Structure and Constructions of 3-Connected Graphs

Jens M. Schmidt
Overview

- Construction Sequences

- Certifying 3-Vertex- and Edge-Connectivity

- Some Open Problems
Def. An edge is **contractible** if contracting it generates a 3-connected graph.
Def. An edge is **contractible** if contracting it generates a 3-connected graph.

**Thm (Tutte '61).** Every 3-connected graph with $n > 4$ contains a **contractible** edge.
**Def.** An edge is **contractible** if contracting it generates a 3-connected graph.

**Thm (Tutte '61).** Every 3-connected graph with \( n > 4 \) contains a contractible edge.

**Thm (Elmasry, Mehlhorn, S. '10).** Every DFS-tree of a 3-connected graph with \( n > 4 \) contains a contractible edge.
Def. An edge is **contractible** if contracting it generates a 3-connected graph.
Def. An edge is **contractible** if contracting it generates a 3-connected graph.
**Def.** An edge is **contractible** if contracting it generates a 3-connected graph.
**Contractible Edges**

**Def.** An edge is *contractible* if contracting it generates a 3-connected graph.
**Contractible Edges**

**Def.** An edge is **contractible** if contracting it generates a 3-connected graph.

**Corollary (of Tutte '61).**
A graph is 3-connected $\iff \exists$ a sequence of contractions from $G$ to a $K_4$-multigraph on contractible edges $xy$ with $|N(x)| \geq 3$ and $|N(y)| \geq 3$. 
**K₄-Subdivisions**

**Thm (Isbell).** Every 3-connected graph contains a subdivision of $K₄$.
Thm (Isbell). Every 3-connected graph contains a subdivision of $K_4$. 
Thm (Isbell). Every 3-connected graph contains a subdivision of $K_4$. 

**K$_4$-Subdivisions**

*links of the K$_4$-subdivision*
BG-Paths

David W. Barnette
Branko Grünbaum

\[ S \cap P = \{x,y\} \]

forbidden

forbidden
Thm (Corollary of Barnette & Grünbaum ’69).

A graph $G$ is 3-connected $\iff$ 
$\delta(G) \geq 3$ and there is a sequence of BG-paths from a $K_4$-subdivision in $G$ to $G$
Thm (Corollary of Barnette & Grünbaum '69).
A graph $G$ is 3-connected $\iff \delta(G) \geq 3$ and there is a sequence of BG-paths from a $K_4$-subdivision in $G$ to $G$

Thm (S., STACS '10).
A graph $G$ is 3-connected $\iff \delta(G) \geq 3$ and there is a sequence of BG-paths from each $K_4$-subdivision in $G$ to $G$
Construction Sequences

Thm (Corollary of Barnette & Grünbaum '69).
A graph $G$ is 3-connected $\iff \delta(G) \geq 3$ and there is a sequence of BG-paths from a $K_4$-subdivision in $G$ to $G$

Thm (S., STACS '10).
A graph $G$ is 3-connected $\iff \delta(G) \geq 3$ and there is a sequence of BG-paths from each $K_4$-subdivision in $G$ to $G$

True even for subdivisions of any 3-connected graph in $G$!
Construction Sequences

Thm (Corollary of Barnette & Grünbaum '69).
A graph $G$ is 3-connected $\iff$ 
$\delta(G) \geq 3$ and there is a sequence of BG-paths from a $K_4$-subdivision in $G$ to $G$

Thm (S., STACS '10).
A graph $G$ is 3-connected $\iff$ 
$\delta(G) \geq 3$ and there is a sequence of BG-paths from each $K_4$-subdivision in $G$ to $G$

This allows for a computational greedy approach: We compute a $K_4$-subdivision and add iteratively BG-paths!
A $K_4$-subdivision can be found easily using a DFS-tree.

Thm (S. '11). A BG-path sequence of a 3-connected graph can be computed in $O(m)$ time.
A $K_4$-subdivision can be found easily using a DFS-tree.

Thm (S. '11). A BG-path sequence of a 3-connected graph can be computed in $O(m)$ time.

Thm (S. '10). Any BG-path sequence can be transformed to Tutte's sequence of contractions in $O(m)$. 
Overview

- Construction Sequences

- Certifying 3-Vertex- and Edge-Connectivity

- Some Open Problems
Certifying 3-Connectivity

not 3-connected

3-connected?
Certifying 3-Connectivity

A certifying algorithm is an algorithm that gives an easy-to-verify certificate of correctness along with its output. (Mehlhorn, Näher, et al. '98 -'10, see also Blum & Kannan '89)
A certifying algorithm is an algorithm that gives an easy-to-verify certificate of correctness along with its output. (Mehlhorn, Näher, et al. '98 -'10, see also Blum & Kannan '89)

History of certifying algorithms for 3-connectivity:

- Straight-forward $O(n^2m^2)$ algorithm
- $O(n^3)$ by applying a preprocessing of Nagamochi&Ibaraki
- 2006 – Albroscheit & Rote: $O(n^2)$, mixed certificates
- 2010 – Mehlhorn, Schweitzer: $O(n^2)$, computes Tutte's sequence
- 2010 – S.: $O(n^2)$, computes Tutte's and the BG-path sequence
- 2010 – Elmasry, Mehlhorn, S.: $O(n+m)$ for Hamiltonian graphs
Certifying 3-Connectivity

A **certifying algorithm** is an algorithm that gives an **easy-to-verify** certificate of correctness along with its output. (Mehlhorn, Näher, et al. '98 -'10, see also Blum & Kannan '89)

History of certifying algorithms for 3-connectivity:

- Straight-forward $O(n*m^2)$ algorithm
- $O(n^3)$ by applying a preprocessing of Nagamochi&Ibaraki
- 2006 – Albroscheit & Rote: $O(n^2)$, mixed certificates
- 2010 – Mehlhorn, Schweitzer: $O(n^2)$, computes Tutte's sequence
- 2010 – S.: $O(n^2)$, computes Tutte's and the BG-path sequence
- 2010 – Elmasry, Mehlhorn, S.: $O(n+m)$ for Hamiltonian graphs

**Thm (S. '10).** A BG-path sequence can be **easily** verified in time $O(m)$. 
Certifying 3-Connectivity

A certifying algorithm is an algorithm that gives an easy-to-verify certificate of correctness along with its output. (Mehlhorn, Näher, et al. '98 -'10, see also Blum & Kannan '89)

History of certifying algorithms for 3-connectivity:

- Straight-forward $O(n^2 m^2)$ algorithm
- $O(n^3)$ by applying a preprocessing of Nagamochi&Ibaraki
- 2006 – Albroscheit & Rote: $O(n^2)$, mixed certificates
- 2010 – Mehlhorn, Schweitzer: $O(n^2)$, computes Tutte's sequence
- 2010 – S.: $O(n^2)$, computes Tutte's and the BG-path sequence
- 2010 – Elmasry, Mehlhorn, S.: $O(n+m)$ for Hamiltonian graphs

**Thm (S. '10).** A BG-path sequence can be easily verified in time $O(m)$.

- 2011 – S.: $O(n+m)$ for general graphs, using BG-paths
- also: 3-edge-connectivity
Overview

- Construction Sequences
- Certifying 3-Vertex- and Edge-Connectivity
- Some Open Problems
Some Open Problems

- Implementation is simple. Performance in practice? Implementations of Rote and of Neumann.
- Can the algorithm be extended to compute SPQR-trees?
- Is the construction sequence approach strong enough for higher connectivity?
BG-Paths

$K_4$-subdivision in $G$
BG-Paths

$K_4$-subdivision in $G$
BG-Paths

$K_4$-subdivision in $G$
BG-Paths

$K_4$-subdivision in $G$
BG-Paths

K₄-subdivision in G
BG-Paths

$K_4$-subdivision in $G$
Certifying 3-Connectivity

A certifying algorithm is an algorithm that gives an easy-to-verify certificate of correctness along with its output. (Mehlhorn, Näher, et al. '98 -'10, see also Blum & Kannan '89)

- **Major goal** for program verification on complicated algorithms
- **Assure** correctness on every instance
- **Surprisingly few results** to certify 3-connectivity although sophisticated linear-time recognition algorithms are known for over 35 years (Hopcroft & Tarjan '73, Vo '83)
Motivation

=> new *linear-time* recognition algorithm for *3-connectivity*.

**Thm (Galil & Italiano ’91).** *k-edge-connectivity* is *linear-time* reducible to *k-connectivity* in linear time.

**Corollary.** There is a *linear-time* certifying algorithm for *3-edge-connectivity*. 
Thm (S. '10). A BG-path sequence can be easily verified in time $O(m)$.

=> new linear-time recognition algorithm for 3-connectivity.

Thm (Galil & Italiano '91). $k$-edge-connectivity is linear-time reducible to $k$-connectivity in linear time.

Corollary. There is a linear-time certifying algorithm for 3-edge-connectivity.