non f.g. ideal
$\overline{M} = \{\overline{m} : m \in M\}$

Do we need $\mathcal{H}^T$-classes?

Example

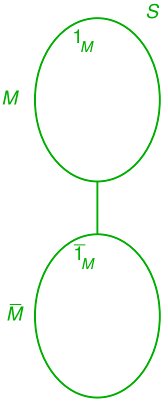
M – a f.g. monoid
J – a non f.g. ideal
$\overline{M} = \{\overline{m} : m \in M\}$
$S = M \dot\cup \overline{M}$



Do we need $\mathcal{H}^T$-classes?

Example

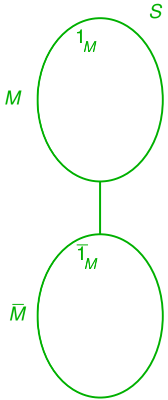
M – a f.g. monoid
J – a non f.g. ideal
$\overline{M} = \{\overline{m} \ : \ m \in M\}$
$S = M \dot\cup \overline{M}$
$m\overline{n} = \overline{mn} = \overline{m}n$



Do we need $\mathcal{H}^T$-classes?

Example

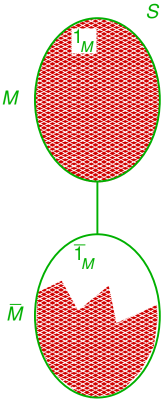
M – a f.g. monoid
J – a non f.g. ideal
$\overline{M} = \{\overline{m} \,:\, m \in M\}$
$S = M \dot\cup \overline{M}$
$m\overline{n} = \overline{mn} = \overline{m}n$
$S = \langle M \cup \{\overline{1_M}\}\rangle$



Do we need $\mathcal{H}^T$-classes?

Example

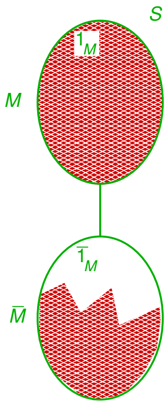
M – a f.g. monoid
J – a non f.g. ideal
$\overline{M} = \{\overline{m} : m \in M\}$
$S = M \dot\cup \overline{M}$
$m\overline{n} = \overline{mn} = \overline{m}n$
$S = \langle M \cup \{\overline{1_M}\}\rangle$
$T = M \cup \overline{J}$



Do we need $\mathcal{H}^T$-classes?

Example

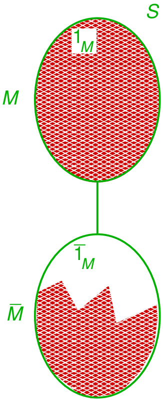
M – a f.g. monoid
J – a non f.g. ideal
$\overline{M} = \{\overline{m} \,:\, m \in M\}$
$S = M \dot\cup \overline{M}$
$m\overline{n} = \overline{mn} = \overline{m}n$
$S = \langle M \cup \{\overline{1_M}\} \rangle$
$T = M \cup \overline{J}$
T not f.g.



Do we need $\mathcal{H}^T$-classes?

Example

M – a f.g. monoid
J – a non f.g. ideal
$\overline{M} = \{\overline{m} \ : \ m \in M\}$
$S = M \dot\cup \overline{M}$
$m\overline{n} = \overline{mn} = \overline{m}n$
$S = \langle M \cup \{\overline{1_M}\}\rangle$
$T = M \cup \overline{J}$
T not f.g.
But: $\overline{m} = \overline{1_M} m = m\overline{1_M}$



What are rewriting mappings for?

What are rewriting mappings for?

Typically, they are involved in statements and proofs regarding properties and conditions related to generators.

For example. . .

What are rewriting mappings for?

Typically, they are involved in statements and proofs regarding properties and conditions related to generators.

For example...

Theorem

Suppose S is defined by a presentation $\langle A \mid R\rangle$, and that $T = \langle B\rangle \leq S$. Furthermore, suppose $\phi : \mathcal{L}(A, T) \to B^*$ is a rewriting mapping. Then the presentation

$$\begin{aligned} \langle B \quad \mid \quad & \phi(\beta) = b, \\ & \phi(w_1 w_2) = \phi(w_1)\phi(w_2), \\ & \phi(w_3 u w_4) = \phi(w_3 v w_4) \\ & (b \in B,\ w_1, w_2, w_3 u w_4 \in \mathcal{L}(A, T),\ (u, v) \in R)\rangle \end{aligned}$$

defines T.

Reidemeister–Schreier type theorems

Reidemeister–Schreier type theorems

Theorem

Suppose $[G : H] < \infty$. Then G is finitely presented if and only if H is finitely presented.

Reidemeister–Schreier type theorems

Theorem

Suppose $[G : H] < \infty$. Then G is finitely presented if and only if H is finitely presented.

Proof

Recall: $\phi(i, aw) = \tau(i, a)\phi(\lambda(i, a), w)$, $\phi(w) = \phi(1, w)$.

$\phi(w_1 w_2) = \phi(w_1)\phi(w_2)$

$\phi(w_3 u w_4) = \phi(w_3)\phi(\lambda(1, w_3), u)\phi(\lambda(1, w_3 u), w_4)$

Reidemeister–Schreier type theorems

Theorem

Suppose $[G : H] < \infty$. Then G is finitely presented if and only if H is finitely presented.

Proof

Recall: $\phi(i, aw) = \tau(i, a)\phi(\lambda(i, a), w)$, $\phi(w) = \phi(1, w)$.

$\phi(w_1 w_2) = \phi(w_1)\phi(w_2)$

$\phi(w_3 u w_4) = \phi(w_3)\phi(\lambda(1, w_3), u)\phi(\lambda(1, w_3 u), w_4)$

Theorem

Suppose $|S \setminus T| < \infty$. Then S is finitely presented if and only if T is finitely presented.

Proof

Similar

Reidemeister–Schreier type theorems

Theorem

Suppose $[G : H] < \infty$. Then G is finitely presented if and only if H is finitely presented.

Proof

Recall: $\phi(i, aw) = \tau(i, a)\phi(\lambda(i, a), w)$, $\phi(w) = \phi(1, w)$.

$\phi(w_1 w_2) = \phi(w_1)\phi(w_2)$

$\phi(w_3 u w_4) = \phi(w_3)\phi(\lambda(1, w_3), u)\phi(\lambda(1, w_3 u), w_4)$

Theorem

Suppose $|S \setminus T| < \infty$. Then S is finitely presented if and only if T is finitely presented.

Proof

Similar but much harder.

Reidemeister–Schreier–Green?

Reidemeister–Schreier–Green?

And if it [the solution] was perhaps less sound than I had thought in the first flush of discovery, its inelegance never diminished. And it was above all inelegant in this, to my mind... (S. Beckett, Molloy)

Reidemeister–Schreier–Green?

And if it [the solution] was perhaps less sound than I had thought in the first flush of discovery, its inelegance never diminished. And it was above all inelegant in this, to my mind... (S. Beckett, Molloy)

Problem

If S is finitely presented and $[S : T]_G < \infty$ is T finitely presented?

Reidemeister–Schreier–Green?

And if it [the solution] was perhaps less sound than I had thought in the first flush of discovery, its inelegance never diminished. And it was above all inelegant in this, to my mind... (S. Beckett, Molloy)

Problem

If S is finitely presented and $[S : T]_G < \infty$ is T finitely presented?

What is at issue:

Reidemeister–Schreier–Green?

And if it [the solution] was perhaps less sound than I had thought in the first flush of discovery, its inelegance never diminished. And it was above all inelegant in this, to my mind... (S. Beckett, Molloy)

Problem

If S is finitely presented and $[S : T]_G < \infty$ is T finitely presented?

What is at issue: The way ϕ is defined (re-re-writing).

Reidemeister–Schreier–Green?

And if it [the solution] was perhaps less sound than I had thought in the first flush of discovery, its inelegance never diminished. And it was above all inelegant in this, to my mind... (S. Beckett, Molloy)

Problem

If S is finitely presented and $[S:T]_G < \infty$ is T finitely presented?

What is at issue: The way ϕ is defined (re-re-writing).

Theorem

Suppose $[S:T]_G < \infty$. If T and all the relative Schützenberger groups are finitely presented then S is finitely presented.

But: Reidemeister–Schreier–Malcev–Green

But: Reidemeister–Schreier–Malcev–Green

Definition

Suppose S is embeddable into a group. A presentation $\langle A \mid R\rangle$ is a Malcev presentation for S if $S \cong A^*/\rho$ where ρ is the smallest congruence on A^* such that A^*/ρ is embeddable into a group.

But: Reidemeister–Schreier–Malcev–Green

Definition

Suppose S is embeddable into a group. A presentation $\langle A \mid R\rangle$ is a Malcev presentation for S if $S \cong A^*/\rho$ where ρ is the smallest congruence on A^* such that A^*/ρ is embeddable into a group.

Theorem

Suppose $[S : T]_G < \infty$. Then S has a finite Malcev presentation if and only if T has a finite Malcev presentation.

Automatic structures and rewriting

Obervation

All the rewriting mappings we have encountered so far can be realised by transducers.

Automatic structures and rewriting

Obervation

All the rewriting mappings we have encountered so far can be realised by transducers.

Theorem

If $[S : T]_G < \infty$ *and* S *is automatic then* T *is automatic.*

Automatic structures and rewriting

Obervation

All the rewriting mappings we have encountered so far can be realised by transducers.

Theorem

If $[S : T]_G < \infty$ *and* S *is automatic then* T *is automatic.*

Remark

The converse holds for groups and finite Rees index, but not in general.

Syntactic congruences

ρ – an equivalence on a semigroup S

$\Sigma(\rho)$ – the largest congruence on S contained in ρ

$\Sigma(T)$ (for $T \leq S$) $= \Sigma((T \times T) \cup ((S \setminus T) \times (S \setminus T)))$

Syntactic congruences

ρ – an equivalence on a semigroup S

$\Sigma(\rho)$ – the largest congruence on S contained in ρ

$\Sigma(T)$ (for $T \leq S$) $= \Sigma((T \times T) \cup ((S \setminus T) \times (S \setminus T)))$

Example

If G is a group and $H \leq G$ then $\Sigma(H)$ is the congruence determined by the kernel of H.

Syntactic congruences

ρ – an equivalence on a semigroup S

$\Sigma(\rho)$ – the largest congruence on S contained in ρ

$\Sigma(T)$ (for $T \leq S$) $= \Sigma((T \times T) \cup ((S \setminus T) \times (S \setminus T)))$

Example

If G is a group and $H \leq G$ then $\Sigma(H)$ is the congruence determined by the kernel of H.

Example

If S is a semigroup and J is an ideal, then $\Sigma(J)$ is the Rees congruence $(J \times J) \cup \Delta_S$.

Syntactic congruences and Green index

Theorem

If $[S : T]_G < \infty$ *then* $\Sigma(T)$ *has finitely many classes.*

Syntactic congruences and Green index

Theorem

If $[S : T]_G < \infty$ *then* $\Sigma(T)$ *has finitely many classes.*

Related to this:

Theorem

Suppose $[S : T]_G < \infty$. *Then* S *is residually finite if and only if* T *is residually finite.*

Syntactic congruences and Green index

Theorem

If $[S : T]_G < \infty$ *then* $\Sigma(T)$ *has finitely many classes.*

Related to this:

Theorem

Suppose $[S : T]_G < \infty$*. Then* S *is residually finite if and only if* T *is residually finite.*

Corollary

Let S *be finitely generated, and let* $m \in \mathbb{N}$*. Then there exist only finitely many subsemigroups* $T \leq S$ *such that* $[S : T]_G \leq m$*.*

Syntactic congruences and Green index

Theorem

If $[S : T]_G < \infty$ *then* $\Sigma(T)$ *has finitely many classes.*

Related to this:

Theorem

Suppose $[S : T]_G < \infty$*. Then* S *is residually finite if and only if* T *is residually finite.*

Corollary

Let S *be finitely generated, and let* $m \in \mathbb{N}$*. Then there exist only finitely many subsemigroups* $T \leq S$ *such that* $[S : T]_G \leq m$*.*

Proof

There are only finitely many actions of S on a finite set.

A surprising undecidability result (after Jura 1980)

A surprising undecidability result (after Jura 1980)

Theorem

There is no algorithm which has as its input a finite presentation $\langle A \mid R \rangle$ defining a semigroup S and a number $n \in \mathbb{N}$ and which outputs generating sets of all subsemigroups of S of Green index $\leq n$.

A surprising undecidability result (after Jura 1980)

Theorem

There is no algorithm which has as its input a finite presentation $\langle A \mid R \rangle$ defining a semigroup S and a number $n \in \mathbb{N}$ and which outputs generating sets of all subsemigroups of S of Green index $\leq n$.

Remark

Such an algorithm exists for groups, and has practical implementations.

A surprising undecidability result (after Jura 1980)

Theorem

There is no algorithm which has as its input a finite presentation $\langle A \mid R \rangle$ defining a semigroup S and a number $n \in \mathbb{N}$ and which outputs generating sets of all subsemigroups of S of Green index $\leq n$.

Remark

Such an algorithm exists for groups, and has practical implementations.

Proof

Suppose otherwise.

A surprising undecidability result (after Jura 1980)

Theorem

There is no algorithm which has as its input a finite presentation $\langle A \mid R \rangle$ defining a semigroup S and a number $n \in \mathbb{N}$ and which outputs generating sets of all subsemigroups of S of Green index $\leq n$.

Remark

Such an algorithm exists for groups, and has practical implementations.

Proof

Suppose otherwise.

$\{0\}$ has Green index $|S|$ in S^0.

A surprising undecidability result (after Jura 1980)

Theorem

There is no algorithm which has as its input a finite presentation $\langle A \mid R \rangle$ defining a semigroup S and a number $n \in \mathbb{N}$ and which outputs generating sets of all subsemigroups of S of Green index $\leq n$.

Remark

Such an algorithm exists for groups, and has practical implementations.

Proof

Suppose otherwise.

$\{0\}$ has Green index $|S|$ in S^0.

Run algorithm for $n = 1$, and see if $\{0\}$ is among the outputs.

A surprising undecidability result (after Jura 1980)

Theorem

There is no algorithm which has as its input a finite presentation $\langle A \mid R\rangle$ defining a semigroup S and a number $n \in \mathbb{N}$ and which outputs generating sets of all subsemigroups of S of Green index $\leq n$.

Remark

Such an algorithm exists for groups, and has practical implementations.

Proof

Suppose otherwise.

$\{0\}$ has Green index $|S|$ in S^0.

Run algorithm for $n = 1$, and see if $\{0\}$ is among the outputs.

This decides triviality of S, a contradiction.