Vertex cuts and tree decomposition

Oberwolfach 2010

Bernhard Krön

University of Vienna

joint with Martin J. Dunwoody (Univ. of Southampton)
cut-points and separation

\[ X = (V_X, E_X) \text{ graph} \]

\[ s \in V_X \text{ cut point if } X - \{s\} \text{ is disconnected} \]

\[ \mathcal{S} = \text{set of cut-points} \]
cut-points and separation

\[ X = (V_X, E_X) \] graph

\( s \in V_X \) cut point if \( X - \{s\} \) is disconnected

\( S = \text{set of cut-points} \)

\( B \subset V_X \) is \( S \)-inseparable if no vertices in \( B \) are separated by any \( s \in S \).

\( S \)-block: maximal \( S \)-inseparable set = max. 2-connected subgraph

\( B = \text{set of } S \)-blocks
one-connected graph
decomposition of 1-connected graphs

tree $T$, $VT = S \cup B$, $ET = \{ \{v, B\} \mid v \in S, B \in B, v \in B \}$
decomposition of 1-connected graphs

tree $T$, $VT = S \cup B$, $ET = \{\{v, B\} \mid v \in S, B \in B, v \in B\}$

$S$ ... cut-points (separators) ... white vertices
$B$ ... blocks ... black vertices
tree decomposition of 1-connected graphs
decomposition of 2-connected graphs (Tutte)
decomposition of 2-connected graphs (Tutte)
decomposition of 2-connected graphs (Tutte)
$X$ connected graph (not necessarily locally finite), $C \subset VX$. 
A connected graph (not necessarily locally finite), $C \subset VX$.

**boundary:** $NC = \{x \in VX \setminus C \mid x \sim C\}$

***-complement:** $C^* = VX \setminus (C \cup NC)$
cuts

$X$ connected graph (not necessarily locally finite), $C \subset VX$.

**boundary:** $NC = \{ x \in VX \setminus C \mid x \sim C \}$

**$*$-complement:** $C^* = VX \setminus (C \cup NC)$

Think of cuts as “large” connected sets with finite boundary and “large” $*$-complement, whatever “large” may mean.
axioms for cut systems

A cut system $\mathcal{C}$ is a family of connected sets of vertices with finite boundaries which satisfies:

(A1) If $C$ is in $\mathcal{C}$ then $C^*$ contains an element of $\mathcal{C}$.

(A2) If $C$ is in $\mathcal{C}$ then every component of $C^*$ which contains an element of $\mathcal{C}$ is in $\mathcal{C}$. 
A cut system $\mathcal{C}$ is a family of connected sets of vertices with finite boundaries which satisfies:

(A1) If $C$ is in $\mathcal{C}$ then $C^*$ contains an element of $\mathcal{C}$. 
A cut system $\mathcal{C}$ is a family of connected sets of vertices with finite boundaries which satisfies:

(A1) If $C$ is in $\mathcal{C}$ then $C^*$ contains an element of $\mathcal{C}$.

(A2) If $C$ is in $\mathcal{C}$ then every component of $C^*$ which contains an element of $\mathcal{C}$ is in $\mathcal{C}$. 
Axioms for cut systems

\[
\begin{array}{ccc}
C \cap D^* & D^* \cap NC & C^* \cap D^* \\
C \cap ND & NC \cap ND & C^* \cap ND \\
C \cap D & D \cap NC & C^* \cap D
\end{array}
\]
### Axioms for Cut Systems

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$C \cap D^*$</td>
<td>$D^* \cap NC$</td>
<td>$C^* \cap D^*$</td>
</tr>
<tr>
<td>$C \cap ND$</td>
<td>$NC \cap ND$</td>
<td>$C^* \cap ND$</td>
</tr>
<tr>
<td>$C \cap D$</td>
<td>$D \cap NC$</td>
<td>$C^* \cap D$</td>
</tr>
</tbody>
</table>

(A3) If $C$ and $D$ are in $\mathcal{C}$ then either a component of $C \cap D$ and a component of $C^* \cap D^*$ are in $\mathcal{C}$ or a component of $C \cap D^*$ and a component of $C^* \cap D$ are in $\mathcal{C}$. 

"Bernhard Krön (Univ. of Vienna) Vertex cuts and tree decomposition 23.02.2010 9 / 32"
axioms for cut systems

\[
\begin{array}{ccc}
C \cap D^* & D^* \cap NC & C^* \cap D^* \\
C \cap ND & NC \cap ND & C^* \cap ND \\
C \cap D & D \cap NC & C^* \cap D \\
\end{array}
\]

(A3) If \( C \) and \( D \) are in \( C \) then either a component of \( C \cap D \) and a component of \( C^* \cap D^* \) are in \( C \) or a component of \( C \cap D^* \) and a component of \( C^* \cap D \) are in \( C \).

(A3)' If \( C \) and \( D \) are in \( C \) then \( C \setminus ND \) has a component which is an element of \( C \).
the cut system for rays/ends

Example

\[ C = \text{connected sets } C \text{ with finite } NC \text{ such that } C \text{ and } C^* \text{ contain a ray (an end).} \]
the cut system for rays/ends

Example

\( \mathcal{C} = \) connected sets \( C \) with finite \( NC \) such that \( C \) and \( C^* \) contain a ray (an end).

This system is used to prove Stallings’ Theorem about groups with more than one end for infinitely generated groups.
the cut system for rays/ends

Example

\( \mathcal{C} = \) connected sets \( C \) with finite \( NC \) such that \( C \) and \( C^* \) contain a ray (an end).

This system is used to prove Stallings’ Theorem about groups with more than one end for infinitely generated groups.

recent pre-print: B. Krön, “Cutting up graphs revisited - a short proof of Stallings’ Structure Theorem”
$B \subseteq VX$ is said to be \textit{k-inseparable} if $|B| \geq k + 1$ and if for every set $C \subseteq VX$ with $|NC| \leq k$, either $B \subseteq C \cup NC$ or $B \subseteq C^* \cup NC$. 

Example: $C$ connected sets with the above property and $|NC| = \kappa$. This system has a subsystem that yields a generalization of Tutte's decomposition of $k$-connected graphs for any $k$. 

Bernhard Krön (Univ. of Vienna)
the cut system for tree decompositions

$B \subset VX$ is said to be $k$-inseparable if $|B| \geq k + 1$ and if for every set $C \subset VX$ with $|NC| \leq k$, either $B \subset C \cup NC$ or $B \subset C^* \cup NC$.

Let $\kappa$ be the smallest positive integer for which there are sets $C$, $B_1$ and $B_2$ such that $|NC| = \kappa$, $B_1$ and $B_2$ are $\kappa$-inseparable, $B_1 \subset C \cup NC$ and $B_2 \subset C^* \cup NC$. 

Example: $C$ connected sets with the above property and $|NC| = \kappa$. This system has a subsystem that yields a generalization of Tutte’s decomposition of $k$-connected graphs for any $k$. 

Bernhard Krön (Univ. of Vienna)
The cut system for tree decompositions

A set $B \subseteq V_X$ is said to be $k$-inseparable if $|B| \geq k + 1$ and if for every set $C \subseteq V_X$ with $|NC| \leq k$, either $B \subseteq C \cup NC$ or $B \subseteq C^* \cup NC$.

Let $\kappa$ be the smallest positive integer for which there are sets $C$, $B_1$ and $B_2$ such that $|NC| = \kappa$, $B_1$ and $B_2$ are $\kappa$-inseparable, $B_1 \subset C \cup NC$ and $B_2 \subset C^* \cup NC$.

Example

$C = $ connected sets $C$ with the above property and $|NC| = \kappa$. 
the cut system for tree decompositions

\( B \subseteq VX \) is said to be \( k \)-inseparable if \( |B| \geq k + 1 \) and if for every set \( C \subseteq VX \) with \( |NC| \leq k \), either \( B \subseteq C \cup NC \) or \( B \subseteq C^* \cup NC \).

Let \( \kappa \) be the smallest positive integer for which there are sets \( C \), \( B_1 \) and \( B_2 \) such that \( |NC| = \kappa \), \( B_1 \) and \( B_2 \) are \( \kappa \)-inseparable, \( B_1 \subseteq C \cup NC \) and \( B_2 \subseteq C^* \cup NC \).

Example

\( C = \) connected sets \( C \) with the above property and \( |NC| = \kappa \).

This system has a subsystem that yields a generalization of Tutte’s decomposition of \( k \)-connected graphs for any \( k \).
the cut system for tree decompositions
the cut system for tree decompositions

\[ \kappa = 3 \]
isolated corner: contains no cut (is small), and adjacent links are empty.
### Diagram Nestedness

<table>
<thead>
<tr>
<th></th>
<th>$C \cap D^*$</th>
<th>$D^* \cap NC$</th>
<th>$C^* \cap D^*$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C \cap ND$</td>
<td>$NC \cap ND$</td>
<td>$C^* \cap ND$</td>
<td></td>
</tr>
<tr>
<td>$C \cap D$</td>
<td>$D \cap NC$</td>
<td>$C^* \cap D$</td>
<td></td>
</tr>
</tbody>
</table>

**Isolated corner:** contains no cut (is small), and adjacent links are empty.

$C$ and $D$ are nested $\iff$ there is an isolated corner.
an isolated corner

Diagram showing vertices labeled as $C$, $NC$, $C^*$, $D^*$, $ND$, and $D$. The diagram illustrates the relationships between these vertices with lines connecting them.
C is **minimal** if \(|NC|\) is minimal, \(C \in \mathcal{C}\).

\(\mathcal{C}\) is **minimal** if all cuts are minimal.
$C$ minimal if $|NC|$ is minimal, $C \in \mathcal{C}$.

$\mathcal{C}$ is minimal if all cuts are minimal.

**Theorem**

*The minimal cuts in a cut system form a cut system.*
When we replace the boundaries $NC$ of cuts by complete graphs and cut off the isolated corners ("slices") then we obtain a connected graph $\hat{X}$. $X \leftrightarrow \hat{X}$, $C \leftrightarrow \hat{C}$, the structure essentially remains the same.
nestedness in $\hat{X}$ and transitive graphs

When we replace the boundaries $NC$ of cuts by complete graphs and cut off the isolated corners ("slices") then we obtain a connected graph $\hat{X}$. $X \leftrightarrow \hat{X}$, $C \leftrightarrow \hat{C}$, the structure essentially remains the same.

Minimal cuts $C, D$ in $\hat{X}$ are nested $\iff$

\[ C \subset D, \quad C^* \subset D, \quad C \subset D^* \quad \text{or} \quad C^* \subset D^*. \]
When we replace the boundaries \( NC \) of cuts by complete graphs and cut off the isolated corners ("slices") then we obtain a connected graph \( \hat{X} \). \( X \leftrightarrow \hat{X} \), \( C \leftrightarrow \hat{C} \), the structure essentially remains the same.

Minimal cuts \( C, D \) in \( \hat{X} \) are nested \( \iff \)
\[
C \subset D, \quad C^* \subset D, \quad C \subset D^* \quad \text{or} \quad C^* \subset D^*.
\]

**Lemma**

Let \( C \) be minimal. Then slices have empty intersection with each separator. Distinct slices are disjoint.
nestedness in $\hat{X}$ and transitive graphs

When we replace the boundaries $\mathcal{NC}$ of cuts by complete graphs and cut off the isolated corners ("slices") then we obtain a connected graph $\hat{X}$. $X \leftrightarrow \hat{X}$, $C \leftrightarrow \hat{C}$, the structure essentially remains the same.

Minimal cuts $C, D$ in $\hat{X}$ are nested $\iff$ $C \subset D$, $C^* \subset D$, $C \subset D^*$ or $C^* \subset D^*$.

**Lemma**

Let $C$ be minimal. Then slices have empty intersection with each separator. Distinct slices are disjoint.

**Corollary**

There are no slices in transitive graph with an automorphism-invariant cut-system. That is, $X = \hat{X}$ and nestedness is defined by inclusion.
Lemma

In a minimal cut system, every cut is nested with all but finitely many cuts.
Lemma

In a minimal cut system, every cut is nested with all but finitely many cuts.

\[ \mu(C) = \text{number of cuts not nested with } C \]

optimally nested cut: \( \mu(C) \) is minimal
Lemma

In a minimal cut system, every cut is nested with all but finitely many cuts.

\( C \in \mathcal{C}, \ \mu(C) = \text{number of cuts not nested with } C \)

optimally nested cut: \( \mu(C) \) is minimal
**Lemma**

In a minimal cut system, every cut is nested with all but finitely many cuts.

\[ C \in \mathcal{C}, \quad \mu(C) = \text{number of cuts not nested with } C \]

**optimally nested cut:** \( \mu(C) \) is minimal

\[ \mu(C) > 0 \]

\[ C^* \text{ NC } C \]

\( C \) is optimally nested
Main Theorem

*Optimally nested cuts are nested with each other.*
Main Theorem

Optimally nested cuts are nested with each other.

cut-system is nested: all pairs of cuts are nested

\( \mathcal{O}(C) = \text{set of optimally nested cuts in } C \)
existence of nested cut-systems

Main Theorem

Optimally nested cuts are nested with each other.

cut-system is nested: all pairs of cuts are nested
\( \mathcal{O}(C) \) = set of optimally nested cuts in \( C \)

Corollary

Every automorphism-invariant cut system \( C \) contains a nested automorphism-invariant subsystem, for instance \( \mathcal{O}(C) \).
blocks

$B \subset VX$ is $C$-inseparable if for every $C \in C$
either $B \subset C \cup NC$ or $B \subset C^* \cup NC$, but not both.
blocks

$B \subset VX$ is $\mathcal{C}$-inseparable if for every $C \in \mathcal{C}$
either $B \subset C \cup NC$ or $B \subset C^* \cup NC$, but not both.

$\mathcal{C}$-block: maximal $\mathcal{C}$-inseparable set, not contained in any set $A \cup NA$, where $A$ is a slice.
blocks

$B \subset VX$ is $\mathcal{C}$-inseparable if for every $C \in \mathcal{C}$ either $B \subset C \cup NC$ or $B \subset C^* \cup NC$, but not both.

$\mathcal{C}$-block: maximal $\mathcal{C}$-inseparable set, not contained in any set $A \cup NA$, where $A$ is a slice.

The blocks in $X$ are the same as the (projected) blocks in $\hat{X}$. 
blocks

$B \subset VX$ is $\mathcal{C}$-inseparable if for every $C \in \mathcal{C}$ either $B \subset C \cup NC$ or $B \subset C^* \cup NC$, but not both.

$\mathcal{C}$-block: maximal $\mathcal{C}$-inseparable set, not contained in any set $A \cup NA$, where $A$ is a slice.

The blocks in $X$ are the same as the (projected) blocks in $\hat{X}$.

Lemma

For each $C \in \mathcal{C}$ there is precisely one block $B(C)$ such that $NC \subset B(C) \subset C \cup NC$. Moreover, $B(C) \setminus NC \neq \emptyset$.
blocks

$B \subset VX$ is $\mathcal{C}$-inseparable if for every $C \in \mathcal{C}$
either $B \subset C \cup NC$ or $B \subset C^* \cup NC$, but not both.

$\mathcal{C}$-block: maximal $\mathcal{C}$-inseparable set, not contained in any set $A \cup NA$, where $A$ is a slice.

The blocks in $X$ are the same as the (projected) blocks in $\hat{X}$.

Lemma

For each $C \in \mathcal{C}$ there is precisely one block $B(C)$ such that
$NC \subset B(C) \subset C \cup NC$. Moreover, $B(C) \setminus NC \neq \emptyset$.

For some block $B$ let $\mathcal{C}(B) = \{C \in \mathcal{C} \mid B(C) = B\}$ then

$$B = \bigcap_{C \in \mathcal{C}(B)} C \cup NC.$$
the general tree construction

\[ \mathcal{B} = \text{set of all } C\text{-blocks} \]

tree \( T = T(\mathcal{C}) \)

\( \mathcal{V}T = \mathcal{S} \cup \mathcal{B} \)

\( \mathcal{S} \) ... separators ... white vertices

\( \mathcal{B} \) ... blocks ... black vertices
the general tree construction

\[ B = \text{set of all } C\text{-blocks} \]
\[ \text{tree } T = T(C) \]
\[ VT = S \cup B \]

S ... separators ... white vertices
B ... blocks ... black vertices

ET: vertices \( S \in S \) are only adjacent to vertices \( B \in B \) (and vice versa)
\[ S \sim B \iff S \subset B. \]
cuts → blocks → tree
tree decomposition of graph with slice

\[ \kappa = 3 \]

\[ \{1, 2, a, b\} \]

\[ \{2, 3, a, b\} \]

\[ \{n - 1, n, a, b\} \]
existence of nested $\mathcal{C}$-subsystems generalizes edge cuts to vertex cuts, see Dunwoody “Cutting up graphs” (1982), Dicks and Dunwoody “Groups acting on graphs” (1989).
edge cuts → vertex cuts

existence of nested $C$-subsystems generalizes edge cuts to vertex cuts, see Dunwoody “Cutting up graphs” (1982), Dicks and Dunwoody “Groups acting on graphs” (1989).

Farey graph with tree decomposition
Tutte’s tree decomposition of $k$-connected graphs, for any $k \geq 1$.

recall: $\kappa$ minimal such that there are at least two $\kappa$-inseparable sets which are separated by $\kappa$ vertices. Consider corresponding cut-system $C$. 
Tutte’s tree decomposition of $k$-connected graphs, for any $k \geq 1$.

recall: $\kappa$ minimal such that there are at least two $\kappa$-inseparable sets which are separated by $\kappa$ vertices. Consider corresponding cut-system $C$.

How to find an visualize the tree-decomposition explicitly:

1. find the set optimally nested cuts $O(C)$
2. cut off slices
3. determine $O$-blocks
4. construct $T(O)$
application: decomposition of \( k \)-connected graphs

Tutte’s tree decomposition of \( k \)-connected graphs, for any \( k \geq 1 \).

recall: \( \kappa \) minimal such that there are at least two \( \kappa \)-inseparable sets which are separated by \( \kappa \) vertices. Consider corresponding cut-system \( C \).

How to find an visualize the tree-decomposition explicitly:

1. find the set optimally nested cuts \( O(C) \)
Tutte’s tree decomposition of \( k \)-connected graphs, for any \( k \geq 1 \).

recall: \( \kappa \) minimal such that there are at least two \( \kappa \)-inseparable sets which are separated by \( \kappa \) vertices. Consider corresponding cut-system \( C \).

How to find an visualize the tree-decomposition explicitly:

1. find the set optimally nested cuts \( O(C) \)
2. cut off slices
Tutte’s tree decomposition of $k$-connected graphs, for any $k \geq 1$.

recall: $\kappa$ minimal such that there are at least two $\kappa$-inseparable sets which are separated by $\kappa$ vertices. Consider corresponding cut-system $\mathcal{C}$.

How to find an visualize the tree-decomposition explicitly:

1. find the set optimally nested cuts $\mathcal{O}(\mathcal{C})$
2. cut off slices
3. determine $\mathcal{O}$-blocks
application: decomposition of $k$-connected graphs

Tutte’s tree decomposition of $k$-connected graphs, for any $k \geq 1$.

recall: $\kappa$ minimal such that there are at least two $\kappa$-inseparable sets which are separated by $\kappa$ vertices. Consider corresponding cut-system $\mathcal{C}$.

How to find an visualize the tree-decomposition explicitly:

1. find the set optimally nested cuts $\mathcal{O}(\mathcal{C})$
2. cut off slices
3. determine $\mathcal{O}$-blocks
4. construct $T(\mathcal{O})$
the dragon neck graph

there is a disconnected \(\mathcal{O}\)-block (which is also a \(\mathcal{C}\)-block)
there are \(\mathcal{O}\)-blocks which are no \(\mathcal{C}\)-blocks
the dragon neck graph

there is a disconnected $\mathcal{O}$-block (which is also a $\mathcal{C}$-block)
there are $\mathcal{O}$-blocks which are no $\mathcal{C}$-blocks
Consider optimally nested cuts.

Leaves of $T$ (blocks without boundary):

- $s_1 = \{1, 10\}$
- $s_2 = \{3, 10\}$
- $s_3 = \{1, 3\}$
- $s_4 = \{11, 13\}$
- $s_5 = \{13, 19\}$
- $s_6 = \{19, 24\}$
- $s_7 = \{19, 20\}$
- $s_8 = \{20, 21\}$
- $s_9 = \{21, 22\}$
- $s_{10} = \{19, 22\}$
Consider optimally nested cuts. Leaves of $T$ (blocks without boundary):

\{2\} ($b_5$), \{5\} ($b_1$), \{6\} ($b_2$), \{7, 8\} ($b_3$), \{9\} ($b_4$), \{12\} ($b_7$),
\{14, 15, 16, 17, 18\} ($b_8$), \{23\} ($b_{13}$), \{25\} ($b_{10}$), \{26\} ($b_{11}$), \{27\} ($b_{12}$).
a more complicated example
a more complicated example

\[ b_6 = \{1, 3, 4, 10\}, \quad b_9 = \{19, 20, 24\} \text{ light brown} \]
a more complicated example

\begin{align*}
b_6 &= \{1, 3, 4, 10\}, \quad b_9 = \{19, 20, 24\} \text{ light brown} \\
b_{14} &= \{1, 10, 11, 13, 19, 24\}, \quad b_{15} = \{19, 20, 21, 22\} \text{ dark brown}
\end{align*}
a more complicated example
What are “structure trees”? 

origin: group actions on infinite graphs
What are “structure trees”? 

origin: group actions on infinite graphs

A structure tree is a tree from a tree decomposition which is invariant under all automorphisms.

A group which acts on the graph will act on the structure tree.

Original purpose: to apply Bass-Serre-Theory to get a proof of Stallings’ Structure Theorem about groups with more than one end.

Further applications mainly for infinite graphs with group actions: accessibility, highly arc-transitive digraphs, transitive maps,...
What are “structure trees”?

origin: group actions on infinite graphs

A structure tree is a tree from a tree decomposition which is invariant under all automorphisms.

A group which acts on the graph will act on the structure tree.
Original purpose: to apply Bass-Serre-Theory to get a proof of Stallings’ Structure Theorem about groups with more than one end.
What are “structure trees”? 

origin: group actions on infinite graphs

A **structure tree** is a tree from a tree decomposition which is invariant under all automorphisms.

A group which acts on the graph will act on the structure tree. Original purpose: to apply Bass-Serre-Theory to get a proof of Stallings’ Structure Theorem about groups with more than one end.

Further applications mainly for infinite graphs with group actions: accessibility, highly arc-transitive digraphs, transitive maps,…
application: Stalling’s Theorem

$G$ acts on (possibly non-locally finite) Cayley-graph with more than one end by left multiplication.

⇒ $\exists$ non-trivial nested $G$-invariant cut system $O$.

⇒ $G$ acts on $T = T(O)$ without edge inversion.

The edge-stabilizers are finite, because they are contained in the stabilizer of a finite set of vertices (i.e. in the stabilizer of a finite subset of $G$).

Bass-Serre theory ⇒ $G$ splits over edge stabilizer (i.e. over finite subgroup) ⇒ Stallings’ theorem for infinitely generated groups.
application: Stalling’s Theorem

$G$ acts on (possibly non-locally finite) Cayley-graph with more than one end by left multiplication

$\Rightarrow \exists$ non-trivial nested $G$-invariant cut system $\mathcal{O}$.
application: Stalling’s Theorem

\[ G \text{ acts on (possibly non-locally finite) Cayley-graph with more than one end by left multiplication} \]
\[ \Rightarrow \exists \text{ non-trivial nested } G\text{-invariant cut system } \mathcal{O}. \]
\[ \Rightarrow G \text{ acts on } T = T(\mathcal{O}) \text{ without edge inversion.} \]
application: Stalling’s Theorem

$G$ acts on (possibly non-locally finite) Cayley-graph with more than one end by left multiplication
⇒ $\exists$ non-trivial nested $G$-invariant cut system $\mathcal{O}$.
⇒ $G$ acts on $T = T(\mathcal{O})$ without edge inversion.

The edge-stabilizers are finite, because they are contained in the stabilizer of a finite set of vertices (i.e. in the stabilizer of a finite subset of $G$).
application: Stalling’s Theorem

$G$ acts on (possibly non-locally finite) Cayley-graph with more than one end by left multiplication

$\Rightarrow \exists$ non-trivial nested $G$-invariant cut system $\mathcal{O}$.

$\Rightarrow G$ acts on $T = T(\mathcal{O})$ without edge inversion.

The edge-stabilizers are finite, because they are contained in the stabilizer of a finite set of vertices (i.e. in the stabilizer of a finite subset of $G$).

Bass-Serre theory $\Rightarrow G$ splits over edge stabilizer (i.e. over finite subgroup)
application: Stalling’s Theorem

$G$ acts on (possibly non-locally finite) Cayley-graph with more than one end by left multiplication

$\Rightarrow \exists$ non-trivial nested $G$-invariant cut system $\mathcal{O}$.

$\Rightarrow G$ acts on $T = T(\mathcal{O})$ without edge inversion.

The edge-stabilizers are finite, because they are contained in the stabilizer of a finite set of vertices (i.e. in the stabilizer of a finite subset of $G$).

Bass-Serre theory $\Rightarrow G$ splits over edge stabilizer (i.e. over finite subgroup)

$\Rightarrow$ Stallings’ theorem for infinitely generated groups.
contents

1 decomposition of 1- and 2-connected graphs
2 axiomatic cut systems
3 minimal subsystems
4 nestedness and slices
5 optimally nested subsystems
6 blocks and trees
7 decompositions of $k$-connected graphs