Word Graphs

Tatiana Jajcayová
FMFI UK
jajcayova@fmph.uniba.sk

January 15, 2013

How do we present an algebraic structure?

Multiplication Table

Example from groups:

	e	a	b	c
e	e	a	b	c
a	a	e	c	b
b	b	c	e	a
c	c	b	a	e

Multiplication Table

Example from groups:

	e	a	b	c
e	e	a	b	c
a	a	e	c	b
b	b	c	e	a
c	c	b	a	e

$\Rightarrow$

e is the identity

Multiplication Table

Example from groups:

	e	a	b	c
e	e	a	b	c
a	a	e	c	b
b	b	c	e	a
c	c	b	a	e

$\Rightarrow$

e is the identity
$a^2 = b^2 = c^2 = e$

Multiplication Table

Example from groups:

	e	a	b	c
e	e	a	b	c
a	a	e	c	b
b	b	c	e	a
c	c	b	a	e

$\Rightarrow$

e is the identity
$a^2 = b^2 = c^2 = e$
$x \cdot y = y \cdot x$

Multiplication Table

Example from groups: Klein 4-Group

	e	a	b	c
e	e	a	b	c
a	a	e	c	b
b	b	c	e	a
c	c	b	a	e

$\Rightarrow$

e is the identity

$a^2 = b^2 = c^2 = e$

$x \cdot y = y \cdot x$

Presentations:

A presentation for an algebraic structure is "contracted" information about the structure allowing for a complete recovery of its original multiplication table.

Presentations:

A presentation for an algebraic structure is "contracted" information about the structure allowing for a complete recovery of its original multiplication table.

For example,
$V_4 = \langle e, a, b, c | e^2 = e, ea = a, \ldots, cb = a, c^2 = e \rangle$

Presentations:

A presentation for an algebraic structure is "contracted" information about the structure allowing for a complete recovery of its original multiplication table.

For example,
$V_4 = \langle e, a, b, c | e^2 = e, ea = a, \ldots, cb = a, c^2 = e \rangle$
Since e is the identity $\Rightarrow$

$$V_4 = \langle a, b, c | a^2 = b^2 = c^2 = e, ab = c, ac = b, ba = c, \\ bc = a, ca = b, cb = a \rangle$$

Presentations:

A presentation for an algebraic structure is "contracted" information about the structure allowing for a complete recovery of its original multiplication table.

For example,

$V_4 = \langle e, a, b, c | e^2 = e, ea = a, \ldots, cb = a, c^2 = e \rangle$

Since e is the identity $\Rightarrow$

$$V_4 = \langle a, b, c | a^2 = b^2 = c^2 = e, ab = c, ac = b, ba = c, bc = a, ca = b, cb = a \rangle$$

In addition, $c = ab \Rightarrow$

$$V_4 = \langle a, b | a^2 = b^2 = e, ab = ba \rangle$$

Presentations:

A presentation for an algebraic structure is "contracted" information about the structure allowing for a complete recovery of its original multiplication table.

For example,

$V_4 = \langle e, a, b, c | e^2 = e, ea = a, \ldots, cb = a, c^2 = e \rangle$

Since e is the identity $\Rightarrow$

$$V_4 = \langle a, b, c | a^2 = b^2 = c^2 = e, ab = c, ac = b, ba = c, \\ bc = a, ca = b, cb = a \rangle$$

In addition, $c = ab \Rightarrow$

$$V_4 = \langle a, b | a^2 = b^2 = e, ab = ba \rangle$$

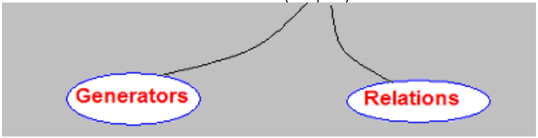
Every other multiplication in V_4 follows from this presentation and the fact that V_4 is a group.

$$V_4 = Gp\langle a, b | a^2 = b^2 = e, ab = ba \rangle$$

Presentations:

Formally, a presentation is a pair

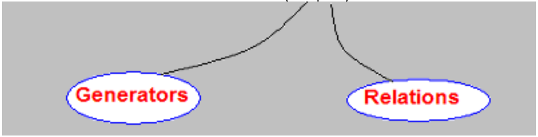
$$A = \langle X | R \rangle$$



with the relations R being equations between expressions formed of the generators from X.

Formally, a presentation is a pair

$$A = \langle X | R \rangle$$



with the relations R being equations between expressions formed of the generators from X.

$$A = Gp\langle X | R \rangle$$
$$A = Inv\langle X | R \rangle$$
$$A = InvM\langle X | R \rangle$$

Presentations

We say $S = Inv\langle X|R\rangle$ (or S is presented by $\langle X|R\rangle$) if

$$S = (X \cup X^{-1})^{+}/\tau$$

where τ is the smallest congruence containing the relation R and Vagner's relations:

$$\{(uu^{-1}u, u)|u \in S\} \cup \{(uu^{-1}vv^{-1}, vv^{-1}uu^{-1})|u, v \in S\}.$$

Properties of Presentations:

- presentations represent ”contracted information” about the structure

Properties of Presentations:

- presentations represent "contracted information" about the structure
- presentations define the algebraic structure uniquely

Properties of Presentations:

- presentations represent "contracted information" about the structure
- presentations define the algebraic structure uniquely
- a given algebraic structure can have more than one presentation

Turning the situation around:

Starting from a presentation, we would like to be able to answer *at least* some of the following questions:

Turning the situation around:

Starting from a presentation, we would like to be able to answer *at least* some of the following questions:

- how many elements has this structure?

Turning the situation around:

Starting from a presentation, we would like to be able to answer *at least* some of the following questions:

- how many elements has this structure?
- is it trivial?

Turning the situation around:

Starting from a presentation, we would like to be able to answer *at least* some of the following questions:

- how many elements has this structure?
- is it trivial?
- what are its substructures?

Turning the situation around:

Starting from a presentation, we would like to be able to answer *at least* some of the following questions:

- how many elements has this structure?
- is it trivial?
- what are its substructures?
- how is a presentation for a substructure related to the original presentation?

Turning the situation around:

Starting from a presentation, we would like to be able to answer *at least* some of the following questions:

- how many elements has this structure?
- is it trivial?
- what are its substructures?
- how is a presentation for a substructure related to the original presentation?
- given a product of generators, is this product equal to some other product of generators? "word problem"

Turning the situation around:

Starting from a presentation, we would like to be able to answer *at least* some of the following questions:

- how many elements has this structure?
- is it trivial?
- what are its substructures?
- how is a presentation for a substructure related to the original presentation?
- given a product of generators, is this product equal to some other product of generators? "word problem"
- given two presentations, do they define the same structure? "isomorphism problem"

Decision problems

A *decision problem* for a class of algebraic structures is a fixed specific question that can be asked for each member of the class and the answer for which is always either yes or no.

Decision problems

A *decision problem* for a class of algebraic structures is a fixed specific question that can be asked for each member of the class and the answer for which is always either yes or no.

A decision problem for a class of algebraic structures is decidable if there exists an effective procedure/algorithm/computer program that will for each specific instance of the question (i.e., for each specific structure from the class) terminate and give the correct answer.

Decision problems

A *decision problem* for a class of algebraic structures is a fixed specific question that can be asked for each member of the class and the answer for which is always either yes or no.

A decision problem for a class of algebraic structures is decidable if there exists an effective procedure/algorithm/computer program that will for each specific instance of the question (i.e., for each specific structure from the class) terminate and give the correct answer.

A decision problem is said to be undecidable, if no such algorithm exists.

Decision problems

A *decision problem* for a class of algebraic structures is a fixed specific question that can be asked for each member of the class and the answer for which is always either yes or no.

A decision problem for a class of algebraic structures is decidable if there exists an effective procedure/algorithm/computer program that will for each specific instance of the question (i.e., for each specific structure from the class) terminate and give the correct answer.

A decision problem is said to be undecidable, if no such algorithm exists.

Decision problems

All of the decision problems mentioned for the presentations are *undecidable* for the classes of groups and inverse semigroups.

Inverse semigroups

Inverse semigroups

While groups can be represented as symmetries:

Theorem (Cayley)

Every group can be embedded in the set of one to one transformations on a set.

Inverse semigroups

While groups can be represented as symmetries:

Theorem (Cayley)

Every group can be embedded in the set of one to one transformations on a set.

Inverse semigroups can be represented as partial symmetries:

Theorem (Vagner-Preston)

Every inverse semigroup can be embedded in the set of partial one to one transformations on a set.

Inverse Semigroup

Group -

- associative binary operation
- identity
- inverses: $a \cdot a^{-1} = a^{-1} \cdot a = e$

Inverse Semigroup

Group -

- associative binary operation
- identity
- inverses: $a \cdot a^{-1} = a^{-1} \cdot a = e$

Semigroup - associative operation

Inverse Semigroup

Group -

- associative binary operation
- identity
- inverses: $a \cdot a^{-1} = a^{-1} \cdot a = e$

Semigroup - associative operation

Inverse Semigroup

- associative binary operation
- generalized inverses:

$$a \cdot a^{-1} \cdot a = a$$

$$a^{-1} \cdot a \cdot a^{-1} = a^{-1}$$

Inverse Semigroup

Group -

- associative binary operation
- identity
- inverses: $a \cdot a^{-1} = a^{-1} \cdot a = e$

permutations, symmetries, bijections,...

Semigroup - associative operation

concatenation of strings

Inverse Semigroup

- associative binary operation
- existence of inverses:
  $a \cdot a^{-1} \cdot a = a$
  $a^{-1} \cdot a \cdot a^{-1} = a^{-1}$

strings, paths in graphs, transition semigroups, partial transformations, do/undo proceses

Presentations of Inverse Semigroups

We say $S = Inv\langle X|R\rangle$ (or S is presented by $\langle X|R\rangle$) if

$$S = (X \cup X^{-1})^{+}/\tau$$

where τ is the smallest congruence generated by the relation R and Vagner's relations:

$$\{(uu^{-1}u, u)|u \in S\} \cup \{(uu^{-1}vv^{-1}, vv^{-1}uu^{-1})|u, v \in S\}.$$

Given an inverse semigroup $S = Inv\langle X | R \rangle$, we will present results on both of the two basic types of questions related to presentations:

- Structural questions
- Decision problems

Structural Questions

One of the most basic structural question concerning inverse semigroups is the classification of the maximal subgroups of a given semigroup S.

Structural Questions

One of the most basic structural question concerning inverse semigroups is the classification of the maximal subgroups of a given semigroup S.

A maximal subgroup of an inverse semigroup $(S, \cdot)$ is a subset G of S "centered around" an idempotent e of S and satisfying the property that $(G, \cdot)$ is in fact a group with e serving as its identity element.

An element a is called an *idempotent* if it satisfies the property $a^2 = a$.

Maximal Subgroups:

An element a is called an *idempotent* if it satisfies the property $a^2 = a$.

For example,
if S is a group, it has only one idempotent - the identity.

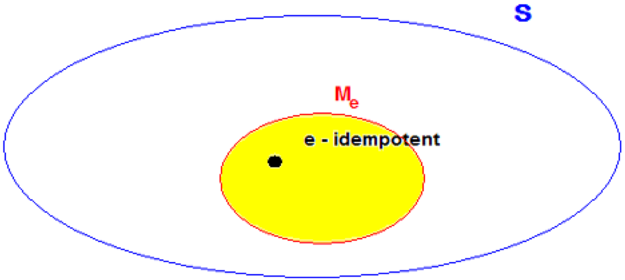
Maximal Subgroups:

An element a is called an *idempotent* if it satisfies the property $a^2 = a$.

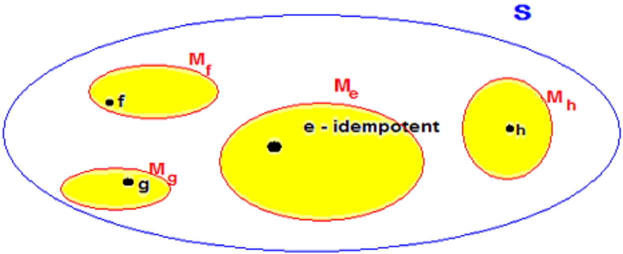
For example,
if S is a group, it has only one idempotent - the identity.

In general, inverse semigroups can have many idempotents.

Maximal Subgroups:



Maximal Subgroups:



Algorithmic problems in semigroups

- One of the most studied: the word problem for finitely presented (inverse) semigroups.

Algorithmic problems in semigroups

- One of the most studied: the word problem for finitely presented (inverse) semigroups.

Definition (Word problem)

Let $S = Inv\langle X | R \rangle$ and let $w, w' \in (X \cup X^{-1})^+$ be two words. Is there an algorithm (eq. is it decidable) w, w' represent the same element in S, (i.e. $w\tau = w'\tau$)?

Algorithmic problems in semigroups

- One of the most studied: the word problem for finitely presented (inverse) semigroups.
- It is a particular case of a more general problem in the framework of rewriting systems.

Definition (Word problem)

Let $S = Inv\langle X|R\rangle$ and let $w, w' \in (X \cup X^{-1})^+$ be two words. Is there an algorithm (eq. is it decidable) w, w' represent the same element in S, (i.e. $w\tau = w'\tau$)?

Definition (Rewriting system)

$\langle X|R\rangle$ where X is a finite alphabet, $R \subseteq X^* \times X^*$ which is symmetric. We say $w_1 \to w_2$ if $w_1 = uxv, w_2 = uyv$ and $(x, y) \in R$, the transitive closure of such relation is denoted by $\xrightarrow{*}$, thus the word problem is reduced to ask whether or not, given $w, w' \in X^*$, $w \xrightarrow{*} w'$

Example: the free case in Groups

Consider the free group $FG(a, b) = Gp\langle a, b|\emptyset\rangle$,

$$w = aba^{-1}ab^{-1}a \qquad\qquad w' = aa$$

Is $w = w'$ in $FG(a, b)$?

Example: the free case in Groups

Consider the free group $FG(a, b) = Gp\langle a, b|\emptyset\rangle$,

$$w = aba^{-1}ab^{-1}a \qquad\qquad w' = aa$$

Is $w = w'$ in $FG(a, b)$?

Seen as a rewriting system:

$$aba^{-1}ab^{-1}a \to abb^{-1}a \to aa$$

so it is always decidable since relations R reduce the length and we have always a normal form...

Example for a Free Inverse Semigroup

Consider the free inverse semigroup $FIS(a, b) = Inv\langle a, b | \emptyset \rangle$

$$w = aa^{-1}bb^{-1}ab \qquad w' = bb^{-1}ab$$

Is $w = w'$ in $FIS(a, b)$?

Example for a Free Inverse Semigroup

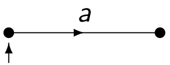
Let $S = FIS(a, b) = Inv\langle a, b \mid \emptyset \rangle$ and $w = aa^{-1}b^2b^{-1}a^2a^{-1}$.

Mann Tree MT(w):

Example for a Free Inverse Semigroup

Let $S = FIS(a, b) = Inv\langle a, b \mid \emptyset \rangle$ and $w = aa^{-1}b^2b^{-1}a^2a^{-1}$.

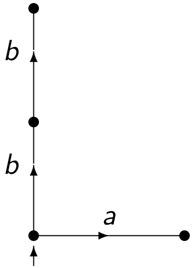
Mann Tree MT(w):



Example for a Free Inverse Semigroup

Let $S = FIS(a, b) = Inv\langle a, b \mid \emptyset \rangle$ and $w = aa^{-1}b^2b^{-1}a^2a^{-1}$.

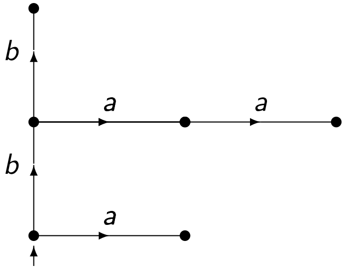
Mann Tree MT(w):



Example for a Free Inverse Semigroup

Let $S = FIS(a, b) = Inv\langle a, b \mid \emptyset \rangle$ and $w = aa^{-1}b^2b^{-1}a^2a^{-1}$.

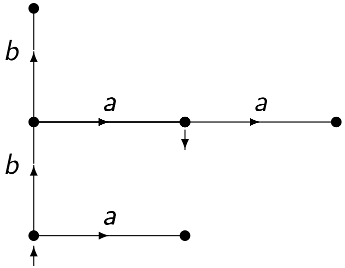
Mann Tree MT(w):



Example for a Free Inverse Semigroup

Let $S = FIS(a, b) = Inv\langle a, b \mid \emptyset \rangle$ and $w = aa^{-1}b^2b^{-1}a^2a^{-1}$.

Mann Tree MT(w):



Free Inverse Semigroup

Theorem (Munn,'74)

$$u = v \textit{ in } FIM(X)$$
$$\textit{iff}$$
$$MT(u) = MT(v) \textit{ and the roots are the same.}$$

Free Inverse Semigroup

Theorem (Munn,'74)

$$u = v \text{ in } FIM(X)$$
$$\text{iff}$$
$$MT(u) = MT(v) \text{ and the roots are the same.}$$

We build the Munn automata for w and w'.
If they recognize the same language, then $w\tau = w'\tau$.

Example for a Free Inverse Semigroup

Consider the free inverse semigroup $FIS(a, b) = Inv\langle a, b|\emptyset\rangle$

$$w = aa^{-1}bb^{-1}ab \qquad w' = bb^{-1}ab$$

Is $w = w'$ in $FIS(a, b)$?

Inverse Word Graph

Let $X \neq \emptyset$ be a set (an alphabet).
An inverse word graph over X is a connected graph whose edges are labeled by the elements from X, and that satisfies the property that for each edge z the oppositely oriented edge $\overline{z}$ is labeled by the inverse of the label of z.

Let $S = Inv\langle X|R\rangle$

Definition (Schützenberger graph)

Let w be a word in $(X \cup X^{-1})^+$. The Schützenberger graph of w relative to the presentation $Inv\langle X|R\rangle$ is the graph $S\Gamma(X, R, w\tau)$ whose vertices are the elements of the $\mathcal{R}$-class $\mathcal{R}_{w\tau}$ of $w\tau$ in S, and whose edges are of the form

$$\{(v_1, x, v_2) \mid v_1, v_2 \in \mathcal{R}_{w\tau} \text{ and } v_1(x\,\tau) = v_2\}.$$

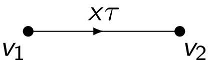
Schützenberger graph

Let $S = Inv\langle X|R\rangle$

Definition (Schützenberger graph)

Let w be a word in $(X \cup X^{-1})^{+}$. The Schützenberger graph of w relative to the presentation $Inv\langle X|R\rangle$ is the graph $S\Gamma(X, R, w\tau)$ whose vertices are the elements of the $\mathcal{R}$-class $\mathcal{R}_{w\tau}$ of $w\tau$ in S, and whose edges are of the form

$$\{(v_1, x, v_2) \mid v_1, v_2 \in \mathcal{R}_{w\tau} \text{ and } v_1(x\,\tau) = v_2\}.$$



if $v_2 = v_1 \cdot x\tau$.

Schützenberger automata

The Schützenberger automaton

$$\mathcal{A}(Y, T, w) = (ww^{-1}\tau, S\Gamma(Y, T, w), w\tau)$$

Schützenberger automata

- Schützenberger automata – tool to approach algorithmic and structural problems in inverse semigroups;

Schützenberger automata

- Schützenberger automata – tool to approach algorithmic and structural problems in inverse semigroups; generalization of Munn automata.

Schützenberger automata

- Schützenberger automata – tool to approach algorithmic and structural problems in inverse semigroups; generalization of Munn automata.
- $H = Inv\langle Y | T \rangle = (Y \cup Y^{-1})^+/\tau$ the Schützenberger graphs $S\Gamma(Y, T, w)$ for $w \in (Y \cup Y^{-1})^+$ are the connected components of the Cayley graph of H containing $w\tau$.

Schützenberger automata

- Schützenberger automata – tool to approach algorithmic and structural problems in inverse semigroups; generalization of Munn automata.
- $H = Inv\langle Y | T \rangle = (Y \cup Y^{-1})^{+}/\tau$ the Schützenberger graphs $S\Gamma(Y, T, w)$ for $w \in (Y \cup Y^{-1})^{+}$ are the connected components of the Cayley graph of H containing $w\tau$.
- $S\Gamma(Y, T, w)$ is a deterministic inverse word graph

Schützenberger automata - Stephen's theorem

Schützenberger automaton $\mathcal{A}(Y, T, w)$ has many nice properties...
[Stephen, 1990]

Schützenberger automata - Stephen's theorem

Schützenberger automaton $\mathcal{A}(Y, T, w)$ has many nice properties... [Stephen, 1990]

- one especially useful for the study of the word problem:

$$w\tau = w'\tau$$

$$\text{iff}$$

$$L[\mathcal{A}(Y, T, w)] = L[\mathcal{A}(Y, T, w')]$$

.

Schützenberger automata - Stephen's theorem

Schützenberger automaton $\mathcal{A}(Y, T, w)$ has many nice properties... [Stephen, 1990]

- one especially useful for the study of the word problem:

$$w\tau = w'\tau$$

$$\text{iff}$$

$$L[\mathcal{A}(Y, T, w)] = L[\mathcal{A}(Y, T, w')]$$

.

- one especially useful for the study of structure:

$$G_e \cong Aut(S\Gamma(X, R, e))$$

General Inverse Semigroups

Applying Stephen's theorem assumes that we already know the Schützenberger graph for the given word and inverse semigroup.

General Inverse Semigroups

Applying Stephen's theorem assumes that we already know the Schützenberger graph for the given word and inverse semigroup.

In general, we do not know any effective procedure for constructing the Schützenberger graphs.

Stephen's iterative procedure.

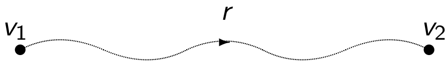
Elementary expansion:

- sewing on a relation $r = s$

Stephen's iterative procedure.

Elementary expansion:

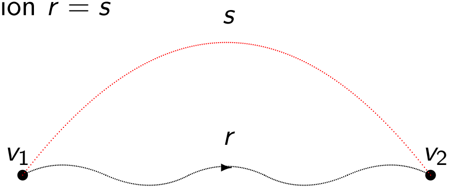
- sewing on a relation $r = s$



Stephen's iterative procedure.

Elementary expansion:

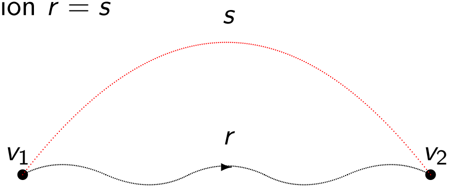
- sewing on a relation $r = s$



Stephen's iterative procedure.

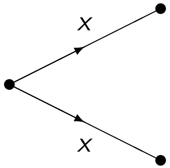
Elementary expansion:

- sewing on a relation $r = s$



Elementary determination:

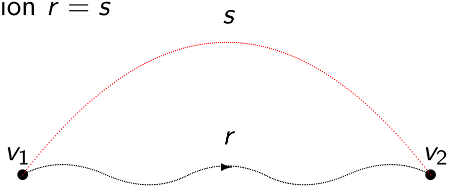
-edge folding



Stephen's iterative procedure.

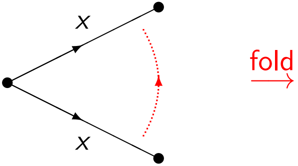
Elementary expansion:

- sewing on a relation $r = s$



Elementary determination:

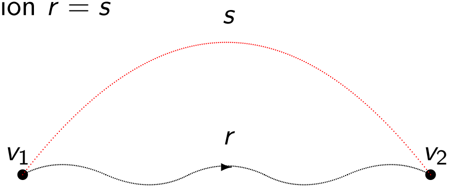
-edge folding



Stephen's iterative procedure.

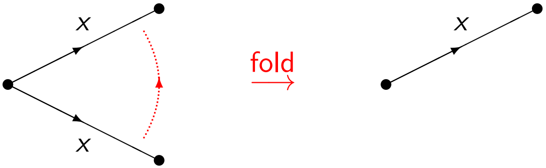
Elementary expansion:

- sewing on a relation $r = s$



Elementary determination:

-edge folding



Schützenberger automata

In this way we get a directed system of inverse automata

$$\mathcal{A}_1 \to \mathcal{A}_2 \to \ldots \to \mathcal{A}_i \to \ldots$$

whose directed limit is the Schützenberger automata $\mathcal{A}(Y, T, w)$.

Products

Combining “simple” objects into larger, more complicated, objects is one of the most fruitful approaches in mathematics.

Products

Combining "simple" objects into larger, more complicated, objects is one of the most fruitful approaches in mathematics.

For example, if we define the property of being simple to be the property of having a solvable word problem, one needs to address the question of which product operations preserve the property of being simple.

Products

Combining "simple" objects into larger, more complicated, objects is one of the most fruitful approaches in mathematics.

For example, if we define the property of being simple to be the property of having a solvable word problem, one needs to address the question of which product operations preserve the property of being simple.

There are many product operations used successfully for both groups and semigroups – direct product, free product, amalgamated product.

Products

Combining "simple" objects into larger, more complicated, objects is one of the most fruitful approaches in mathematics.

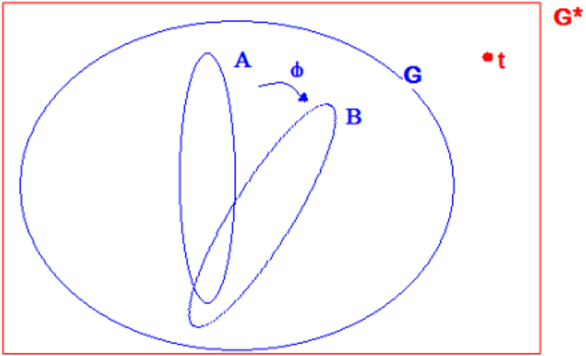
For example, if we define the property of being simple to be the property of having a solvable word problem, one needs to address the question of which product operations preserve the property of being simple.

There are many product operations used successfully for both groups and semigroups – direct product, free product, amalgamated product.

Our focus will be on the product operation originally introduced for groups and called an HNN-extension.

HNN-extensions for groups

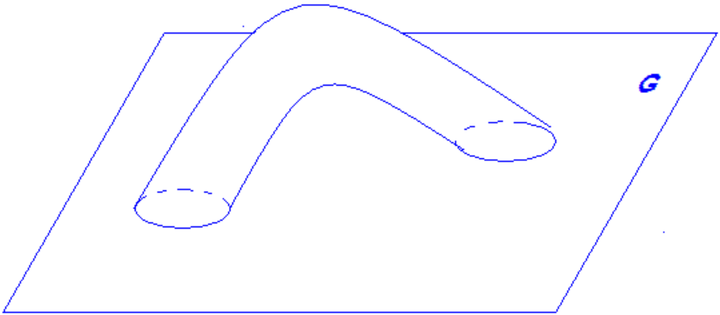
HigmanNeumannNeumann - extensions



$t^{-1}at = a\phi \quad \text{for} \quad \forall a \in A$

Handle

For example, the fundamental group of a surface with a handle is an HNN-extension of the fundamental group of the surface without the handle attached.



Definition of HNN-extensions for inverse semigroups

Definition (A.Yamamura)

Let $S = Inv\langle X \mid R\rangle$ be an inverse semigroup.
Let A, B be inverse subsemigroups of S,
$\varphi : A \longrightarrow B$ be an isomorphism

Then

$$S^* = Inv\langle S, t \mid t^{-1}at = a\varphi,$$

Definition (A.Yamamura)

Let $S = Inv\langle X \mid R\rangle$ be an inverse semigroup.
Let A, B be inverse subsemigroups of S,
$\varphi : A \longrightarrow B$ be an isomorphism

Then

$$S^* = Inv\langle S, t \mid t^{-1}at = a\varphi, t^{-1}t = f, tt^{-1} = e, \forall a \in A\rangle$$

is called the *HNN-extension* of S associated with φ.

Definition of HNN-extensions for inverse semigroups

Definition (A.Yamamura)

Let $S = Inv\langle X \mid R\rangle$ be an inverse semigroup.
Let A, B be inverse subsemigroups of S,
$\varphi : A \longrightarrow B$ be an isomorphism

$e \in A \subseteq eSe$ and $f \in B \subseteq fSf$ (or $e \notin A \subseteq eSe$ and $f \notin B \subseteq fSf$ for some $e, f \in E(S)$).
Then

$$S^* = Inv\langle S, t \mid t^{-1}at = a\varphi, t^{-1}t = f, tt^{-1} = e, \forall a \in A\rangle$$

is called the *HNN-extension* of S associated with φ.

Definition of HNN for inverse semigroups

Definition (A.Yamamura)

Let $S = Inv\langle X \mid R\rangle$ be an inverse semigroup.
Let A, B be inverse subsemigroups of S,
$\varphi : A \longrightarrow B$ be an isomorphism

$e \in A \subseteq eSe$ and $f \in B \subseteq fSf$ (or $e \notin A \subseteq eSe$ and $f \notin B \subseteq fSf$ for some $e, f \in E(S)$).

Then

$$S^* = Inv\langle X, t \mid R \cup R_{HNN}\rangle$$

is called the *HNN-extension* of S associated with φ.

Definition of HNN for inverse semigroups

Definition (A.Yamamura)

Let $S = Inv\langle X \mid R \rangle$ be an inverse semigroup.
Let A, B be inverse subsemigroups of S,
$\varphi : A \longrightarrow B$ be an isomorphism

$e \in A \subseteq eSe$ and $f \in B \subseteq fSf$ (or $e \notin A \subseteq eSe$ and $f \notin B \subseteq fSf$ for some $e, f \in E(S)$).

Then

$$S^* = Inv\langle X, t \mid R \cup R_{HNN} \rangle$$

is called the *HNN-extension* of S associated with φ.

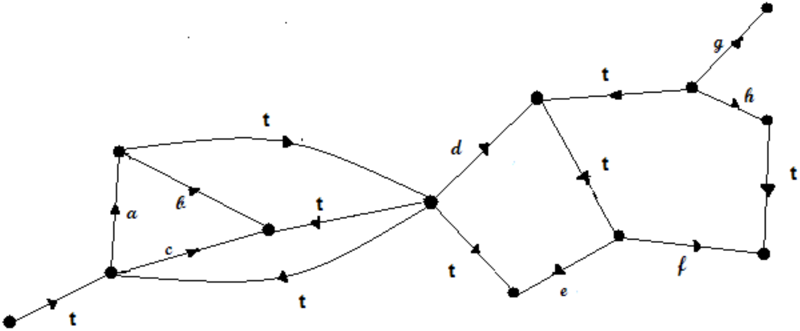
$$S \hookrightarrow S^*$$

HNN-extensions for inverse semigroups

In what follows, we shall address the structural and decision questions concerning the HNN-extensions of inverse semigroups, $S^* = Inv\langle X, t \mid R \cup R_{HNN} \rangle$, via the use of the very visual and intuitive concept of a graph "constructed from a word in X according to the rules in $R \cup R_{HNN}$".

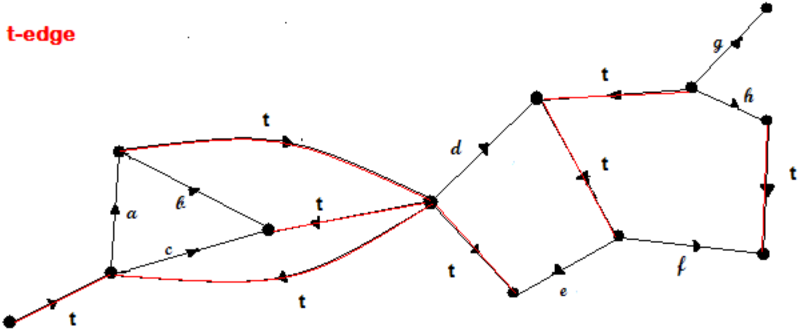
HNN-extensions for inverse semigroups

In the special case when $S = Inv\langle X, t | R \cup R_{HNN} \rangle$, a part of the word graph over $X \cup \{t\}$ may look something like this:



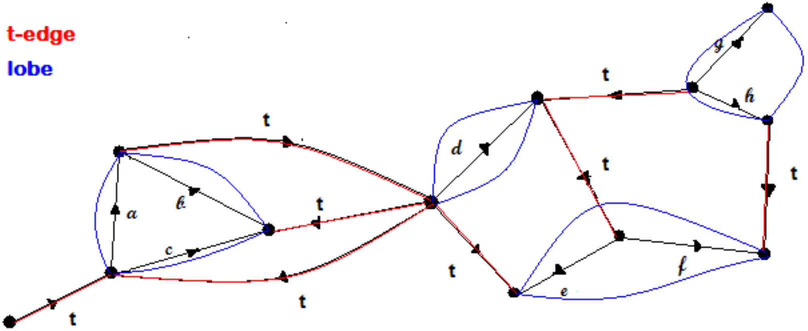
HNN-extensions for inverse semigroups

In the special case when $S = Inv\langle X, t | R \cup R_{HNN} \rangle$, a part of the word graph over $X \cup \{t\}$ may look something like this:



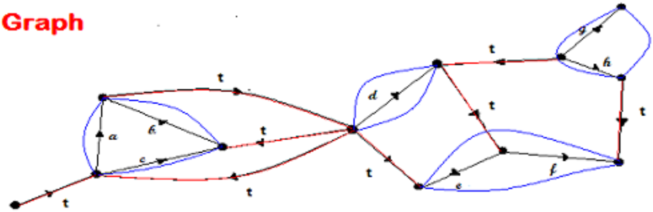
HNN-extensions for inverse semigroups

In the special case when $S = Inv\langle X, t | R \cup R_{HNN} \rangle$, a part of the word graph over $X \cup \{t\}$ may look something like this:



HNN-extensions for inverse semigroups

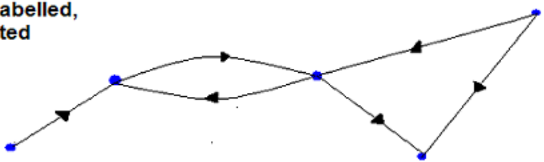
Graph



$$\Downarrow$$

Lobe Graph

non-labelled, oriented



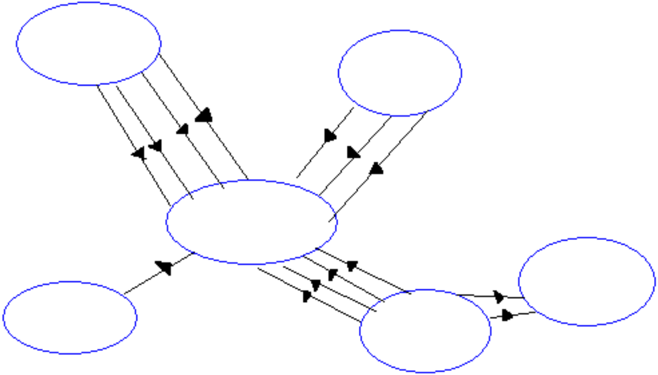
T(Γ)

The tree structure of lobe graphs

Theorem (T.J.)

The lobe graph $T(\Gamma)$ of a Schützenberger graph Γ relative to the presentation $Inv\langle X, t \mid R \cup R_{HNN} \rangle$ is an oriented tree.

The tree structure of lobe graphs



Characterization of the Schützenberger automata for HNN-extension.

Theorem (T.J.)

Let S^* be a lower bounded HNN-extension. The Schützenberger automata of S^* relative to the presentation $Inv\langle X \cup \{t\} | R \cup R_{HNN}\rangle$ are precisely the complete T-automata that possess a host.

Characterization of the Schützenberger automata for HNN-extension.

Theorem (T.J.)

Let S^* be a lower bounded HNN-extension. The Schützenberger automata of S^* relative to the presentation $Inv\langle X \cup \{t\} | R \cup R_{HNN} \rangle$ are precisely the complete T-automata that possess a host.

- Schützenberger graphs of HNN-extensions have tree like lobe structure and many other "nice" features – e.g., they contain a special subgraph with only finitely many lobes that contains the information for the whole graph.

Characterization of the Schützenberger automata for HNN-extension.

Theorem (T.J.)

Let S^* be a lower bounded HNN-extension. The Schützenberger automata of S^* relative to the presentation $Inv\langle X \cup \{t\} | R \cup R_{HNN}\rangle$ are precisely the complete T-automata that possess a host.

- Schützenberger graphs of HNN-extensions have tree like lobe structure and many other "nice" features – e.g., they contain a special subgraph with only finitely many lobes that contains the information for the whole graph.
- the tree like lobe structure of these graphs allows for the use of the Bass-Serre Theory of group actions on trees and graphs of groups.

Word Problem for HNN-extension.

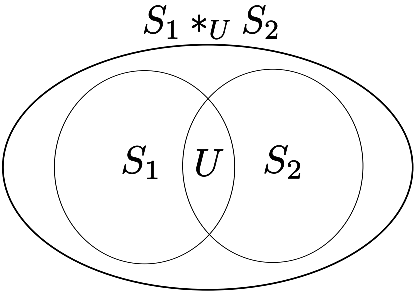
Theorem (T.J.)

The word problem is decidable for any HNN-extension of the form $S^* = [S; A, B; \varphi]$, where A and B are isomorphic finitely generated inverse subsemigroups of $FIS(X)$.

Amalgams of Inverse Semigroups

Amalgam is a 5-uple $[S_1, S_2; U, \omega_1, \omega_2]$ where S_1, S_2, U are inverse semigroups and $\omega_i : U \hookrightarrow S_i, i = 1, 2$.

Amalgams of Inverse Semigroups



If $S_1 = Inv\langle X_1 | R_1 \rangle$, $S_2 = Inv\langle X_2 | R_2 \rangle$ with $X_1 \cap X_2 = \emptyset$

$$S_1 *_U S_2 = Inv\langle X | R_1, R_2, R_w \rangle = Inv\langle X | R \rangle$$

where $X = X_1 \cup X_2$, $R_w = \{(\omega_1(u), \omega_2(u)) : u \in U\}$

Word problem for amalgams of (inverse)-semigroups

The word problem for amalgams of (inverse)-semigroups Given two (inverse)-semigroups S_1, S_2 which have decidable word problem and the embeddings $\omega_i : U \hookrightarrow S_i$ are computable, does $S_1 *_U S_2$ have decidable word problem?

Word problem for amalgams of (inverse)-semigroups

The word problem for amalgams of (inverse)-semigroups Given two (inverse)-semigroups S_1, S_2 which have decidable word problem and the embeddings $\omega_i : U \hookrightarrow S_i$ are computable, does $S_1 *_U S_2$ have decidable word problem?

- Proof based on an ordered way to build Schützenberger automata

Theorem (Cherubini, Meakin, Piochi)

The word problem in $S_1 *_U S_2$ where S_1, S_2 are finite inverse semigroups is decidable.

Word problem for amalgams of (inverse)-semigroups

The word problem for amalgams of (inverse)-semigroups Given two (inverse)-semigroups S_1, S_2 which have decidable word problem and the embeddings $\omega_i : U \hookrightarrow S_i$ are computable, does $S_1 *_U S_2$ have decidable word problem?

- Proof based on an ordered way to build Schützenberger automata
- Result in contrast with Sapir's results using Minsky machines.

Theorem (Sapir)

There are two finite semigroups for which the word problem in $S_1 *_U S_2$ is undecidable.

Word problem for amalgams of (inverse)-semigroups

The word problem for amalgams of (inverse)-semigroups Given two (inverse)-semigroups S_1, S_2 which have decidable word problem and the embeddings $\omega_i : U \hookrightarrow S_i$ are computable, does $S_1 *_U S_2$ have decidable word problem?

- Proof based on an ordered way to build Schützenberger automata
- Result in contrast with Sapir's results using Minsky machines.
- Group case is decidable.

Theorem

If S_1, S_2 are two groups which have decidable word problem and the embeddings $\omega_i : U \hookrightarrow S_i$ are computable, then $S_1 *_U S_2$ have decidable word problem.

Word problem for amalgams of (inverse)-semigroups

The word problem for amalgams of (inverse)-semigroups Given two (inverse)-semigroups S_1, S_2 which have decidable word problem and the embeddings $\omega_i : U \hookrightarrow S_i$ are computable, does $S_1 *_U S_2$ have decidable word problem?

- Proof based on an ordered way to build Schützenberger automata
- Result in contrast with Sapir's results using Minsky machines.
- Group case is decidable.
- What about inverse semigroups?

Theorem

If S_1, S_2 are two groups which have decidable word problem and the embeddings $\omega_i : U \hookrightarrow S_i$ are computable, then $S_1 *_U S_2$ have decidable word problem.

Word problem for amalgams of (inverse)-semigroups

The word problem for amalgams of (inverse)-semigroups Given two (inverse)-semigroups S_1, S_2 which have decidable word problem and the embeddings $\omega_i : U \hookrightarrow S_i$ are computable, does $S_1 *_U S_2$ have decidable word problem?

- Proof based on an ordered way to build Schützenberger automata
- Result in contrast with Sapir's results using Minsky machines.
- Group case is decidable.
- What about inverse semigroups?

Theorem (Rodaro, Silva)

The word problem for $S_1 *_U S_2$ of inverse semigroups may be undecidable even if we assume S_1 and S_2 to have finite $\mathcal{R}$-classes and ω_1, ω_2 to be computable functions.

Idea of the proof

- use Schützenberger automata to simulate the behavior of a two counter machine building a correspondence

  iterative construction $\longleftrightarrow$ computations of the machine

Idea of the proof

- use Schützenberger automata to simulate the behavior of a two counter machine building a correspondence

  iterative construction $\longleftrightarrow$ computations of the machine

- given a two-counter machine, S_1 simulates the first tape, S_2 simulates the second and U simulates the control unit.

Idea of the proof

- use Schützenberger automata to simulate the behavior of a two counter machine building a correspondence

  iterative construction $\longleftrightarrow$ computations of the machine

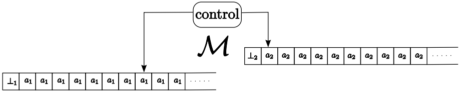
- given a two-counter machine, S_1 simulates the first tape, S_2 simulates the second and U simulates the control unit.

Idea of the proof

- use Schützenberger automata to simulate the behavior of a two counter machine building a correspondence

  iterative construction $\longleftrightarrow$ computations of the machine

- given a two-counter machine, S_1 simulates the first tape, S_2 simulates the second and U simulates the control unit.

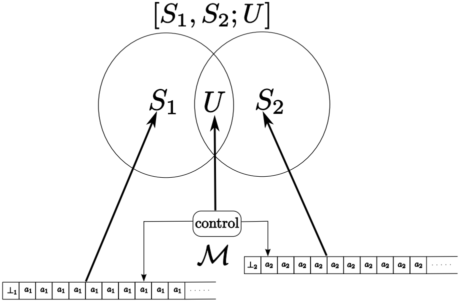


Idea of the proof

- use Schützenberger automata to simulate the behavior of a two counter machine building a correspondence

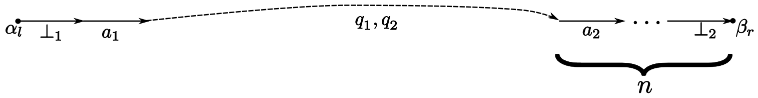
  iterative construction $\longleftrightarrow$ computations of the machine

- given a two-counter machine, S_1 simulates the first tape, S_2 simulates the second and U simulates the control unit.



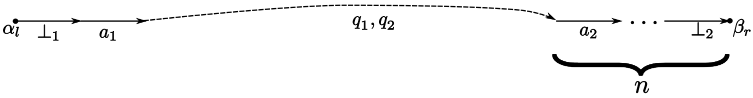
Idea of the proof

Starting from linear automaton of the word $\perp_1 a_1 q a_2^n \perp_2$ representing the configuration $(\mathcal{Q}, 1, n)$.



Idea of the proof

Since the machine is reversible there is a unique computation $(\mathcal{Q}, 1, n) \vdash_{\mathcal{M}} (\mathcal{Q}', 0, n)$ due to the instruction (for instance) $(\mathcal{Q}, 1, -, \mathcal{Q}')$

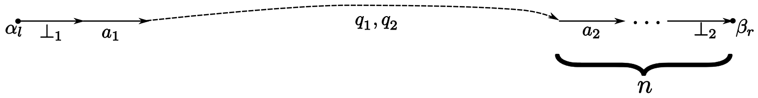


Idea of the proof

This corresponds to the relations

$$sa_1 q_1 = st_1 q_1' t_1^{-1}, sa_2 q_2 = st_2 q_2' t_2^{-1}$$

...so we apply an expansion

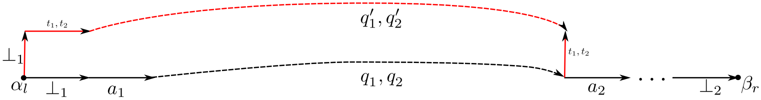


Idea of the proof

This corresponds to the relations

$$sa_1q_1 = st_1q_1't_1^{-1}, sa_2q_2 = st_2q_2't_2^{-1}$$

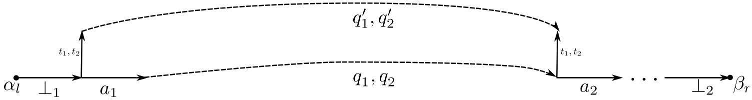
...so we apply an expansion



Idea of the proof

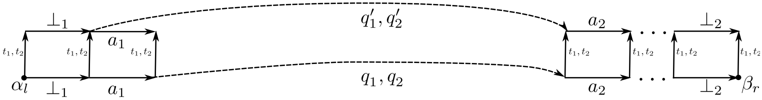
followed by folding...

∘



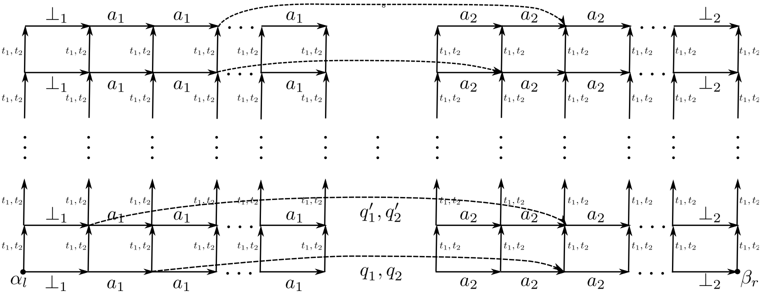
Idea of the proof

The extra relation in S_1, S_2 ensure the cloning of the configuration to the next step.



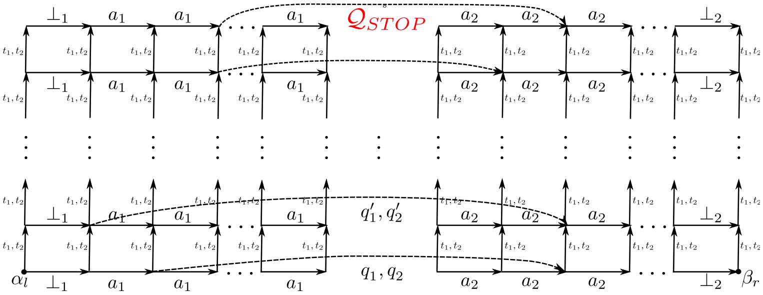
Idea of the proof

Continuing in this way we obtain a structure of this form...



Idea of the proof

If we reach the STOP instruction, some extra relations ensure that the final state is a zero...



Děkuji!