Non-Planarity Measures and Small Cycles

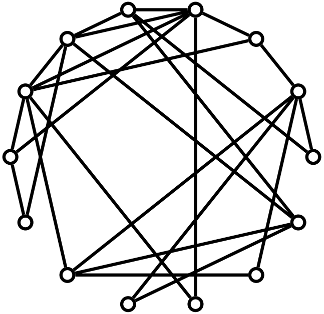
EuroCG 2020 – PhD School

Markus Chimani

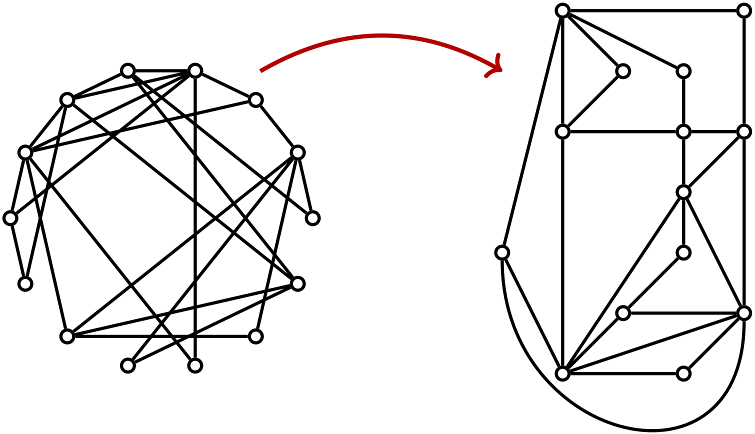
Theoretical Computer Science, Osnabrück University

March 19th, 2020

Planarity

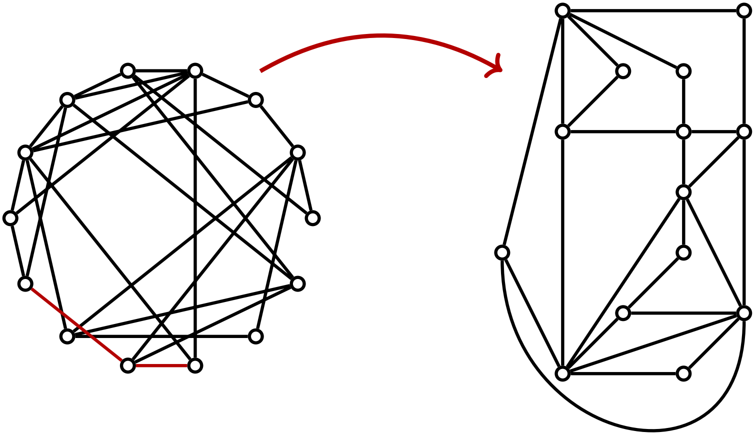


Planarity

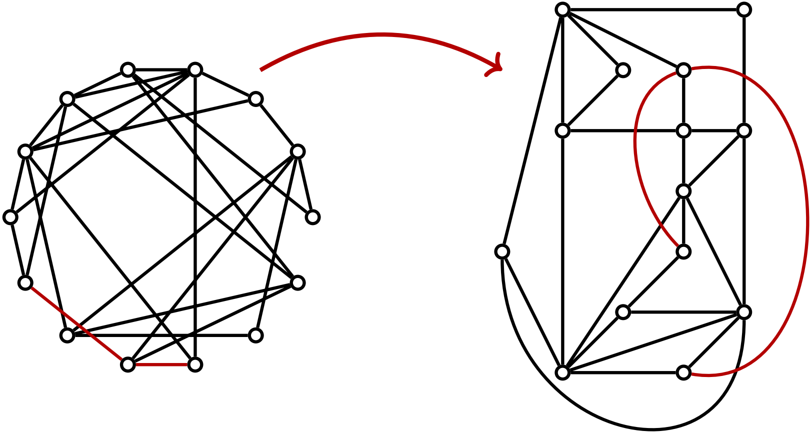


Planarity helps **a lot** algorithmically!

Planarity



Planarity

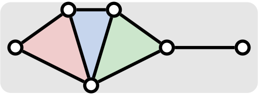


Euler

Theorem (Euler's formula)

Given a planarly drawn graph with n vertices, m edges, and f faces, then:

$$n - m + f = 2$$



Euler

Theorem (Euler's formula)

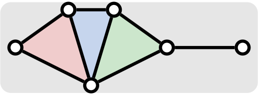
Given a planarly drawn graph with n vertices, m edges, and f faces, then:

$$n - m + f = 2$$

Consider the number of face-edge incidences:

≥ 3 edges per face

≤ 2 faces per edge



Euler

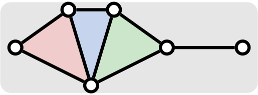
Theorem (Euler's formula)

Given a planarly drawn graph with n vertices, m edges, and f faces, then:

$$n - m + f = 2$$

Consider the number of face-edge incidences: ≥ 3 edges per face

≤ 2 faces per edge



$\implies 3f \geq 2m \quad \rightarrow \quad n - m + (2/3)m \geq 2 \quad \rightarrow \quad m \leq 3n - 6$

Euler

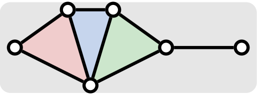
Theorem (Euler's formula)

Given a planarly drawn graph with n vertices, m edges, and f faces, then:

$$n - m + f = 2$$

Consider the number of face-edge incidences: ≥ 3 edges per face

≤ 2 faces per edge



$\implies 3f \geq 2m \quad \rightarrow \quad n - m + (2/3)m \geq 2 \quad \rightarrow \quad m \leq 3n - 6$

Definition (Girth)

Girth $g(G)$ is the length of the shortest cycle in G.

Euler

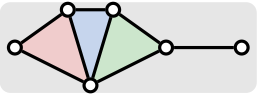
Theorem (Euler's formula)

Given a planarly drawn graph with n vertices, m edges, and f faces, then:

$$n - m + f = 2$$

Consider the number of face-edge incidences: ≥ 3 edges per face

≤ 2 faces per edge



$\implies 3f \geq 2m \rightarrow n - m + (2/3)m \geq 2 \rightarrow m \leq 3n - 6$

Definition (Girth)

Girth $g(G)$ is the length of the shortest cycle in G.

Bipartite graphs (girth ≥ 4): ≥ 4 edges per face $\implies m \leq 2n - 4$

Euler

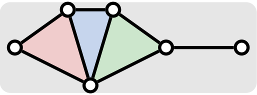
Theorem (Euler's formula)

Given a planarly drawn graph with n vertices, m edges, and f faces, then:

$$n - m + f = 2$$

Consider the number of face-edge incidences: ≥ 3 edges per face

≤ 2 faces per edge



$\implies 3f \geq 2m \quad \rightarrow \quad n - m + (2/3)m \geq 2 \quad \rightarrow \quad m \leq 3n - 6$

Definition (Girth)

Girth $g(G)$ is the length of the shortest cycle in G.

Bipartite graphs (girth ≥ 4): ≥ 4 edges per face $\implies m \leq 2n - 4$

Girth g: $\geq g$ edges per face $\implies m \leq \frac{g}{g-2}(n-2)$

high girth $\rightarrow$ few edges

Kuratowski

Theorem (Kuratowski, 1930)

*Graph G is **non-planar***

$$\iff$$

*G contains a K_5- or $K_{3,3}$-**subdivision** as a subgraph.*

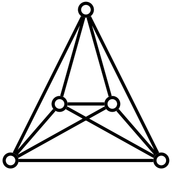
Kuratowski

Theorem (Kuratowski, 1930)

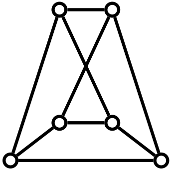
*Graph G is **non-planar***

$\Longleftrightarrow$

*G contains a K_5- or $K_{3,3}$-**subdivision** as a subgraph.*



K_5



$K_{3,3}$

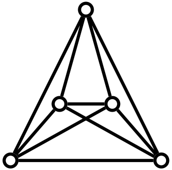
Kuratowski

Theorem (Kuratowski, 1930)

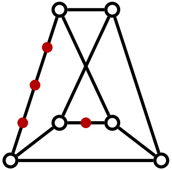
Graph G is ***non-planar***

$\Longleftrightarrow$

*G contains a K_5- or $K_{3,3}$-**subdivision** as a subgraph.*



K_5



$K_{3,3}$-subdivision

Non-Planarity Measures

Can we gain an algorithmic advantage is graph is close to planar?

What does close mean?

Non-Planarity Measures

Can we gain an algorithmic advantage is graph is close to planar?
What does close mean?

- **Crossing Number:** min. number of crossings in the plane

Non-Planarity Measures

Can we gain an algorithmic advantage is graph is close to planar?
What does close mean?

- **Crossing Number:** min. number of crossings in the plane
- **Skewness:** min. number of edge deletions to obtain planar graph

Non-Planarity Measures

Can we gain an algorithmic advantage is graph is close to planar?
What does close mean?

- **Crossing Number:** min. number of crossings in the plane
- **Skewness:** min. number of edge deletions to obtain planar graph
- **Vertex Deletion Number:** min. number of vertex deletions to obtain planar graph

Non-Planarity Measures

Can we gain an algorithmic advantage is graph is close to planar?
What does close mean?

- **Crossing Number:** min. number of crossings in the plane
- **Skewness:** min. number of edge deletions to obtain planar graph
- **Vertex Deletion Number:** min. number of vertex deletions to obtain planar graph
- **Vertex Splitting Number:** min. number of vertex splits to obtain planar graph

Non-Planarity Measures

Can we gain an algorithmic advantage is graph is close to planar?
What does close mean?

- **Crossing Number:** min. number of crossings in the plane
- **Skewness:** min. number of edge deletions to obtain planar graph
- **Vertex Deletion Number:** min. number of vertex deletions to obtain planar graph
- **Vertex Splitting Number:** min. number of vertex splits to obtain planar graph
- **Thickness:** min. k, s.t. $\exists$ partition of edges into k planar subgraphs

Non-Planarity Measures

Can we gain an algorithmic advantage is graph is close to planar?
What does close mean?

- **Crossing Number:** min. number of crossings in the plane
- **Skewness:** min. number of edge deletions to obtain planar graph
- **Vertex Deletion Number:** min. number of vertex deletions to obtain planar graph
- **Vertex Splitting Number:** min. number of vertex splits to obtain planar graph
- **Thickness:** min. k, s.t. $\exists$ partition of edges into k planar subgraphs
- **Coarseness:** max. number of disjoint Kuratowski-subdivisions

Non-Planarity Measures

Can we gain an algorithmic advantage is graph is close to planar?
What does close mean?

- **Crossing Number:** min. number of crossings in the plane
- **Skewness:** min. number of edge deletions to obtain planar graph
- **Vertex Deletion Number:** min. number of vertex deletions to obtain planar graph
- **Vertex Splitting Number:** min. number of vertex splits to obtain planar graph
- **Thickness:** min. k, s.t. $\exists$ partition of edges into k planar subgraphs
- **Coarseness:** max. number of disjoint Kuratowski-subdivisions
- **Genus:** min. number of handles to embed on an (orientable) surface

Non-Planarity Measures

Can we gain an algorithmic advantage is graph is close to planar?
What does close mean?

- **Crossing Number:** min. number of crossings in the plane
- **Skewness:** min. number of edge deletions to obtain planar graph
- **Vertex Deletion Number:** min. number of vertex deletions to obtain planar graph
- **Vertex Splitting Number:** min. number of vertex splits to obtain planar graph
- **Thickness:** min. k, s.t. $\exists$ partition of edges into k planar subgraphs
- **Coarseness:** max. number of disjoint Kuratowski-subdivisions
- **Genus:** min. number of handles to embed on an (orientable) surface

all of these measures are NP-hard to compute...

Non-Planarity Measures

Can we gain an algorithmic advantage is graph is close to planar?
What does close mean?

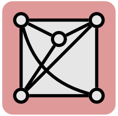
- **Crossing Number:** min. number of crossings in the plane $\mathrm{cr}(G)$
- **Skewness:** min. number of edge deletions to obtain planar graph $\mathrm{sk}(G)$
- **Vertex Deletion Number:** min. number of vertex deletions to obtain planar graph
- **Vertex Splitting Number:** min. number of vertex splits to obtain planar graph
- **Thickness:** min. k, s.t. $\exists$ partition of edges into k planar subgraphs
- **Coarseness:** max. number of disjoint Kuratowski-subdivisions
- **Genus:** min. number of handles to embed on an (orientable) surface $\gamma(G)$

all of these measures are NP-hard to compute...

Relating Measures of Non-Planarity

Lemma

$\mathrm{cr}(G) \geq \mathrm{sk}(G) \geq \gamma(G)$

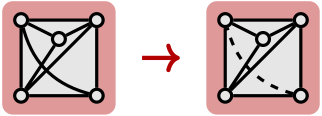


Relating Measures of Non-Planarity

Lemma

$\mathrm{cr}(G) \geq \mathrm{sk}(G) \geq \gamma(G)$

- remove arbitrary edge of each crossing

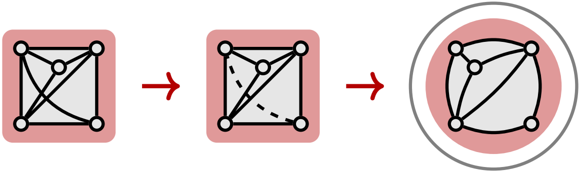


Relating Measures of Non-Planarity

Lemma

$\mathrm{cr}(G) \geq \mathrm{sk}(G) \geq \gamma(G)$

- remove arbitrary edge of each crossing
- route each removed edge over new handle

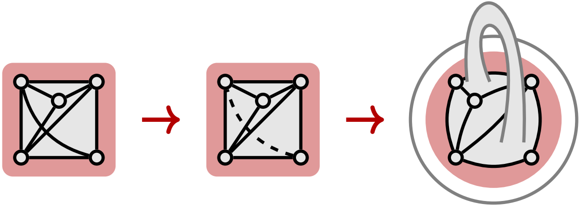


Relating Measures of Non-Planarity

Lemma

$\mathrm{cr}(G) \geq \mathrm{sk}(G) \geq \gamma(G)$

- remove arbitrary edge of each crossing
- route each removed edge over new handle

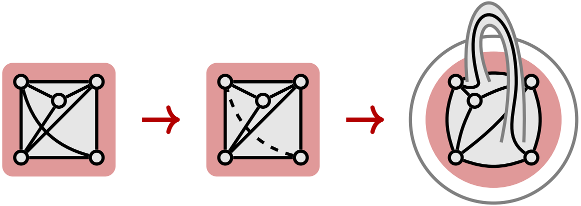


Relating Measures of Non-Planarity

Lemma

$\mathrm{cr}(G) \geq \mathrm{sk}(G) \geq \gamma(G)$

- remove arbitrary edge of each crossing
- route each removed edge over new handle

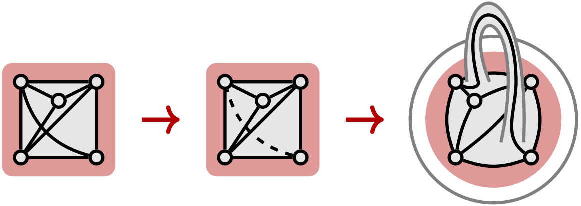


Relating Measures of Non-Planarity

Lemma

$\mathrm{cr}(G) \geq \mathrm{sk}(G) \geq \gamma(G)$

- remove arbitrary edge of each crossing
- route each removed edge over new handle



... and the gaps can be **large**!

Exploiting Low Non-Planarity

$$\{G : \gamma(G) \leq k\} \supset \{G : \mathrm{sk}(G) \leq k\} \supset \{G : \mathrm{cr}(G) \leq k\}$$

Exploiting Low Non-Planarity

$$\{G : \gamma(G) \leq k\} \supset \{G : \mathrm{sk}(G) \leq k\} \supset \{G : \mathrm{cr}(G) \leq k\}$$

Bounded Genus.

- **maaaaaany** publications for many problems!
- P vs. NP, FPT, (better) approximable,...

Exploiting Low Non-Planarity

$$\{G : \gamma(G) \leq k\} \supset \{G : \mathrm{sk}(G) \leq k\} \supset \{G : \mathrm{cr}(G) \leq k\}$$

Bounded Genus.

- **maaaaaany** publications for many problems!
- P vs. NP, FPT, (better) approximable,...

- rich graph classes
- strong theory, but harsh dependency on γ

Exploiting Low Non-Planarity

$$\{G : \gamma(G) \leq k\} \supset \{G : \mathrm{sk}(G) \leq k\} \supset \{G : \mathrm{cr}(G) \leq k\}$$

Bounded Genus.

- **maaaaaany** publications for many problems!
- P vs. NP, FPT, (better) approximable,...

- rich graph classes
- strong theory, but harsh dependency on γ
- computing γ very hard in practice, no heuristics

Exploiting Low Non-Planarity

$$\{G : \gamma(G) \leq k\} \supset \{G : \mathrm{sk}(G) \leq k\} \supset \{G : \mathrm{cr}(G) \leq k\}$$

Bounded Genus.

- **maaaaaany** publications for many problems!
- P vs. NP, FPT, (better) approximable,...

- rich graph classes
- strong theory, but harsh dependency on γ
- computing γ very hard in practice, no heuristics

Bounded Skewness / Crossing Number.

sk MaximumFlow in time-complexity of planar graphs [Hochstein&Weihe 2007]

Exploiting Low Non-Planarity

$$\{G : \gamma(G) \leq k\} \supset \{G : \mathrm{sk}(G) \leq k\} \supset \{G : \mathrm{cr}(G) \leq k\}$$

Bounded Genus.

- **maaaaaany** publications for many problems!
- P vs. NP, FPT, (better) approximable,...

- rich graph classes
- strong theory, but harsh dependency on γ
- computing γ very hard in practice, no heuristics

Bounded Skewness / Crossing Number.

- sk MaximumFlow in time-complexity of planar graphs [Hochstein&Weihe 2007]
- sk constant-factor approximation of crossing number [Ch.&Hliněný 2016]

Exploiting Low Non-Planarity

$$\{G : \gamma(G) \le k\} \supset \{G : \mathrm{sk}(G) \le k\} \supset \{G : \mathrm{cr}(G) \le k\}$$

Bounded Genus.

- **maaaaaany** publications for many problems!
- P vs. NP, FPT, (better) approximable,...

- rich graph classes
- strong theory, but harsh dependency on γ
- computing γ very hard in practice, no heuristics

Bounded Skewness / Crossing Number.

- sk MaximumFlow in time-complexity of planar graphs [Hochstein&Weihe 2007]
- sk constant-factor approximation of crossing number [Ch.&Hliněný 2016]
- cr FPT-parameter for MaxCut [Ch. et al. 2020]

Exploiting Low Non-Planarity

$$\{G : \gamma(G) \leq k\} \supset \{G : \mathrm{sk}(G) \leq k\} \supset \{G : \mathrm{cr}(G) \leq k\}$$

Bounded Genus.

- **maaaaaany** publications for many problems!
- P vs. NP, FPT, (better) approximable,...

- rich graph classes
- strong theory, but harsh dependency on γ
- computing γ very hard in practice, no heuristics

Bounded Skewness / Crossing Number.

- sk MaximumFlow in time-complexity of planar graphs [Hochstein&Weihe 2007]
- sk constant-factor approximation of crossing number [Ch.&Hliněný 2016]
- cr FPT-parameter for MaxCut [Ch. et al. 2020]

- finer-grained parameterization may be more practical

Exploiting Low Non-Planarity

$$\{G : \gamma(G) \leq k\} \supset \{G : \mathrm{sk}(G) \leq k\} \supset \{G : \mathrm{cr}(G) \leq k\}$$

Bounded Genus.

- **maaaaaany** publications for many problems!
- P vs. NP, FPT, (better) approximable,...

- rich graph classes
- strong theory, but harsh dependency on γ
- computing γ very hard in practice, no heuristics

Bounded Skewness / Crossing Number.

- sk MaximumFlow in time-complexity of planar graphs [Hochstein&Weihe 2007]
- sk constant-factor approximation of crossing number [Ch.&Hliněný 2016]
- cr FPT-parameter for MaxCut [Ch. et al. 2020]

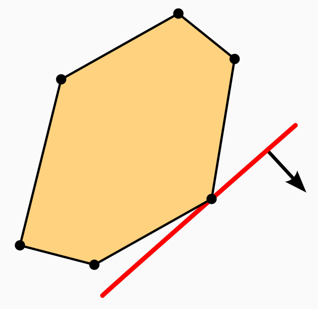
- finer-grained parameterization may be more practical
- more tractable in practice, reasonable exact algorithms & strong heuristics

How to **exactly** compute
$\mathrm{cr}(G)$, $\mathrm{sk}(G)$, and $\gamma(G)$?

Integer Linear Programming

Formulate the problem as an **ILP**

$$
\begin{aligned}
\max \quad & c^\top x && (1)\\
s.t. \quad & Ax \le b && (2)\\
& x \ge 0 && (3)\\
& x \in \mathbb{Z}^n && (4)
\end{aligned}
$$

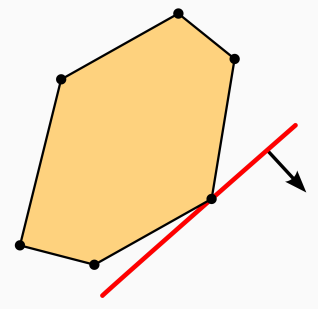


Integer Linear Programming

Formulate the problem as an **ILP**

$$\max \quad c^\top x \tag{1}$$
$$s.t. \quad Ax \leq b \tag{2}$$
$$x \geq 0 \tag{3}$$
$$x \in \mathbb{Z}^n \tag{4}$$

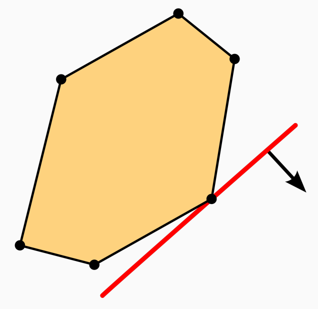
- relaxation can be computed in polynomial time
- fast solvers using branch-and-cut



Integer Linear Programming

Formulate the problem as an **ILP**

$$\begin{aligned} \max \quad & c^\top x && (1)\\ s.t. \quad & Ax \leq b && (2)\\ & x \geq 0 && (3)\\ & x \in \mathbb{Z}^n && (4) \end{aligned}$$



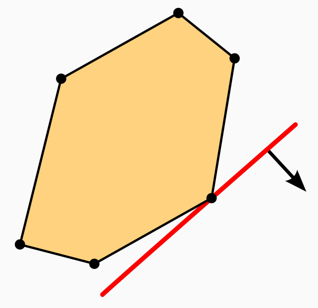
- relaxation can be computed in polynomial time
- fast solvers using branch-and-cut

Only strong formulations yield effective algorithms!

Integer Linear Programming

Formulate the problem as an **ILP**

$$\begin{aligned} \max \quad & c^\top x & (1)\\ s.t. \quad & Ax \leq b & (2)\\ & x \geq 0 & (3)\\ & x \in \mathbb{Z}^n & (4) \end{aligned}$$



- relaxation can be computed in polynomial time
- fast solvers using branch-and-cut

Only strong formulations yield effective algorithms!

Roadmap (of this talk)

Skewness:	Original ILP $\rightarrow$ Improvement by considering Short Cycles
Genus:	Original ILP $\rightarrow$ Improvement by considering Short Cycles

Skewness $\mathbf{sk}(G)$

= Maximum Planar Subgraph (MPS)

Kuratowski-based Formulation

Theorem (Kuratowski, 1930)

G is non-planar $\iff$ G contains a K_5- or $K_{3,3}$-subdivision.

Kuratowski-based Formulation

Theorem (Kuratowski, 1930)

G is non-planar $\iff$ G contains a K_5- or $K_{3,3}$-subdivision.

Kuratowski-based ILP [Mutzel, 1994]

$$\min \sum_{e \in E} s_e$$

$$\text{s.t.} \sum_{e \in K} s_e \geq 1 \qquad \forall \text{ Kuratowski subdivisions } K \subseteq E$$

$$s_e \in \{0, 1\} \qquad \forall e \in E$$

exponentially many

Variable is **1** iff edge e is **deleted**

Kuratowski-based Formulation

Theorem (Kuratowski, 1930)

G is non-planar $\iff$ G contains a K_5- or $K_{3,3}$-subdivision.

Kuratowski-based ILP [Mutzel, 1994]

exponentially many

$$\min \sum_{e \in E} s_e$$

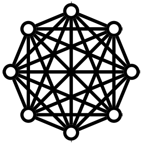
$$s.t. \sum_{e \in K} s_e \geq 1 \qquad \forall \text{ Kuratowski subdivisions } K \subseteq E$$

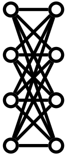
$$s_e \in \{0, 1\} \qquad \forall e \in E$$

Variable is **1** iff edge e is **deleted**

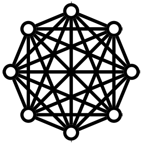
Solve via branch-and-cut with heuristic separation.

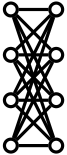
Generalized Euler Constraints





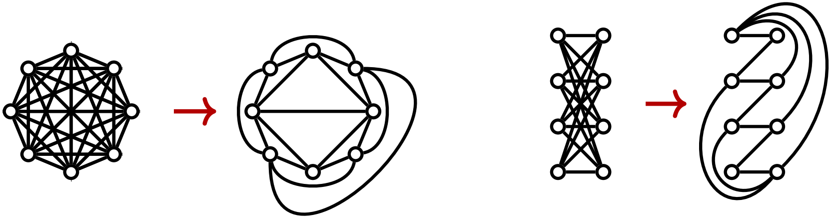
Generalized Euler Constraints





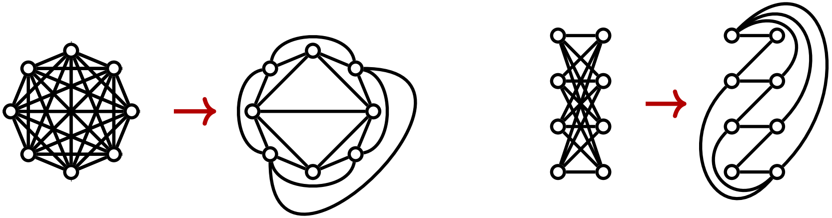
- **Kuratowski** constraints are **weak on dense** graphs

Generalized Euler Constraints



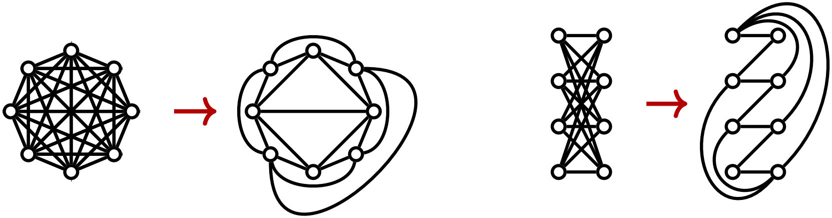
- **Kuratowski** constraints are **weak on dense** graphs

Generalized Euler Constraints



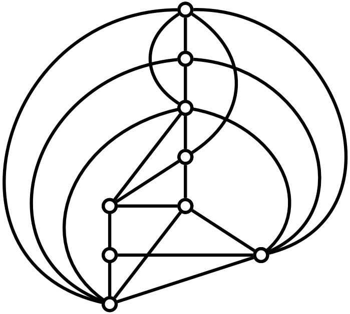
- **Kuratowski** constraints are **weak on dense** graphs
- On complete (complete bipartite) graphs, **optimality** follows directly from **Euler's formula**
  $\Rightarrow$no Kuratowski constraints needed

Generalized Euler Constraints



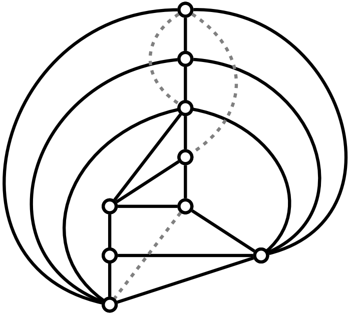
- **Kuratowski** constraints are **weak on dense** graphs
- On complete (complete bipartite) graphs, **optimality** follows directly from **Euler's formula**
  $\Rightarrow$no Kuratowski constraints needed
- Real-world graphs typically **neither** have large girth **nor** is their MPS tri- or quadrangulated...

Key Observations



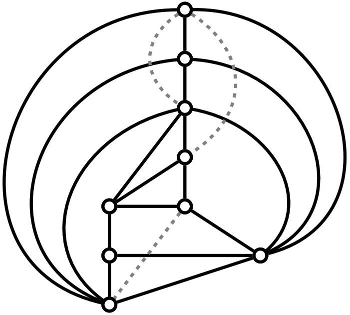
“real-world” graph

Key Observations



MPS of “real-world” graph

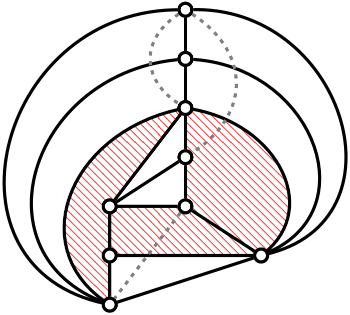
Key Observations



MPS of "real-world" graph

- few **large faces** in MPS (or few **short cycles** in input)

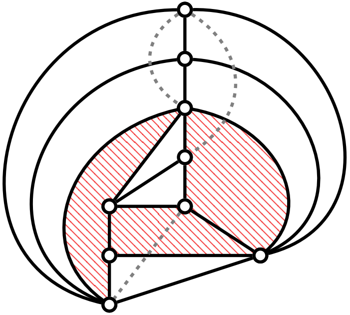
Key Observations



MPS of “real-world” graph

- few **large faces** in MPS (or few **short cycles** in input)

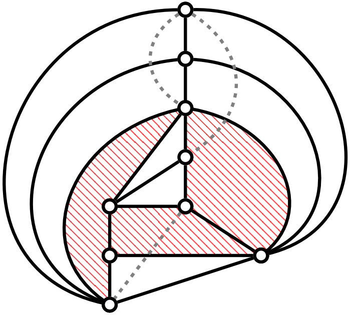
Key Observations



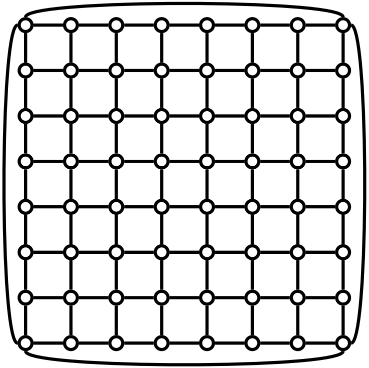
MPS of "real-world" graph

- few **large faces** in MPS (or few **short cycles** in input)
- MPS is typically **biconnected**

Key Observations



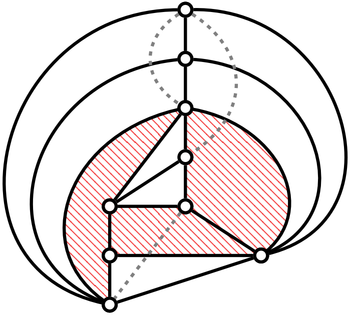
MPS of "real-world" graph



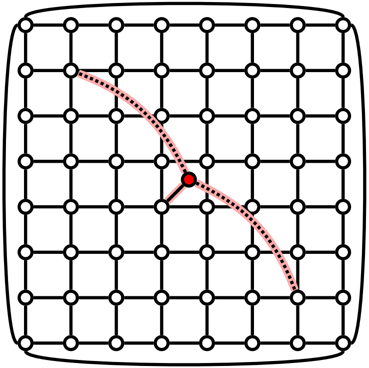
grid-like graph

- few **large faces** in MPS (or few **short cycles** in input)
- MPS is typically **biconnected**

Key Observations



MPS of “real-world” graph



triconnected artificial graph
with no biconnected MPS

- few **large faces** in MPS (or few **short cycles** in input)
- MPS is typically **biconnected**

Idea for Cycle Model

Assume: MPS is biconnected.

$\mathcal{C}_G :=$ set of cycles in input G.

new variable is **1** iff α is a face in the solution

$$c_\alpha \in \{0, 1\} \qquad \forall \alpha \in \mathcal{C}_G$$

Idea for Cycle Model

Assume: MPS is biconnected.

$\mathcal{C}_G :=$ set of cycles in input G.

new variable is **1** iff α is a face in the solution

$$c_\alpha \in \{0,1\} \qquad \forall \alpha \in \mathcal{C}_G$$

$$\sum_{\alpha \in \mathcal{C}_G : e \in \alpha} c_\alpha = 2(1 - s_e) \qquad \forall e \in E$$

Idea for Cycle Model

Assume: MPS is biconnected.

$\mathcal{C}_G :=$ set of cycles in input G.

new variable is **1** iff α is a face in the solution

$$c_\alpha \in \{0,1\} \qquad \forall \alpha \in \mathcal{C}_G$$

$$\sum_{\alpha \in \mathcal{C}_G : e \in \alpha} c_\alpha = 2(1 - s_e) \qquad \forall e \in E$$

$$3|V(G)| - 6 - \sum_{\alpha \in \mathcal{C}_G} (|\alpha| - 3) c_\alpha = \sum_{e \in E} (1 - s_e)$$

Idea for Cycle Model

Assume: MPS is biconnected.
$\mathcal{C}_G :=$ set of cycles in input G.

new variable is **1** iff α is a face in the solution

$$c_\alpha \in \{0,1\} \qquad \forall \alpha \in \mathcal{C}_G$$

$$\sum_{\alpha \in \mathcal{C}_G : e \in \alpha} c_\alpha = 2(1 - s_e) \qquad \forall e \in E$$

$$3|V(G)| - 6 - \sum_{\alpha \in \mathcal{C}_G} (|\alpha| - 3) c_\alpha = \sum_{e \in E} (1 - s_e)$$

Problems:

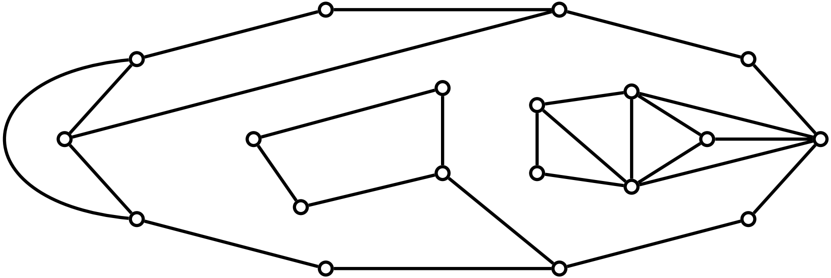
- We cannot enumerate **all** cycles in practice
- MPS is not biconnected

Obstacle: Non-Biconnectedness

Lemma

(MPS will not be outerplanar.) To each a face-walk β we can associate a sub-cycle $\alpha \subseteq \beta$ such that each cycle is used at most once.

- faces in the solution $\overset{1:1}{\longleftrightarrow}$ cycles
- each edge is incident to **at most** one such cycle

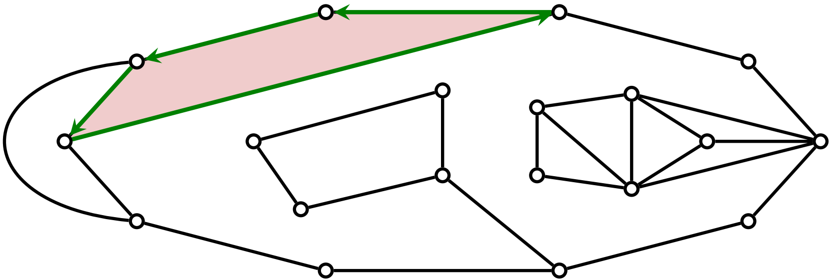


Obstacle: Non-Biconnectedness

Lemma

(MPS will not be outerplanar.) To each a face-walk β we can associate a sub-cycle $\alpha \subseteq \beta$ such that each cycle is used at most once.

- faces in the solution $\overset{1:1}{\longleftrightarrow}$ cycles
- each edge is incident to **at most** one such cycle

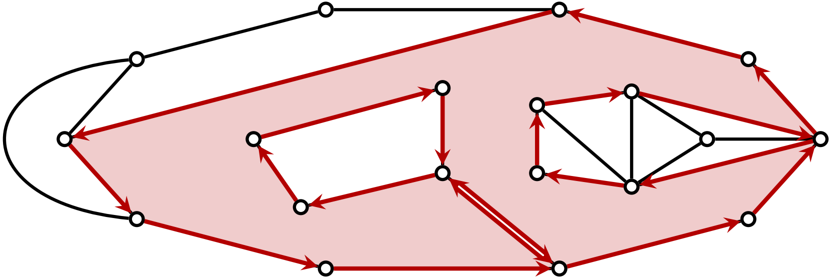


Obstacle: Non-Biconnectedness

Lemma

(MPS will not be outerplanar.) To each a face-walk β *we can associate a sub-cycle* $\alpha \subseteq \beta$ *such that each cycle is used at most once.*

- faces in the solution $\overset{1:1}{\longleftrightarrow}$ cycles
- each edge is incident to **at most** one such cycle

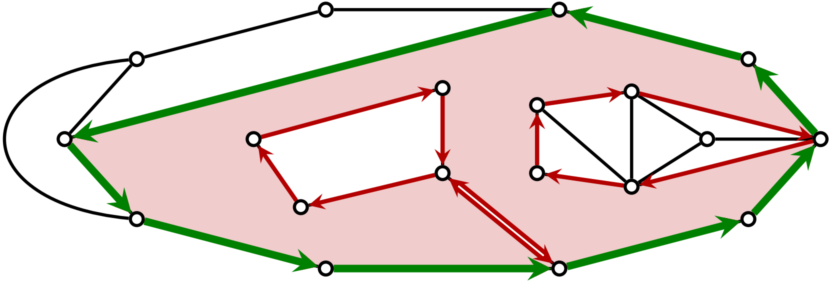


Obstacle: Non-Biconnectedness

Lemma

(MPS will not be outerplanar.) To each a face-walk β we can associate a sub-cycle $\alpha \subseteq \beta$ such that each cycle is used at most once.

- faces in the solution $\overset{1:1}{\longleftrightarrow}$ cycles
- each edge is incident to **at most** one such cycle



Cycle Model

$$c_\alpha \in \{0,1\} \qquad \forall \alpha \in \mathcal{C}_G$$

$$\sum_{\alpha \in \mathcal{C}_G :\ e \in \alpha} c_\alpha = 2(1 - s_e) \qquad \forall e \in E$$

$$3|V(G)| - 6 - \sum_{\alpha \in \mathcal{C}_G} (|\alpha| - 3) c_\alpha = \sum_{e \in E} (1 - s_e)$$

Cycle Model

$$c_\alpha \in \{0,1\} \qquad \forall \alpha \in \mathcal{C}_G$$

$$\sum_{\alpha \in \mathcal{C}_G :\ e \in \alpha} c_\alpha \leq 2(1 - s_e) \qquad \forall e \in E$$

$$3|V(G)| - 6 - \sum_{\alpha \in \mathcal{C}_G} (|\alpha| - 3) c_\alpha = \sum_{e \in E} (1 - s_e)$$

Cycle Model

$$c_\alpha \in \{0,1\} \qquad \forall \alpha \in \mathcal{C}_G$$

$$\sum_{\alpha \in \mathcal{C}_G :\, e \in \alpha} c_\alpha \leq 2(1 - s_e) \qquad \forall e \in E$$

$$3|V(G)| - 6 - \sum_{\alpha \in \mathcal{C}_G} (|\alpha| - 3) c_\alpha = \sum_{e \in E} (1 - s_e)$$

Cycle Model

Pick $D \in \mathbb{N}$ such that number of variables is “reasonable”.

$$c_\alpha \in \{0,1\} \qquad \forall \alpha \in \mathcal{C}_G, |\alpha| \leq D$$

$$\sum_{\alpha \in \mathcal{C}_G:\ e \in \alpha, |\alpha| \leq D} c_\alpha \leq 2(1 - s_e) \qquad \forall e \in E$$

~~$$3|V(G)| - 6 - \sum_{\alpha \in \mathcal{C}_G} (|\alpha| - 3) c_\alpha = \sum_{e \in E} (1 - s_e)$$~~

Cycle Model

Pick $D \in \mathbb{N}$ such that number of variables is "reasonable".

$$c_\alpha \in \{0,1\} \qquad \forall \alpha \in \mathcal{C}_G, |\alpha| \leq D$$

$$\sum_{\alpha \in \mathcal{C}_G:\ e \in \alpha, |\alpha| \leq D} c_\alpha \leq 2(1 - s_e) \qquad \forall e \in E$$

~~$$3|V(G)| - 6 - \sum_{\alpha \in \mathcal{C}_G} (|\alpha| - 3) c_\alpha = \sum_{e \in E} (1 - s_e)$$~~

$$(D+1)(|V(G)| - 2) + \sum_{d=3,4,\ldots,D} (D + 1 - d) \sum_{\alpha \in \mathcal{C}_G^{=d}} c_\alpha \geq (D-1) \sum_{e \in E} (1 - s_e)$$

D-Hierarchy

D-Hierarchy

Theorem

The ILP formulations become strictly stronger when increasing D (but the number of variables increases drastically, as well).

D-Hierarchy

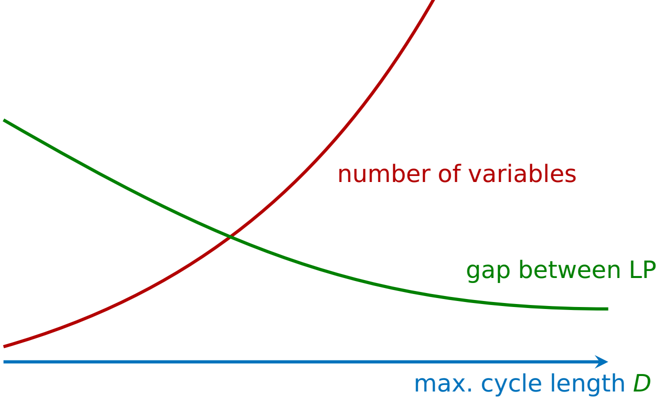
Theorem

The ILP formulations become strictly stronger when increasing D (but the number of variables increases drastically, as well).

D-Hierarchy

Theorem

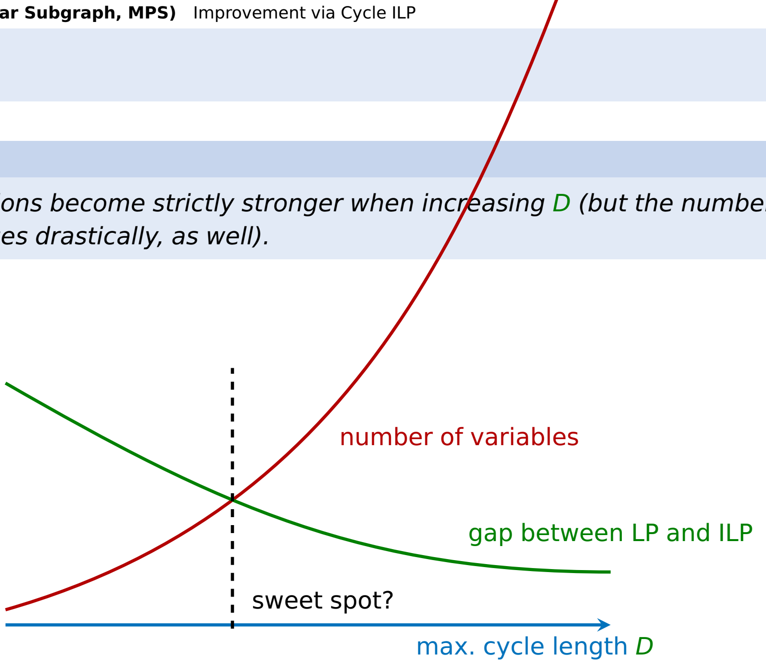
The ILP formulations become strictly stronger when increasing D (but the number of variables increases drastically, as well).



D-Hierarchy

Theorem

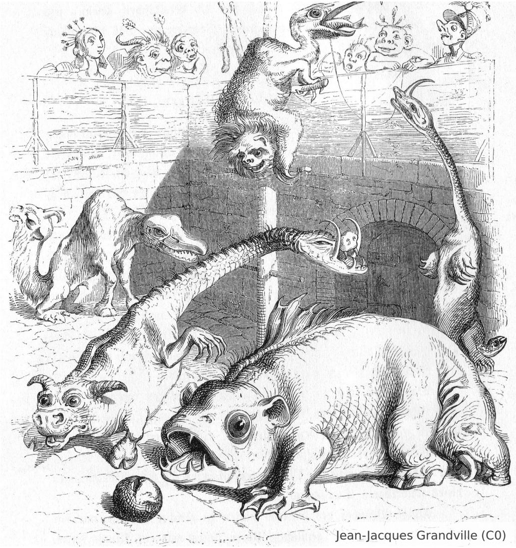
The ILP formulations become strictly stronger when increasing D (but the number of variables increases drastically, as well).



Supplemental Constraint Zoo

Once you have cycle variables,
you can do a lot more with them!

Goal: Tightly link edge- and cycle-variables



Jean-Jacques Grandville (C0)

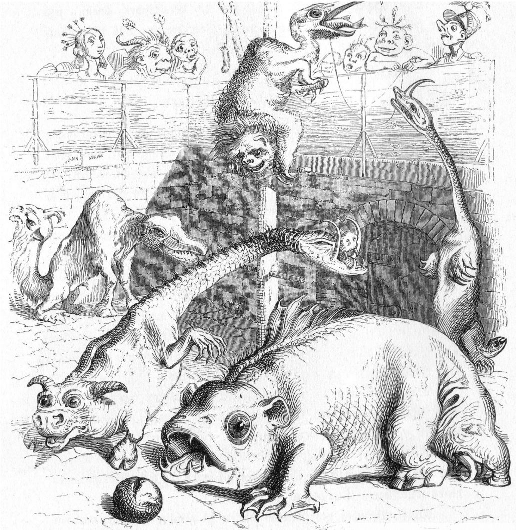
Supplemental Constraint Zoo

Once you have cycle variables,
you can do a lot more with them!

Goal: Tightly link edge- and cycle-variables

For example:

- pseudo-tree extension
- cycle-edge cons.
- Kuratowski-cycle cons.
- k-cycles-path cons.
- cycle-clique cons.
- cycle-two-paths cons.



Jean-Jacques Grandville (C0)

Experimental Setting

Framework

- C++, GCC 6.3
- OGDF 2018.03 [www.ogdf.net]
- SCIP 6.0 [scip.zib.de]
- CPLEX 12.8

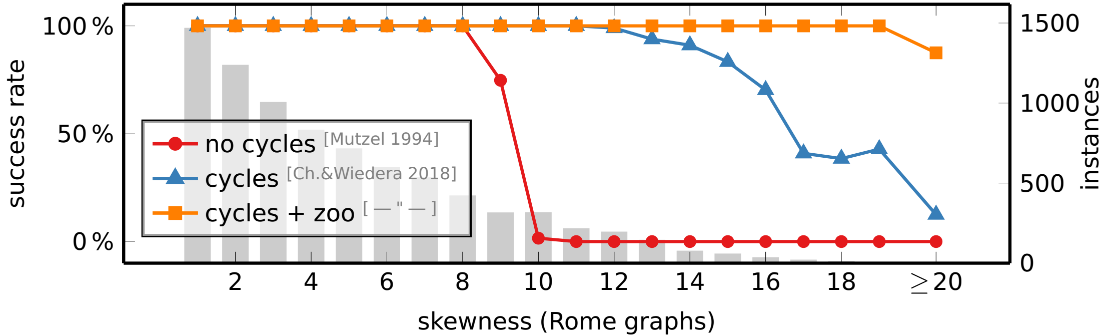
Computations

- Xeon Gold 6134
- limit: 20 minutes, 8 GB

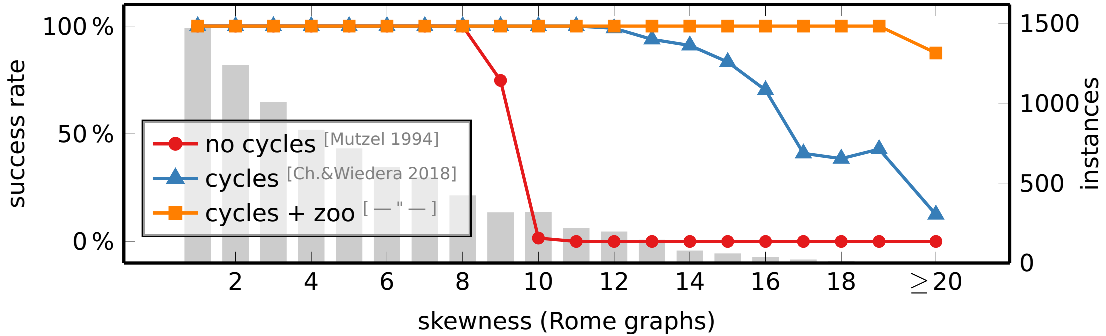
Instances

- Rome [Di Battista et al., 1995]
- Expander [Steger & Wormald, 1997]
- North [North, 1995]
- SteinLib [Koch et al., 2000]

Experimental Evaluation

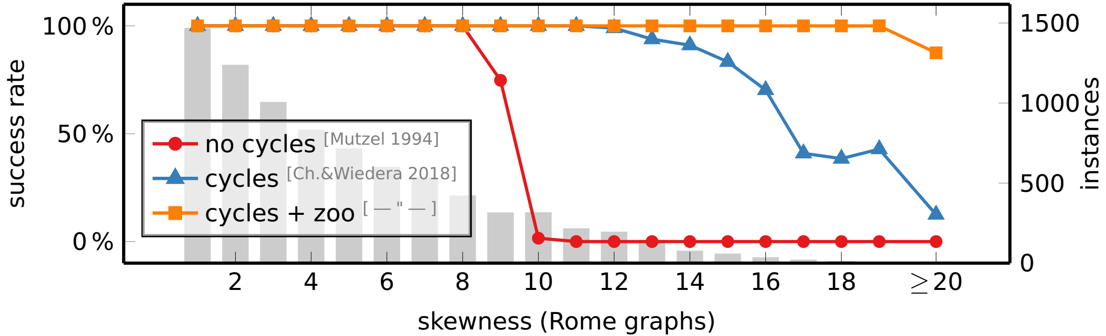


Experimental Evaluation



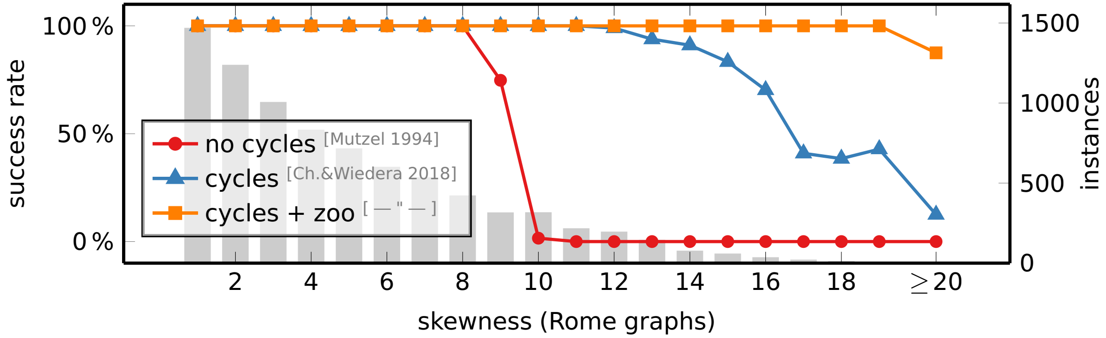
Without vs. with cycles (+zoo)	Rome	North	Expander	SteinLib
% solved of previously unsolved	“100%”	75%	30%	30%

Experimental Evaluation



Without vs. with cycles (+zoo)	Rome	North	Expander	SteinLib
% solved of previously unsolved	“100%”	75%	30%	30%
speed-up (avg) on commonly solved	66x	34x	20x	12x

Experimental Evaluation



Without vs. with cycles (+zoo)	Rome	North	Expander	SteinLib
% solved of previously unsolved	"100%"	75%	30%	30%
speed-up (avg) on commonly solved	66x	34x	20x	12x
D (raise until ≥1000 vars)	11	8	6	7

Genus $\gamma(G)$

Genus $\gamma(G)$

computing it is quite hard practice...

- $\exists$ linear time FPT-algorithm for bounded genus g [Mohar 1999], but **doubly exponential** in g and no known implementation even for toroidal case
- no (reasonable) heuristics

Faces in Embeddings of Higher Genus

Theorem (Euler's formula)

... is sensitive to the genus of the surface on with the drawing is: $n - m + f = 2 - 2\gamma$.

Faces in Embeddings of Higher Genus

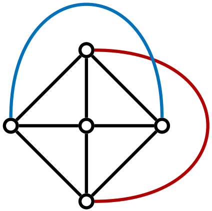
Theorem (Euler's formula)

... is sensitive to the genus of the surface on with the drawing is: $n - m + f = 2 - 2\gamma$.

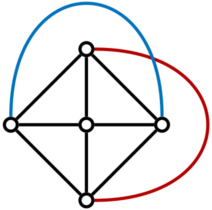
Face-Tracing-based ILP [Beyer et al. 2016]

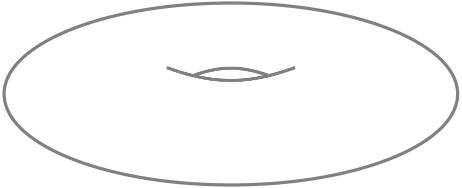
- Minimize γ by finding an embedding ($=$ rotation system) that **maximizes** f.
- Count faces via **face tracing**.

Crossing-free Non-planar Graphs

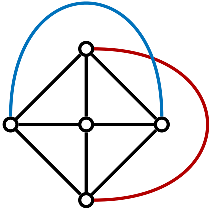


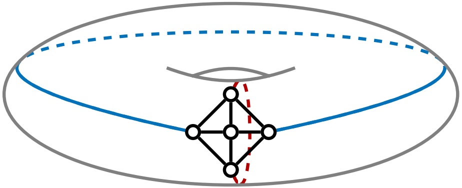
Crossing-free Non-planar Graphs



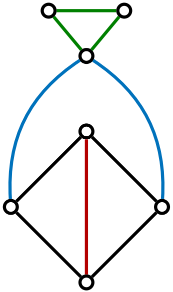


Crossing-free Non-planar Graphs



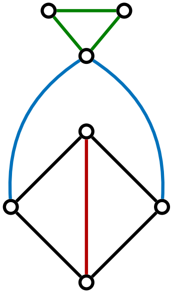


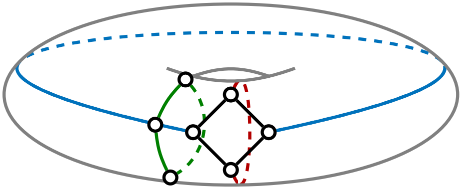
Embeddings and Face Tracing



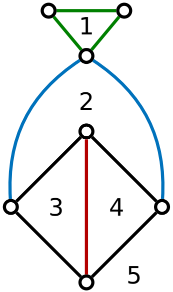
Embeddings and Face Tracing

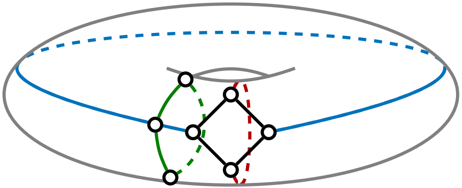
Embeddings and Face Tracing



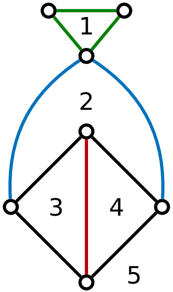


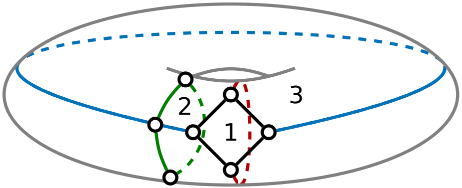
Embeddings and Face Tracing





Embeddings and Face Tracing





Face-Tracing-based Formulation

upper bound on number of faces: $\bar{f} := \min\{m - n, 2m/3\}$

Face-Tracing-based Formulation

upper bound on number of faces: $\bar{f} := \min\{m - n, 2m/3\}$

$$\max \sum_{i \in [\bar{f}]} x_i$$

Use the i-th face?	$x_i \in \{0, 1\}$	$\forall i \in [\bar{f}]$
Arc a on i-th face?	$x_i^a \in \{0, 1\}$	$\forall i \in [\bar{f}], a \in A$

Face-Tracing-based Formulation

upper bound on number of faces: $\bar{f} := \min\{m-n, 2m/3\}$

$$\max \textstyle\sum_{i\in[\bar{f}]} x_i$$

s.t. $3\,x_i \leq \sum_{a\in A} x_i^a$

Use the *i*-th face? $x_i \in \{0,1\}$ $\forall i \in [\bar{f}]$

Arc *a* on *i*-th face? $x_i^a \in \{0,1\}$ $\forall i \in [\bar{f}], a \in A$

Face-Tracing-based Formulation

upper bound on number of faces: $\bar{f} := \min\{m - n, 2m/3\}$

$$\max \sum_{i \in [\bar{f}]} x_i$$

s.t.

$$3\, x_i \leq \sum_{a \in A} x_i^a$$

$$\sum_{i \in [\bar{f}]} x_i^a = 1 \qquad \forall a \in A$$

Use the i-th face? $\quad x_i \in \{0, 1\} \qquad \forall i \in [\bar{f}]$

Arc a on i-th face? $\quad x_i^a \in \{0, 1\} \qquad \forall i \in [\bar{f}], a \in A$

Face-Tracing-based Formulation

upper bound on number of faces: $\bar{f} := \min\{m - n, 2m/3\}$

$$\max \sum_{i \in [\bar{f}]} x_i$$

s.t.

$$3\,x_i \leq \sum_{a \in A} x_i^a$$

$$\sum_{i \in [\bar{f}]} x_i^a = 1 \qquad \forall a \in A$$

$$\sum_{a \in \delta^+(v)} x_i^a = \sum_{a \in \delta^-(v)} x_i^a \qquad \forall i \in [\bar{f}],\ v \in V$$

Use the i-th face? $x_i \in \{0, 1\}$ $\forall i \in [\bar{f}]$

Arc a on i-th face? $x_i^a \in \{0, 1\}$ $\forall i \in [\bar{f}], a \in A$

Face-Tracing-based Formulation

upper bound on number of faces: $\bar{f} := \min\{m - n, 2m/3\}$

$$\max \sum_{i \in [\bar{f}]} x_i$$

s.t.

$$3\, x_i \leq \sum_{a \in A} x_i^a$$

$$\sum_{i \in [\bar{f}]} x_i^a = 1 \qquad \forall a \in A$$

$$\sum_{a \in \delta^+(v)} x_i^a = \sum_{a \in \delta^-(v)} x_i^a \qquad \forall i \in [\bar{f}],\ v \in V$$

[+ simulate face tracing via [Beyer et al. 2016] or [Ch.&Wiedera 2019]]

Use the *i*-th face? $x_i \in \{0, 1\}$ $\forall i \in [\bar{f}]$

Arc *a* on *i*-th face? $x_i^a \in \{0, 1\}$ $\forall i \in [\bar{f}], a \in A$

Face-Tracing-based Formulation with small faces

upper bound on number of large faces: $\bar{f} := \min\{m - n, 2m/(D+1)\}$
all **walks** of length $\leq D$: $\mathcal{F}_D$

$$\max \textstyle\sum_{i \in [\bar{f}]} x_i$$

s.t.

$$3\,x_i \leq \textstyle\sum_{a \in A} x_i^a$$

$$\textstyle\sum_{i \in [\bar{f}]} x_i^a = 1 \qquad \forall a \in A$$

$$\textstyle\sum_{a \in \delta^+(v)} x_i^a = \sum_{a \in \delta^-(v)} x_i^a \qquad \forall i \in [\bar{f}],\ v \in V$$

[+ simulate face tracing via [Beyer et al. 2016] or [Ch.&Wiedera 2019]]

Use the i-th large face? $\quad x_i \in \{0,1\} \quad \forall i \in [\bar{f}]$

Arc a on i-th face? $\quad x_i^a \in \{0,1\} \quad \forall i \in [\bar{f}], a \in A$

Use small face α? $\quad c_\alpha \in \{0,1\} \quad \forall \alpha \in \mathcal{F}_D$

Face-Tracing-based Formulation with small faces

upper bound on number of large faces: $\bar{f} := \min\{m - n, 2m/(D+1)\}$
all **walks** of length $\leq D$: $\mathcal{F}_D$

$$\max \sum_{i \in [\bar{f}]} x_i + \sum_{\alpha \in \mathcal{F}_D} c_\alpha$$

s.t.

$$(D+1)\, x_i \leq \sum_{a \in A} x_i^a$$

$$\sum_{i \in [\bar{f}]} x_i^a + \sum_{\alpha \in \mathcal{F}_D : a \in \alpha} c_\alpha = 1 \qquad \forall a \in A$$

$$\sum_{a \in \delta^+(v)} x_i^a = \sum_{a \in \delta^-(v)} x_i^a \qquad \forall i \in [\bar{f}],\ v \in V$$

[+ simulate face tracing via [Beyer et al. 2016] or [Ch.&Wiedera 2019]]

Use the i-th large face? $x_i \in \{0,1\}$ $\forall i \in [\bar{f}]$

Arc a on i-th face? $x_i^a \in \{0,1\}$ $\forall i \in [\bar{f}], a \in A$

Use small face α? $c_\alpha \in \{0,1\}$ $\forall \alpha \in \mathcal{F}_D$

also needs amendment, no details

Hierarchie and Zoo

Analogous to before:

Theorem

The ILP formulations become strictly stronger when increasing D (but the number of variables increases drastically, as well).

Hierarchie and Zoo

Analogous to before:

Theorem

The ILP formulations become strictly stronger when increasing D (but the number of variables increases drastically, as well).

Now that we have cycle variables, we can again give additional add-ons to bind them tigher with the face-tracing (but fewer options than for skewness; no details now).

Experimental Evaluation

No cycles. [Beyer et al. 2016]

- First (at least somewhat) practical approach, but:
- only able to solve real-world graphs with $\gamma(G) = 1$.

Experimental Evaluation

No cycles. [Beyer et al. 2016]

- First (at least somewhat) practical approach, but:
- only able to solve real-world graphs with $\gamma(G) = 1$.

With cycles (and little zoo). [Ch.&Wiedera 2019]

- solves instances with up to $\gamma(G) = 10$

Experimental Evaluation

No cycles. [Beyer et al. 2016]

- First (at least somewhat) practical approach, but:
- only able to solve real-world graphs with $\gamma(G) = 1$.

With cycles (and little zoo). [Ch.&Wiedera 2019]

- solves instances with up to $\gamma(G) = 10$
- increase in success rate: Rome $28\% \to 82\%$
  North $48\% \to 76\%$
  Expander $5\% \to 24\%$

Experimental Evaluation

No cycles. [Beyer et al. 2016]

- First (at least somewhat) practical approach, but:
- only able to solve real-world graphs with $\gamma(G) = 1$.

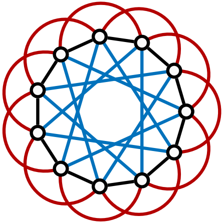
With cycles (and little zoo). [Ch.&Wiedera 2019]

- solves instances with up to $\gamma(G) = 10$
- increase in success rate:
  Rome 28% $\to$ 82%
  North 48% $\to$ 76%
  Expander 5% $\to$ 24%
- speed-up on commonly solved:
  Rome 200x
  North 100x
  Expander 80x

Evaluation: Genera from Literature

The cycle-based ILP can confirm results from literature (all with non-trivial dual bounds):

- **Circulants with genus** ≤ 2 [Conder&Grande 2015]
  hardest case: $C_{11}(1, 2, 4)$
  3 pages analysis, 85h $\Longrightarrow$ 180h [Beyer et al. 2016] $\Longrightarrow$ **10s**

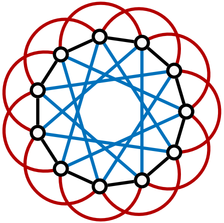


Evaluation: Genera from Literature

The cycle-based ILP can confirm results from literature (all with non-trivial dual bounds):

- **Circulants with genus** ≤ 2 [Conder&Grande 2015]
  hardest case: $C_{11}(1, 2, 4)$
  3 pages analysis, 85h $\Longrightarrow$ 180h [Beyer et al. 2016] $\Longrightarrow$ **10s**

- **Gray graph has genus** 7 [Marusic et al. 2005]
  full paper $\Longrightarrow$ 42 hours



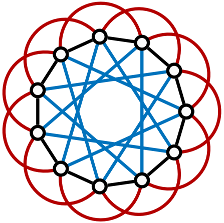
Evaluation: Genera from Literature

The cycle-based ILP can confirm results from literature (all with non-trivial dual bounds):

- **Circulants with genus** ≤ 2 [Conder&Grande 2015]
  hardest case: $C_{11}(1, 2, 4)$
  3 pages analysis, 85h $\Longrightarrow$ 180h [Beyer et al. 2016] $\Longrightarrow$ **10s**

- **Gray graph has genus** 7 [Marusic et al. 2005]
  full paper $\Longrightarrow$ 42 hours

- $\mathbb{Z}_9 \times \mathbb{Z}_3$ **has genus** 4 [Brin et al. 1989]
  full paper $\Longrightarrow$ 5 minutes



Conclusion

- We know for a long time that non-planarity measures are dependent on the graph’s girth… but short cycles are underused in current algorithmics!

Conclusion

- We know for a long time that non-planarity measures are dependent on the graph's girth... but short cycles are underused in current algorithmics!
- Don't be afraid of the fact that there are exponentially many cycles in a graph! Already considering only **short** ones can help!

Conclusion

- We know for a long time that non-planarity measures are dependent on the graph's girth... but short cycles are underused in current algorithmics!
- Don't be afraid of the fact that there are exponentially many cycles in a graph! Already considering only **short** ones can help!

- One may also consider cycles in the dual!

Conclusion

- We know for a long time that non-planarity measures are dependent on the graph's girth... but short cycles are underused in current algorithmics!
- Don't be afraid of the fact that there are exponentially many cycles in a graph! Already considering only **short** ones can help!

- One may also consider cycles in the dual!
- Can something similar can be done for the crossing number?

The famous **Crossing Lemma** $\mathrm{cr}(G) = \Omega(m^3/n^2)$ can be made girth-aware:
$\mathrm{cr}(G) = \Omega(m^{r+2}/n^{r+1})$ for girth $>2r$ [Pach et al. 2000]

Conclusion

- We know for a long time that non-planarity measures are dependent on the graph's girth... but short cycles are underused in current algorithmics!
- Don't be afraid of the fact that there are exponentially many cycles in a graph! Already considering only **short** ones can help!

- One may also consider cycles in the dual!
- Can something similar can be done for the crossing number?

  The famous **Crossing Lemma** $\mathrm{cr}(G) = \Omega(m^3/n^2)$ can be made girth-aware: $\mathrm{cr}(G) = \Omega(m^{r+2}/n^{r+1})$ for girth $>2r$ [Pach et al. 2000]
- Many opportunities!

Conclusion

- We know for a long time that non-planarity measures are dependent on the graph's girth... but short cycles are underused in current algorithmics!
- Don't be afraid of the fact that there are exponentially many cycles in a graph! Already considering only **short** ones can help!

- One may also consider cycles in the dual!
- Can something similar can be done for the crossing number?

  The famous **Crossing Lemma** $\mathrm{cr}(G) = \Omega(m^3/n^2)$ can be made girth-aware: $\mathrm{cr}(G) = \Omega(m^{r+2}/n^{r+1})$ for girth $>2r$ [Pach et al. 2000]

- Many opportunities!

Thank you and good health!

In particular also to the organizers Sascha, Steven and Philipp!