Generalisations of Small Cancellation Theory

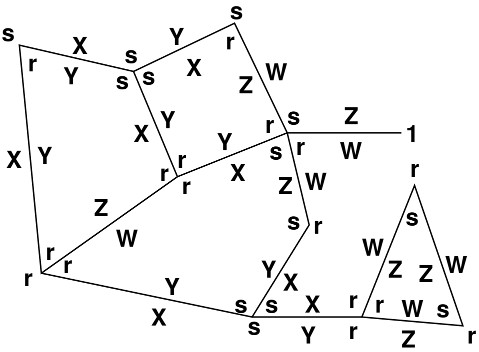
Max Neunhöffer

joint work with Jeffrey Burdges, Stephen Linton,

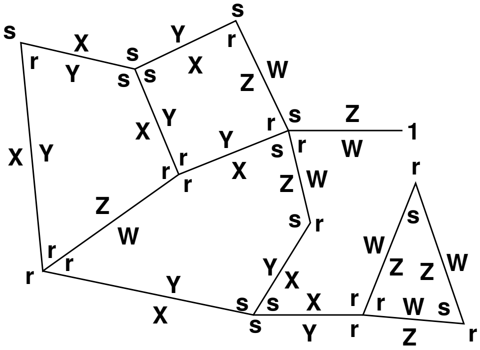
Richard Parker and Colva Roney-Dougal

NBSAN meeting St Andrews, 9 April 2013

We draw connected finite graphs in the plane and label them:

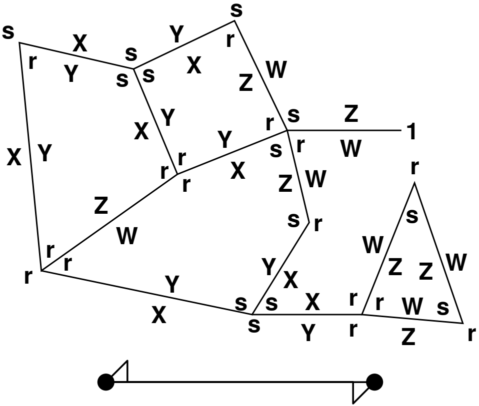


We draw connected finite graphs in the plane and label them:



Faces are oriented clockwise.

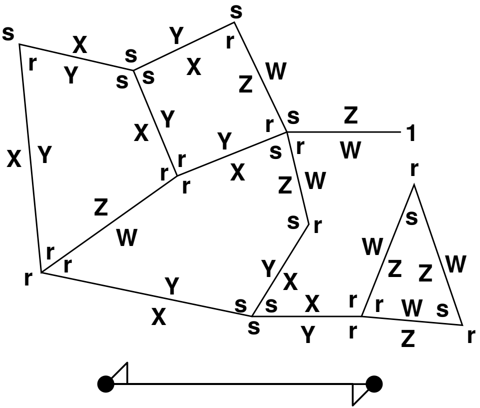
We draw connected finite graphs in the plane and label them:



Faces are oriented clockwise.

We view each edge as a pair of opposite directed edges: half-edges.

We draw connected finite graphs in the plane and label them:



Faces are oriented clockwise.

We view each edge as a pair of opposite directed edges: half-edges.

Each half-edge is labelled at the start vertex and along the half-edge.

The diagram boundary problem

Let R be a finite set of cyclic words, called relators.

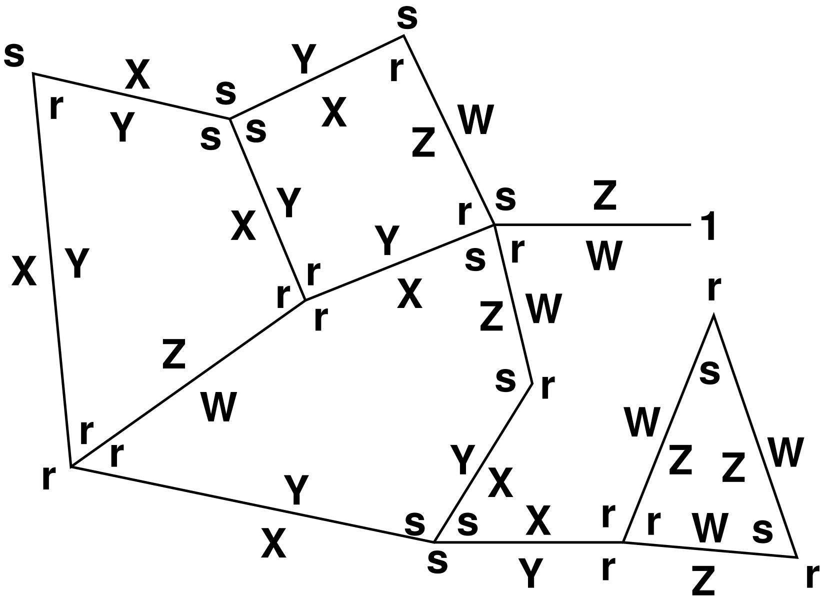
The diagram boundary problem

Let R be a finite set of cyclic words, called relators.

Problem (Diagram boundary problem)

Algorithmically devise a procedure that decides for any cyclic word w, whether or not there is a diagram such that

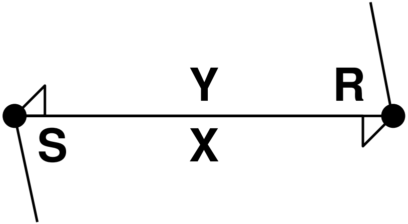
- *every internal region is labelled by a relator, and*
- *the external boundary is labelled by w.*


s
s Y X r
s X s r Y
Y s X Z W
s r Z
X Y Y r 1
X Y s r W
r r X
r r Z W
Z r
W s r
W
r W Z Z
r r Y X
Y
s s X r r W s
X s Y r Z r

Rules for the labels

We label every half-edge with two symbols,

- one for the corner to the right of where it starts, and
- one for the right hand side of it:



We now need rules for the corner labels and the edge labels.

Definition (Corner structures)

A corner structure is a set S with a subset $S_+ \subset S$, such that $S_0 := S \,\dot\cup\, \{0\}$ is a semigroup with 0 and:

$$\text{if } xy \in S_+ \text{ for } x, y \in S, \text{ then } yx \in S_+.$$

The elements in S_+ are called acceptors.

Definition (Corner structures)

A corner structure is a set S with a subset $S_+ \subset S$, such that $S_0 := S \mathbin{\dot\cup} \{0\}$ is a semigroup with 0 and:

$$\text{if } xy \in S_+ \text{ for } x, y \in S, \ \text{then } yx \in S_+.$$

The elements in S_+ are called acceptors.
Usually we will have: for all $x \in S$ there is a $y \in S$ with $xy \in S_+$.

Definition (Corner structures)

A corner structure is a set S with a subset $S_+ \subset S$, such that $S_0 := S \,\dot\cup\, \{0\}$ is a semigroup with 0 and:

$$\text{if } xy \in S_+ \text{ for } x, y \in S, \text{ then } yx \in S_+.$$

The elements in S_+ are called acceptors.
Usually we will have: for all $x \in S$ there is a $y \in S$ with $xy \in S_+$.

Lemma (Cyclicity)

Let S be a corner structure, if $s_1 s_2 \cdots s_k \in S_+$, then all rotations $s_i s_{i+1} \cdots s_k s_1 s_2 \cdots s_{i-1} \in S_+$.

Definition (Corner structures)

A corner structure is a set S with a subset $S_+ \subset S$, such that $S_0 := S \dot{\cup} \{0\}$ is a semigroup with 0 and:

$$\text{if } xy \in S_+ \text{ for } x, y \in S, \text{ then } yx \in S_+.$$

The elements in S_+ are called acceptors.
Usually we will have: for all $x \in S$ there is a $y \in S$ with $xy \in S_+$.

Lemma (Cyclicity)

Let S be a corner structure, if $s_1 s_2 \cdots s_k \in S_+$, then all rotations $s_i s_{i+1} \cdots s_k s_1 s_2 \cdots s_{i-1} \in S_+$.

Vertex rules

The corner labels are from a corner structure S,

Definition (Corner structures)

A corner structure is a set S with a subset $S_+ \subset S$, such that $S_0 := S \dot{\cup} \{0\}$ is a semigroup with 0 and:

$$\text{if } xy \in S_+ \text{ for } x, y \in S, \text{ then } yx \in S_+.$$

The elements in S_+ are called acceptors.
Usually we will have: for all $x \in S$ there is a $y \in S$ with $xy \in S_+$.

Lemma (Cyclicity)

Let S be a corner structure, if $s_1 s_2 \cdots s_k \in S_+$, then all rotations $s_i s_{i+1} \cdots s_k s_1 s_2 \cdots s_{i-1} \in S_+$.

Vertex rules

The corner labels are from a corner structure S, a vertex is valid, if the clockwise product of its corner labels is an acceptor.

Examples of corner structures

Examples of corner structures

- Let G be a group. Let $P := G$ and $P_+ := \{1\}$.

Examples of corner structures

- Let G be a group. Let $P := G$ and $P_+ := \{1\}$.
- Let $G_1, \ldots, G_k$ be groups. Let $Q := \dot{\bigcup} G_i$

Examples of corner structures

- Let G be a group. Let $P := G$ and $P_+ := \{1\}$.
- Let $G_1, \ldots, G_k$ be groups. Let $Q := \bigcup G_i$ and $Q_+ := \{1_{G_i} \mid 1 \leq i \leq k\}$.

Examples of corner structures

- Let G be a group. Let $P := G$ and $P_{\dot{+}} := \{1\}$.
- Let $G_1, \ldots, G_k$ be groups. Let $Q := \bigcup G_i$ and $Q_+ := \{1_{G_i} \mid 1 \leq i \leq k\}$. Elements of a single G_i multiply as before.

Examples of corner structures

- Let G be a group. Let $P := G$ and $P_{\dot{+}} := \{1\}$.
- Let $G_1, \ldots, G_k$ be groups. Let $Q := \dot{\bigcup} G_i$ and $Q_+ := \{1_{G_i} \mid 1 \leq i \leq k\}$. Elements of a single G_i multiply as before. Products across factors are all 0.

Examples of corner structures

- Let G be a group. Let $P := G$ and $P_+ := \{1\}$.
- Let $G_1, \ldots, G_k$ be groups. Let $Q := \dot{\bigcup} G_i$ and $Q_+ := \{1_{G_i} \mid 1 \leq i \leq k\}$. Elements of a single G_i multiply as before. Products across factors are all 0.
- Take any groupoid, undefined products are 0, identities accept.

Examples of corner structures

- Let G be a group. Let $P := G$ and $P_+ := \{1\}$.
- Let $G_1, \ldots, G_k$ be groups. Let $Q := \dot{\bigcup} G_i$ and $Q_+ := \{1_{G_i} \mid 1 \leq i \leq k\}$. Elements of a single G_i multiply as before. Products across factors are all 0.
- Take any groupoid, undefined products are 0, identities accept.
- $K_6 := \{s, t, e, b, r, l\}$,

Examples of corner structures

- Let G be a group. Let $P := G$ and $P_+ := \{1\}$.
- Let $G_1, \ldots, G_k$ be groups. Let $Q := \dot{\bigcup} G_i$ and $Q_+ := \{1_{G_i} \mid 1 \leq i \leq k\}$. Elements of a single G_i multiply as before. Products across factors are all 0.
- Take any groupoid, undefined products are 0, identities accept.
- $K_6 := \{s, t, e, b, r, l\}$, $K_{6+} := \{s, e\}$,

Examples of corner structures

- Let G be a group. Let $P := G$ and $P_+ := \{1\}$.
- Let $G_1, \ldots, G_k$ be groups. Let $Q := \dot{\bigcup} G_i$ and $Q_+ := \{1_{G_i} \mid 1 \leq i \leq k\}$. Elements of a single G_i multiply as before. Products across factors are all 0.
- Take any groupoid, undefined products are 0, identities accept.
- $K_6 := \{s, t, e, b, r, l\}$, $K_{6+} := \{s, e\}$,

	s	t	e	b	r	l
s	·	s	·	·	·	·
t	s	t	·	·	·	l
e	·	·	·	e	·	·
b	·	·	e	b	r	·
r	·	r	·	·	·	e
l	·	·	·	l	s	·

Note: $rl = e$ and $lr = s$, cyclicity, “inverses”, two idempotents.

Definition (Edge alphabet)

An edge alphabet is a set X with an involution $\bar{\ } : X \to X$.

Definition (Edge alphabet)

An edge alphabet is a set X with an involution $\overline{} : X \to X$.

(This is actually a special case of a corner structure.)

Definition (Edge alphabet)

An edge alphabet is a set X with an involution $\overline{} : X \to X$.

(This is actually a special case of a corner structure.)

Edge rules

The edge labels are from an edge alphabet, a pair of half-edges forming an edge with labels X and Y is valid, if $Y = \overline{X}$.

Definition (Edge alphabet)

An edge alphabet is a set X with an involution $\overline{} : X \to X$.

(This is actually a special case of a corner structure.)

Edge rules

The edge labels are from an edge alphabet, a pair of half-edges forming an edge with labels X and Y is valid, if $Y = \overline{X}$.

(For the experts:
This is a generalisation of the rules of van Kampen diagrams.)

Let S be a corner structure and X be an edge alphabet.

Definition (Set of relators)

A set of relators R is a finite set of cyclic alternating words in S and X.

Let S be a corner structure and X be an edge alphabet.

Definition (Set of relators)

A set of relators R is a finite set of cyclic alternating words in S and X.

Definition (Valid diagram)

Let R be a set of relators in S and X. A valid diagram is:
a finite plane graph with half-edge set $\hat{E}$ and a labelling function $\ell : \hat{E} \to S \times X, e \mapsto (\ell_S(e), \ell_X(e))$, such that

Let S be a corner structure and X be an edge alphabet.

Definition (Set of relators)

A set of relators R is a finite set of cyclic alternating words in S and X.

Definition (Valid diagram)

Let R be a set of relators in S and X. A valid diagram is:
a finite plane graph with half-edge set $\hat{E}$ and a labelling function $\ell : \hat{E} \to S \times X, e \mapsto (\ell_S(e), \ell_X(e))$, such that

- $\ell_S(e_1) \cdot \ell_S(e_2) \cdot \ell_S(e_3) \cdot \cdots \cdot \ell_S(e_k) \in S_+$ for every clockwise cyclic sequence $e_1, e_2, \ldots, e_k$ of half-edges leaving the same vertex,
- $\ell_X(e) = \overline{\ell_X(e')}$ for all edges $\{e, e'\}$ consisting of half-edges e, e',

Let S be a corner structure and X be an edge alphabet.

Definition (Set of relators)

A set of relators R is a finite set of cyclic alternating words in S and X.

Definition (Valid diagram)

Let R be a set of relators in S and X. A valid diagram is:
a finite plane graph with half-edge set $\hat{E}$ and a labelling function
$\ell : \hat{E} \to S \times X, e \mapsto (\ell_S(e), \ell_X(e))$, such that

- $\ell_S(e_1) \cdot \ell_S(e_2) \cdot \ell_S(e_3) \cdot \cdots \cdot \ell_S(e_k) \in S_+$ for every clockwise cyclic sequence $e_1, e_2, \ldots, e_k$ of half-edges leaving the same vertex,
- $\ell_X(e) = \overline{\ell_X(e')}$ for all edges $\{e, e'\}$ consisting of half-edges e, e',
- $(\ell_S(e_1), \ell_X(e_1), \ldots, \ell_S(e_k), \ell_X(e_k))^{\circlearrowleft} \in R$ for every clockwise cycle $(e_1, e_2, \ldots, e_k)^{\circlearrowleft}$ of half-edges around an internal face.

Let $\langle S; X \mid R \rangle$ be a presentation, that is:

- S is a corner structure,
- X is an edge alphabet and
- R is a set of relators in S and X.

Problem (Diagram boundary problem)

Algorithmically devise a procedure that decides for any cyclic alternating word w in S and X whether or not there is a valid diagram such that the external face is labelled by w.

Let $\langle S; X \mid R\rangle$ be a presentation, that is:

- S is a corner structure,
- X is an edge alphabet and
- R is a set of relators in S and X.

Problem (Diagram boundary problem)

Algorithmically devise a procedure that decides for any cyclic alternating word w in S and X whether or not there is a valid diagram such that the external face is labelled by w.

Problem (Isoperimetric inequality)

Algorithmically find and prove a function $\mathcal{D} : \mathbb{N} \to \mathbb{N}$, such that for every cyclic alternating word w in S and X of length $2k$ that is the boundary label of a valid diagram,

Let $\langle S; X \mid R \rangle$ be a presentation, that is:

- S is a corner structure,
- X is an edge alphabet and
- R is a set of relators in S and X.

Problem (Diagram boundary problem)

Algorithmically devise a procedure that decides for any cyclic alternating word w in S and X whether or not there is a valid diagram such that the external face is labelled by w.

Problem (Isoperimetric inequality)

Algorithmically find and prove a function $\mathcal{D} : \mathbb{N} \to \mathbb{N}$, such that for every cyclic alternating word w in S and X of length $2k$ that is the boundary label of a valid diagram, there is one with at most $\mathcal{D}(k)$ internal faces.

Let $\langle S; X \mid R \rangle$ be a presentation, that is:

- S is a corner structure,
- X is an edge alphabet and
- R is a set of relators in S and X.

Problem (Diagram boundary problem)

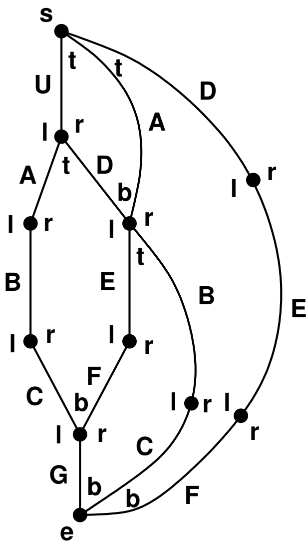
Algorithmically devise a procedure that decides for any cyclic alternating word w in S and X whether or not there is a valid diagram such that the external face is labelled by w.

Problem (Isoperimetric inequality)

Algorithmically find and prove a function $\mathcal{D} : \mathbb{N} \to \mathbb{N}$, such that for every cyclic alternating word w in S and X of length $2k$ that is the boundary label of a valid diagram, there is one with at most $\mathcal{D}(k)$ internal faces.

If there is a linear $\mathcal{D}$, we call $\langle S; X \mid R \rangle$ hyperbolic.

With K_6 we can do rewrite systems, if no rewrite has an empty side:



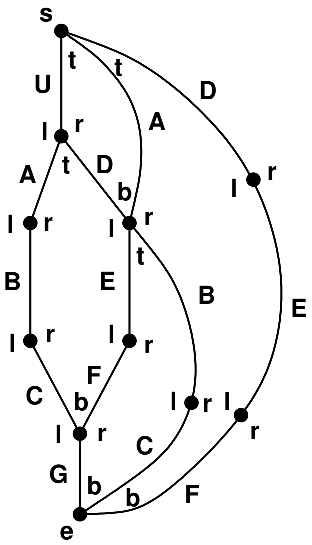
$$S = \begin{array}{c||c|c|c|c|c|c} & s & t & e & b & r & l \\ \hline s & \cdot & s & \cdot & \cdot & \cdot & \cdot \\ t & s & t & \cdot & \cdot & \cdot & l \\ e & \cdot & \cdot & \cdot & e & \cdot & \cdot \\ b & \cdot & \cdot & e & b & r & \cdot \\ r & \cdot & r & \cdot & \cdot & \cdot & e \\ l & \cdot & \cdot & \cdot & l & s & \cdot \end{array}, \quad S_+ = \{s, e\}$$

$X = \{A, B, C, D, E, F, G, U\}$ ($\bar{\ }$ is id_X)

This encodes $UABCG \to DEF$ using:

$$\{ABC \to DE, UD \to A, EFG \to BC\}$$

With K_6 we can do rewrite systems, if no rewrite has an empty side:



$S =$

	s	t	e	b	r	l
s	·	s	·	·	·	·
t	s	t	·	·	·	l
e	·	·	·	e	·	·
b	·	·	e	b	r	·
r	·	r	·	·	·	e
l	·	·	·	l	s	·

, $S_+ = \{s, e\}$

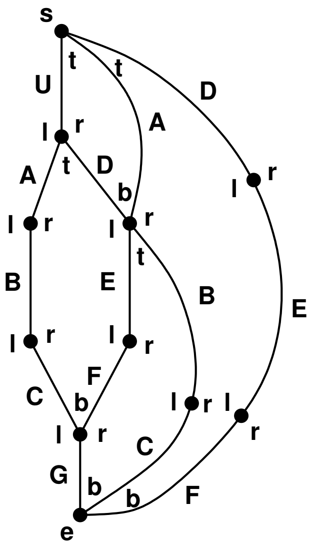
$X = \{A, B, C, D, E, F, G, U\}$ ($\bar{\ }$ is id_X)

This encodes $UABCG \to DEF$ using:

$$\{ABC \to DE, UD \to A, EFG \to BC\}$$

$ABC \to DE$ is encoded as $(bCrBrAtDlE)^{\circlearrowleft}$,

With K_6 we can do rewrite systems, if no rewrite has an empty side:



$S =$

	s	t	e	b	r	l
s	$\cdot$	s	$\cdot$	$\cdot$	$\cdot$	$\cdot$
t	s	t	$\cdot$	$\cdot$	$\cdot$	l
e	$\cdot$	$\cdot$	$\cdot$	e	$\cdot$	$\cdot$
b	$\cdot$	$\cdot$	e	b	r	$\cdot$
r	$\cdot$	r	$\cdot$	$\cdot$	$\cdot$	e
l	$\cdot$	$\cdot$	$\cdot$	l	s	$\cdot$

$, S_+ = \{s, e\}$

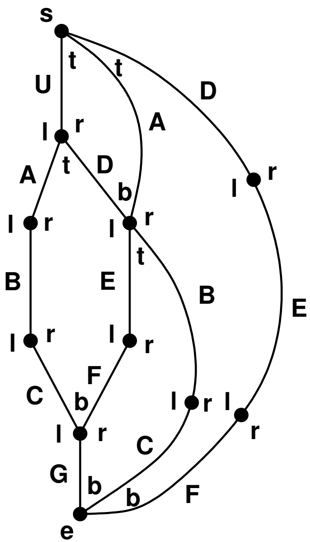
$X = \{A, B, C, D, E, F, G, U\}$ ($\bar{\ }$ is id_X)

This encodes $UABCG \to DEF$ using:

$$\{ABC \to DE, UD \to A, EFG \to BC\}$$

$ABC \to DE$ is encoded as $(bCrBrAtDlE)^{\circlearrowleft}$,
we "prove" $(sUlAlBlClGeFrErD)^{\circlearrowleft}$.

With K_6 we can do rewrite systems, if no rewrite has an empty side:



$S =$

	s	t	e	b	r	l
s	$\cdot$	s	$\cdot$	$\cdot$	$\cdot$	$\cdot$
t	s	t	$\cdot$	$\cdot$	$\cdot$	l
e	$\cdot$	$\cdot$	$\cdot$	e	$\cdot$	$\cdot$
b	$\cdot$	$\cdot$	e	b	r	$\cdot$
r	$\cdot$	r	$\cdot$	$\cdot$	$\cdot$	e
l	$\cdot$	$\cdot$	$\cdot$	l	s	$\cdot$

$, S_+ = \{s, e\}$

$X = \{A, B, C, D, E, F, G, U\}$ ($\bar{\ }$ is id_X)

This encodes $UABCG \to DEF$ using:

$$\{ABC \to DE, UD \to A, EFG \to BC\}$$

$ABC \to DE$ is encoded as $(bCrBrAtDlE)^{\circlearrowleft}$,
we "prove" $(sUlAlBlClGeFrErD)^{\circlearrowleft}$.

S accepts $st^* + eb^* + rt^*lb^*$ and all rotations.

Other Applications

These diagrams and their two fundamental problems encode

Other Applications

These diagrams and their two fundamental problems encode

- the word problem in quotients of the free group,

Other Applications

These diagrams and their two fundamental problems encode

- the word problem in quotients of the free group,
- the word problem in quotients of the modular group,

Other Applications

These diagrams and their two fundamental problems encode

- the word problem in quotients of the free group,
- the word problem in quotients of the modular group,
- the word problem for relative presentations (relative to one subgroup gives a Howie diagram)

Other Applications

These diagrams and their two fundamental problems encode

- the word problem in quotients of the free group,
- the word problem in quotients of the modular group,
- the word problem for relative presentations (relative to one subgroup gives a Howie diagram)
- the rewrite decision problem for rewrite systems, in which no side of a rewrite is empty,

Other Applications

These diagrams and their two fundamental problems encode

- the word problem in quotients of the free group,
- the word problem in quotients of the modular group,
- the word problem for relative presentations (relative to one subgroup gives a Howie diagram)
- the rewrite decision problem for rewrite systems, in which no side of a rewrite is empty,
- the word problem in finite semigroup presentations,

Other Applications

These diagrams and their two fundamental problems encode

- the word problem in quotients of the free group,
- the word problem in quotients of the modular group,
- the word problem for relative presentations (relative to one subgroup gives a Howie diagram)
- the rewrite decision problem for rewrite systems, in which no side of a rewrite is empty,
- the word problem in finite semigroup presentations,
- jigsaw-puzzles

Other Applications

These diagrams and their two fundamental problems encode

- the word problem in quotients of the free group,
- the word problem in quotients of the modular group,
- the word problem for relative presentations (relative to one subgroup gives a Howie diagram)
- the rewrite decision problem for rewrite systems, in which no side of a rewrite is empty,
- the word problem in finite semigroup presentations,
- jigsaw-puzzles in which you can use arbitrarily many copies of each piece,

Other Applications

These diagrams and their two fundamental problems encode

- the word problem in quotients of the free group,
- the word problem in quotients of the modular group,
- the word problem for relative presentations (relative to one subgroup gives a Howie diagram)
- the rewrite decision problem for rewrite systems, in which no side of a rewrite is empty,
- the word problem in finite semigroup presentations,
- jigsaw-puzzles in which you can use arbitrarily many copies of each piece,
- the word problem in monoids?

Other Applications

These diagrams and their two fundamental problems encode

- the word problem in quotients of the free group,
- the word problem in quotients of the modular group,
- the word problem for relative presentations (relative to one subgroup gives a Howie diagram)
- the rewrite decision problem for rewrite systems, in which no side of a rewrite is empty,
- the word problem in finite semigroup presentations,
- jigsaw-puzzles in which you can use arbitrarily many copies of each piece,
- the word problem in monoids?
- computations of non-deterministic Turing machines?

Other Applications

These diagrams and their two fundamental problems encode

- the word problem in quotients of the free group,
- the word problem in quotients of the modular group,
- the word problem for relative presentations (relative to one subgroup gives a Howie diagram)
- the rewrite decision problem for rewrite systems, in which no side of a rewrite is empty,
- the word problem in finite semigroup presentations,
- jigsaw-puzzles in which you can use arbitrarily many copies of each piece,
- the word problem in monoids?
- computations of non-deterministic Turing machines?
- etc. ???

Other Applications

These diagrams and their two fundamental problems encode

- the word problem in quotients of the free group,
- the word problem in quotients of the modular group,
- the word problem for relative presentations (relative to one subgroup gives a Howie diagram)
- the rewrite decision problem for rewrite systems, in which no side of a rewrite is empty,
- the word problem in finite semigroup presentations,
- jigsaw-puzzles in which you can use arbitrarily many copies of each piece,
- the word problem in monoids?
- computations of non-deterministic Turing machines?
- etc. ???

You just have to chose the right corner structure and edge alphabet!

Combinatorial Curvature

Find “pieces”, and remove vertices of valency 1 and 2:

Combinatorial Curvature

Find “pieces”, and remove vertices of valency 1 and 2:

- compute the finite list of all possible edges,

Combinatorial Curvature

Find "pieces", and remove vertices of valency 1 and 2:

- compute the finite list of all possible edges,
- this produces a new edge alphabet,

Combinatorial Curvature

Find “pieces”, and remove vertices of valency 1 and 2:

- compute the finite list of all possible edges,
- this produces a new edge alphabet, edges now have different lengths,

Combinatorial Curvature

Find “pieces”, and remove vertices of valency 1 and 2:

- compute the finite list of all possible edges,
- this produces a new edge alphabet, edges now have different lengths, refer to original edges as mini-edges,

Combinatorial Curvature

Find “pieces”, and remove vertices of valency 1 and 2:

- compute the finite list of all possible edges,
- this produces a new edge alphabet, edges now have different lengths, refer to original edges as mini-edges,
- denote the new set of half-edges in a diagram by $\hat{E}$.

Combinatorial Curvature

Find "pieces", and remove vertices of valency 1 and 2:

- compute the finite list of all possible edges,
- this produces a new edge alphabet, edges now have different lengths, refer to original edges as mini-edges,
- denote the new set of half-edges in a diagram by $\hat{E}$.

Combinatorical curvature: We endow

- each vertex with $+1$ unit of combinatorial curvature,
- each edge with -1 unit of combinatorial curvature and
- each internal face with $+1$ unit of combinatorial curvature.

Combinatorial Curvature

Find “pieces”, and remove vertices of valency 1 and 2:

- compute the finite list of all possible edges,
- this produces a new edge alphabet, edges now have different lengths, refer to original edges as mini-edges,
- denote the new set of half-edges in a diagram by $\hat{E}$.

Combinatorical curvature: We endow

- each vertex with $+1$ unit of combinatorial curvature,
- each edge with -1 unit of combinatorial curvature and
- each internal face with $+1$ unit of combinatorial curvature.

Euler’s formula

The total sum of our combinatorial curvature is always $+1$.

Idea (Curvature redistribution — Officers)

We redistribute the curvature locally in a conservative way.

Idea (Curvature redistribution — Officers)

We redistribute the curvature locally in a conservative way.
We call a curvature redistribution scheme an "officer".

Idea (Curvature redistribution — Officers)

We redistribute the curvature locally in a conservative way.
We call a curvature redistribution scheme an "officer".

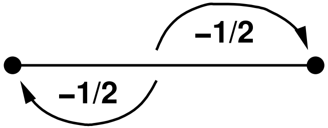
Here, I want to describe our "Officer Tom":

Idea (Curvature redistribution — Officers)

We redistribute the curvature locally in a conservative way.
We call a curvature redistribution scheme an "officer".

Here, I want to describe our "Officer Tom":

In Phase 1 Tom moves the negative curvature to the vertices:



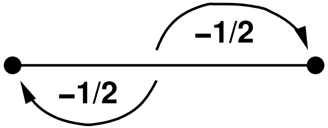
A vertex with valency $v \geq 3$ will now have $+1 - \frac{v}{2} < 0$.

Idea (Curvature redistribution — Officers)

We redistribute the curvature locally in a conservative way.
We call a curvature redistribution scheme an "officer".

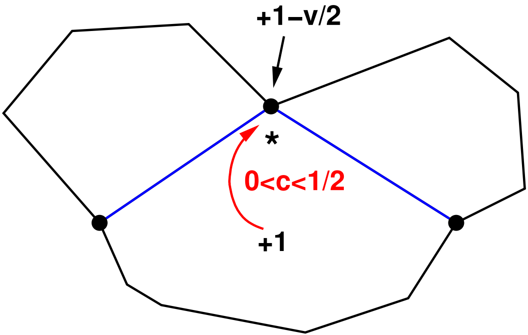
Here, I want to describe our "Officer Tom":

In Phase 1 Tom moves the negative curvature to the vertices:



A vertex with valency $v \geq 3$ will now have $+1 - \frac{v}{2} < 0$.
Faces still have $+1$, edges now have 0.

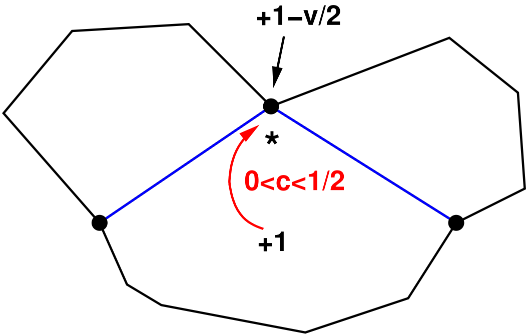
In Phase 2 Tom moves the negative curvature to the vertices:



Corner values for Tom

A corner value c of Tom depends on two edges that are adjacent on a face.

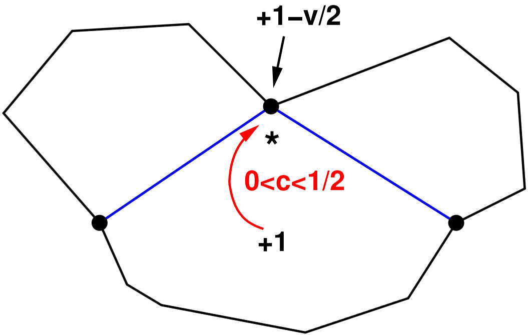
In Phase 2 Tom moves the negative curvature to the vertices:



Corner values for Tom

A corner value c of Tom depends on two edges that are adjacent on a face. Tom moves c units of curvature from the face to the vertex.

In Phase 2 Tom moves the negative curvature to the vertices:



Corner values for Tom

A corner value c of Tom depends on two edges that are adjacent on a face. Tom moves c units of curvature from the face to the vertex. The default value for c is $1/6$ if the vertex can have valency 3 and $1/4$ otherwise.

Tom — and officers in general — want to redistribute the curvature, such that for all permitted diagrams after redistribution

- every internal face has $< -\varepsilon$ curvature (for some explicit $\varepsilon > 0$),
- every vertex has ≤ 0 curvature.
- every edge has 0 curvature,

Tom — and officers in general — want to redistribute the curvature, such that for all permitted diagrams after redistribution

- every internal face has $< -\varepsilon$ curvature (for some explicit $\varepsilon > 0$),
- every vertex has ≤ 0 curvature.
- every edge has 0 curvature,
- every face with more than one external edge has ≤ 0 curvature.

Tom — and officers in general — want to redistribute the curvature, such that for all permitted diagrams after redistribution

- every internal face has $< -\varepsilon$ curvature (for some explicit $\varepsilon > 0$),
- every vertex has ≤ 0 curvature.
- every edge has 0 curvature,
- every face with more than one external edge has ≤ 0 curvature.

Consequence

$\Longrightarrow$ All the positive curvature is on faces touching the boundary once.

Tom — and officers in general — want to redistribute the curvature, such that for all permitted diagrams after redistribution

- every internal face has $< -\varepsilon$ curvature (for some explicit $\varepsilon > 0$),
- every vertex has ≤ 0 curvature.
- every edge has 0 curvature,
- every face with more than one external edge has ≤ 0 curvature.

Consequence

$\Longrightarrow$ All the positive curvature is on faces touching the boundary once.

Facts:

- All boundaries of diagrams have a permitted diagram as proof.

Tom — and officers in general — want to redistribute the curvature, such that for all permitted diagrams after redistribution

- every internal face has $< -\varepsilon$ curvature (for some explicit $\varepsilon > 0$),
- every vertex has ≤ 0 curvature.
- every edge has 0 curvature,
- every face with more than one external edge has ≤ 0 curvature.

Consequence

$\Longrightarrow$ All the positive curvature is on faces touching the boundary once.

Facts:

- All boundaries of diagrams have a permitted diagram as proof.
- The total positive curvature $\leq n$ (boundary length).

Tom — and officers in general — want to redistribute the curvature, such that for all permitted diagrams after redistribution

- every internal face has $< -\varepsilon$ curvature (for some explicit $\varepsilon > 0$),
- every vertex has ≤ 0 curvature.
- every edge has 0 curvature,
- every face with more than one external edge has ≤ 0 curvature.

Consequence

$\Longrightarrow$ All the positive curvature is on faces touching the boundary once.

Facts:

- All boundaries of diagrams have a permitted diagram as proof.
- The total positive curvature $\leq n$ (boundary length).
- Let $F := \#$internal faces, then

$$1 < n - F \cdot \varepsilon$$

Tom — and officers in general — want to redistribute the curvature, such that for all permitted diagrams after redistribution

- every internal face has $< -\varepsilon$ curvature (for some explicit $\varepsilon > 0$),
- every vertex has ≤ 0 curvature.
- every edge has 0 curvature,
- every face with more than one external edge has ≤ 0 curvature.

Consequence

$\Longrightarrow$ All the positive curvature is on faces touching the boundary once.

Facts:

- All boundaries of diagrams have a permitted diagram as proof.
- The total positive curvature $\leq n$ (boundary length).
- Let $F := \#$internal faces, then

$$1 < n - F \cdot \varepsilon \quad \Longrightarrow \quad F < \varepsilon^{-1} \cdot n$$

Tom — and officers in general — want to redistribute the curvature, such that for all permitted diagrams after redistribution

- every internal face has $< -\varepsilon$ curvature (for some explicit $\varepsilon > 0$),
- every vertex has ≤ 0 curvature.
- every edge has 0 curvature,
- every face with more than one external edge has ≤ 0 curvature.

Consequence

$\Longrightarrow$ All the positive curvature is on faces touching the boundary once.

Facts:

- All boundaries of diagrams have a permitted diagram as proof.
- The total positive curvature $\leq n$ (boundary length).
- Let $F := \#$internal faces, then

$$1 < n - F \cdot \varepsilon \quad \Longrightarrow \quad F < \varepsilon^{-1} \cdot n \quad \Longrightarrow \quad \text{hyperbolic}$$

Let $L := \{1, 2, \ldots, \ell\}$ and $a_1, a_2, \ldots, a_\ell \in \mathbb{R}$ and $S := \sum_{m \in L} a_m$.

Let $L := \{1, 2, \ldots, \ell\}$ and $a_1, a_2, \ldots, a_\ell \in \mathbb{R}$ and $S := \sum_{m \in L} a_m$.
Define $\pi_L : \mathbb{Z} \to L$ such that $z \equiv \pi_L(z) \pmod{\ell}$.

Let $L := \{1, 2, \ldots, \ell\}$ and $a_1, a_2, \ldots, a_\ell \in \mathbb{R}$ and $S := \sum_{m \in L} a_m$.
Define $\pi_L : \mathbb{Z} \to L$ such that $z \equiv \pi_L(z) \pmod{\ell}$.

Lemma (Goes up and stays up)

If $S \geq 0$ then there is a $j \in L$ such that for all $i \in \mathbb{N}$ the partial sum

$$s_{j,i} := \sum_{m=0}^{i-1} a_{\pi_L(j+m)} \geq 0.$$

Let $L := \{1, 2, \ldots, \ell\}$ and $a_1, a_2, \ldots, a_\ell \in \mathbb{R}$ and $S := \sum_{m \in L} a_m$.
Define $\pi_L : \mathbb{Z} \to L$ such that $z \equiv \pi_L(z) \pmod{\ell}$.

Lemma (Goes up and stays up)

If $S \geq 0$ then there is a $j \in L$ such that for all $i \in \mathbb{N}$ the partial sum

$$s_{j,i} := \sum_{m=0}^{i-1} a_{\pi_L(j+m)} \geq 0.$$

i	1	2	3	4	5	6	7
a_i	2	-3	4	1	-5	3	2
$s_{1,i}$	2	-1	3	4	-1	2	4
$s_{6,i}$	3	5	7	4	8	9	4

Let $L := \{1, 2, \ldots, \ell\}$ and $a_1, a_2, \ldots, a_\ell \in \mathbb{R}$ and $S := \sum_{m \in L} a_m$.
Define $\pi_L : \mathbb{Z} \to L$ such that $z \equiv \pi_L(z) \pmod{\ell}$.

Lemma (Goes up and stays up)

If $S \geq 0$ then there is a $j \in L$ such that for all $i \in \mathbb{N}$ the partial sum

$$s_{j,i} := \sum_{m=0}^{i-1} a_{\pi_L(j+m)} \geq 0.$$

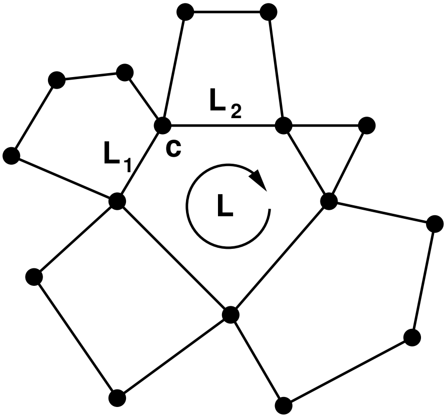
i	1	2	3	4	5	6	7
a_i	2	-3	4	1	-5	3	2
$s_{1,i}$	2	-1	3	4	-1	2	4
$s_{6,i}$	3	5	7	4	8	9	4

Corollary

Assume that there are $k \in \mathbb{N}$ and $\varepsilon \geq 0$ such that for all $j \in L$ there is an $i \leq k$ with $s_{j,i} < -\varepsilon$, then $S < -\varepsilon \cdot \ell / k$.

Sunflower

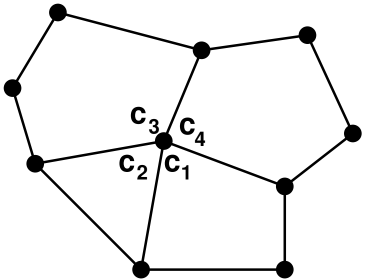
To show that every internal face has curvature $< -\varepsilon$:



Use Goes Up and Stays Up on $\frac{L_1+L_2}{2L} - c$.

Poppy

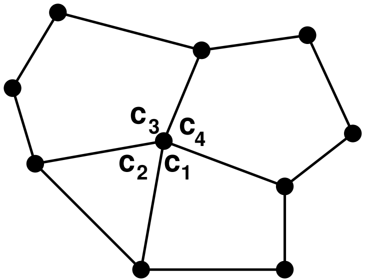
To show that every internal vertex has curvature ≤ 0:



Use Goes Up and Stays Up on $c + \frac{1-v/2}{v} = c + \frac{2-v}{v}$.

Poppy

To show that every internal vertex has curvature ≤ 0:

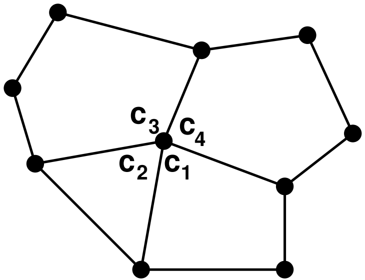


Use Goes Up and Stays Up on $c + \frac{1-v/2}{v} = c + \frac{2-v}{v}$.

Do valency $v = 3$ first, if nothing found, increase v.

Poppy

To show that every internal vertex has curvature ≤ 0:



Use Goes Up and Stays Up on $c + \frac{1-v/2}{v} = c + \frac{2-v}{v}$.

Do valency $v = 3$ first, if nothing found, increase v.

This terminates: higher valencies tend to be negatively curved anyway.

Overview over Tom analysis

What have we achieved?

Overview over Tom analysis

What have we achieved?

If we did not find any bad sunflower or poppy, we have

- determined an explicit ε,

Overview over Tom analysis

What have we achieved?

If we did not find any bad sunflower or poppy, we have

- determined an explicit ε,
- proved hyperbolicity, and

Overview over Tom analysis

What have we achieved?

If we did not find any bad sunflower or poppy, we have

- determined an explicit ε,
- proved hyperbolicity, and
- can in principle solve the diagram boundary problem.

Overview over Tom analysis

What have we achieved?

If we did not find any bad sunflower or poppy, we have

- determined an explicit ε,
- proved hyperbolicity, and
- can in principle solve the diagram boundary problem.

If we did find bad sunflowers or poppy, we can still

- improve our choices for the corner values
  (leads to difficult optimisation/linear program problems),

Overview over Tom analysis

What have we achieved?

If we did not find any bad sunflower or poppy, we have

- determined an explicit ε,
- proved hyperbolicity, and
- can in principle solve the diagram boundary problem.

If we did find bad sunflowers or poppy, we can still

- improve our choices for the corner values
  (leads to difficult optimisation/linear program problems),
- forbid more diagrams (if possible)
  (need to show that every boundary is proved by a permitted one),

Overview over Tom analysis

What have we achieved?

If we did not find any bad sunflower or poppy, we have

- determined an explicit ε,
- proved hyperbolicity, and
- can in principle solve the diagram boundary problem.

If we did find bad sunflowers or poppy, we can still

- improve our choices for the corner values (leads to difficult optimisation/linear program problems),
- forbid more diagrams (if possible) (need to show that every boundary is proved by a permitted one),
- or switch to a more powerful officer (with further sight or redistribution), ...

Overview over Tom analysis

What have we achieved?

If we did not find any bad sunflower or poppy, we have

- determined an explicit ε,
- proved hyperbolicity, and
- can in principle solve the diagram boundary problem.

If we did find bad sunflowers or poppy, we can still

- improve our choices for the corner values
  (leads to difficult optimisation/linear program problems),
- forbid more diagrams (if possible)
  (need to show that every boundary is proved by a permitted one),
- or switch to a more powerful officer
  (with further sight or redistribution), ...

and try again.

Overview over Tom analysis

What have we achieved?

If we did not find any bad sunflower or poppy, we have

- determined an explicit ε,
- proved hyperbolicity, and
- can in principle solve the diagram boundary problem.

If we did find bad sunflowers or poppy, we can still

- improve our choices for the corner values
  (leads to difficult optimisation/linear program problems),
- forbid more diagrams (if possible)
  (need to show that every boundary is proved by a permitted one),
- or switch to a more powerful officer
  (with further sight or redistribution), ...

and try again. If $\langle S, X; R \rangle$ is not hyperbolic, this will not work.