Generating p-extremal graphs

Derrick Stolee

Iowa State University
dstolee@iastate.edu
http://www.math.iastate.edu/dstolee/

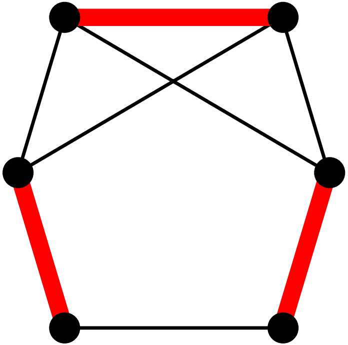
December 9, 2013
ISU MECS Seminar

Last Week

1. Discussed generation of combinatorial objects.
2. “Defined” symmetry in terms of automorphism groups.
3. Presented **canonical deletion**, a method to remove isomorphic duplicates.
4. Discussed example for generating connected graphs by vertex augmentations.

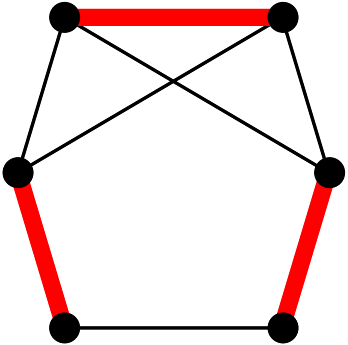
Perfect Matchings

A **perfect matching** is a set of edges which cover each vertex exactly once.



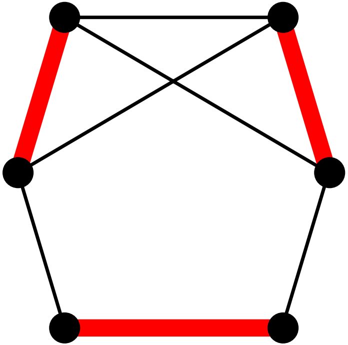
Perfect Matchings

A **perfect matching** is a set of edges which cover each vertex exactly once.
$\Phi(G)$ is the number of perfect matchings in the graph G.



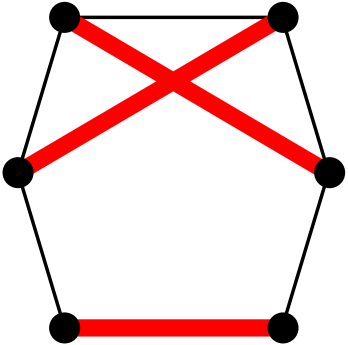
Perfect Matchings

A **perfect matching** is a set of edges which cover each vertex exactly once.
$\Phi(G)$ is the number of perfect matchings in the graph G.



Perfect Matchings

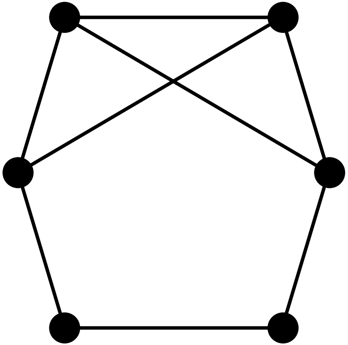
A **perfect matching** is a set of edges which cover each vertex exactly once.
$\Phi(G)$ is the number of perfect matchings in the graph G.



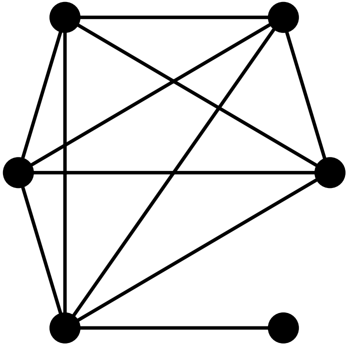
$\Phi(G) = 3$
8 edges

Perfect Matchings

A **perfect matching** is a set of edges which cover each vertex exactly once.



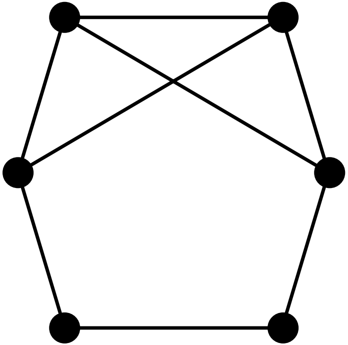
$\Phi(G) = 3$
8 edges



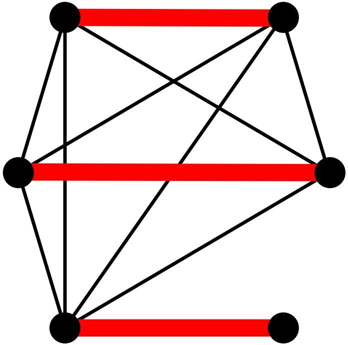
11 edges

Perfect Matchings

A **perfect matching** is a set of edges which cover each vertex exactly once.



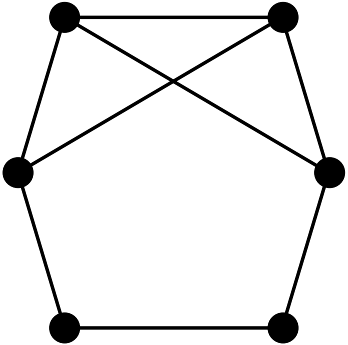
$\Phi(G) = 3$
8 edges



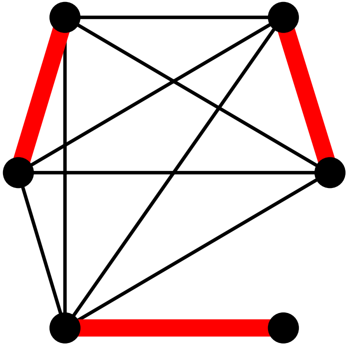
11 edges

Perfect Matchings

A **perfect matching** is a set of edges which cover each vertex exactly once.



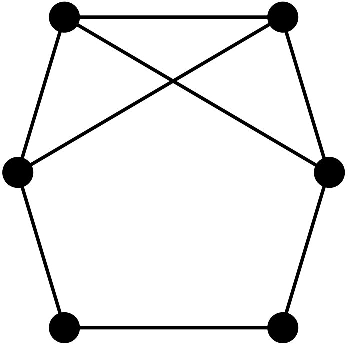
$\Phi(G) = 3$
8 edges



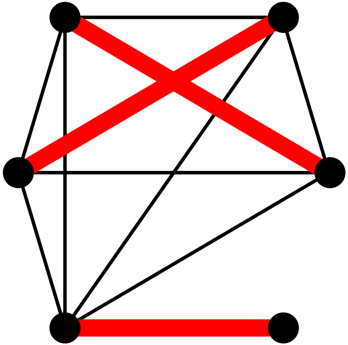
11 edges

Perfect Matchings

A **perfect matching** is a set of edges which cover each vertex exactly once.



$\Phi(G) = 3$
8 edges



$\Phi(G) = 3$
11 edges

Perfect Matchings

A **perfect matching** is a set of edges which cover each vertex exactly once.

Question (Dudek, Schmitt, '12) What is the maximum number of edges in a graph with exactly n vertices and p perfect matchings?

Definition Let n be an even number and fix $p \geq 1$.

$$f(n,p) = \max\{|E(G)| : |V(G)| = n, \Phi(G) = p\}.$$

Graphs attaining this number of edges are ***p*-extremal**.

Hetyei's Theorem

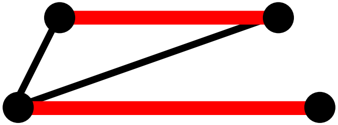
Theorem (Hetyei's Theorem, 1986) For all even $n \geq 2$,

$$f(n, 1) = \frac{n^2}{4}.$$

Hetyei's Theorem

Theorem (Hetyei's Theorem, 1986) For all even $n \geq 2$,

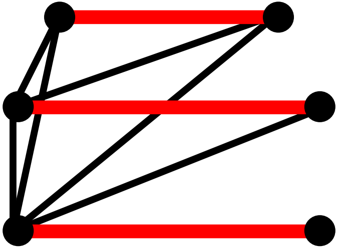
$$f(n, 1) = \frac{n^2}{4}.$$



Hetyei's Theorem

Theorem (Hetyei's Theorem, 1986) For all even $n \geq 2$,

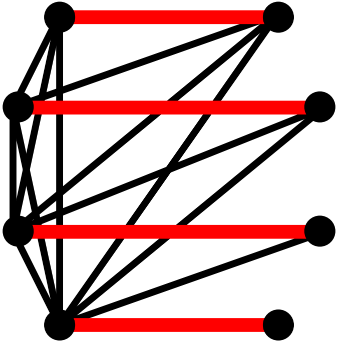
$$f(n,1) = \frac{n^2}{4}.$$



Hetyei's Theorem

Theorem (Hetyei's Theorem, 1986) For all even $n \geq 2$,

$$f(n, 1) = \frac{n^2}{4}.$$

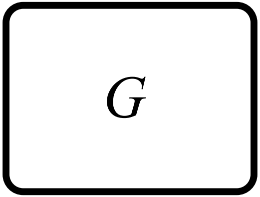


The Form of $f(n, p)$

Theorem (Dudek & Schmitt)

For each p, there exist constants n_p, c_p so that for all $n \geq n_p$,

$$f(n, p) = \frac{n^2}{4} + c_p.$$



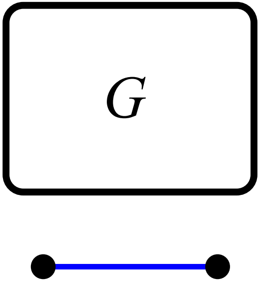
Take G with $\frac{n^2}{4} + c$ edges.

The Form of $f(n,p)$

Theorem (Dudek & Schmitt)

For each p, there exist constants n_p, c_p so that for all $n \geq n_p$,

$$f(n,p) = \frac{n^2}{4} + c_p.$$



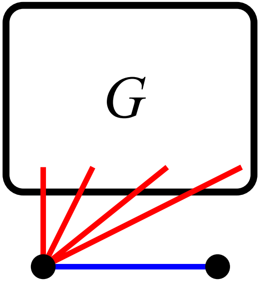
Add two new vertices.

The Form of $f(n, p)$

Theorem (Dudek & Schmitt)

For each p, there exist constants n_p, c_p so that for all $n \geq n_p$,

$$f(n, p) = \frac{n^2}{4} + c_p.$$



Add edges to get $\frac{(n+2)^2}{4} + c$ edges.

The Excess of a Graph

Let $\Phi(G) > 0$. The **excess** $c(G)$ is

$$c(G) = |E(G)| - \frac{|V(G)|^2}{4}.$$

The Excess of a Graph

Let $\Phi(G) > 0$. The **excess** $c(G)$ is

$$c(G) = |E(G)| - \frac{|V(G)|^2}{4}.$$

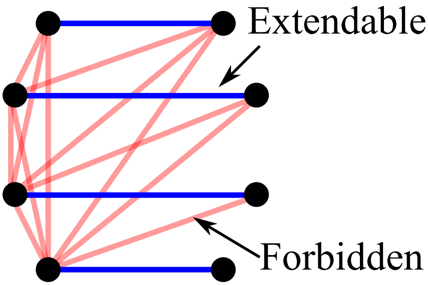
In this sense, lower bounds on c_p are "easy"
(any G with $\Phi(G) = p$, has $c(G) \leq c_p$).

Upper bounds are hard: must prove NO graph achieves a higher constant!

Edge Types

Let $\Phi(G) > 0$ and $e \in E(G)$.

- e is **extendable** if there exists a perfect matching containing e.
- e is **forbidden** otherwise.



Types of Graphs

Let G be connected with $\Phi(G) > 0$.

Types of Graphs

Let G be connected with $\Phi(G) > 0$.

- G is **extendable** if all edges are extendable.

Types of Graphs

Let G be connected with $\Phi(G) > 0$.

- G is **extendable** if all edges are extendable.
- G is a **chamber** if the set of extendable edges forms a connected (spanning) subgraph.

Types of Graphs

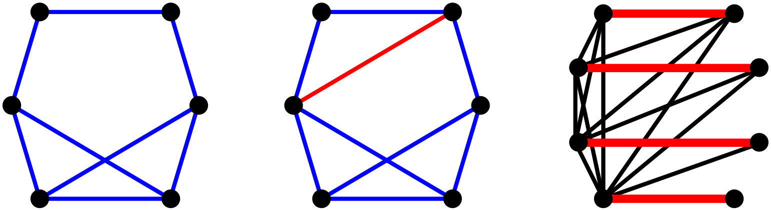
Let G be connected with $\Phi(G) > 0$.

- G is **extendable** if all edges are extendable.
- G is a **chamber** if the set of extendable edges forms a connected (spanning) subgraph.
- G is ***p*-extremal** if $\Phi(G) = p$ and $c(G) = c_p$.

Types of Graphs

Let G be connected with $\Phi(G) > 0$.

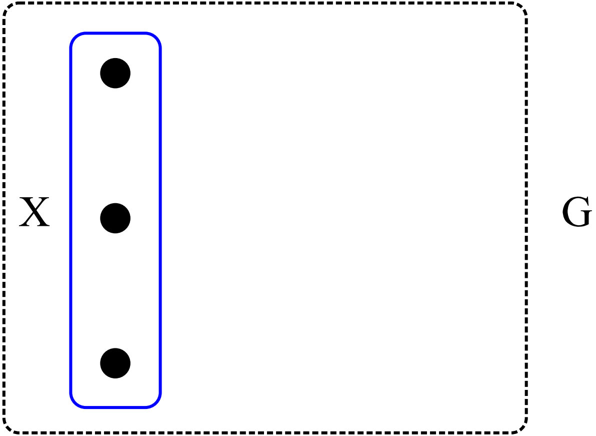
- G is **extendable** if all edges are extendable.
- G is a **chamber** if the set of extendable edges forms a connected (spanning) subgraph.
- G is ***p*-extremal** if $\Phi(G) = p$ and $c(G) = c_p$.



Barriers

Let $\Phi(G) > 0$. A set $X \subset V(G)$ is a **barrier** if

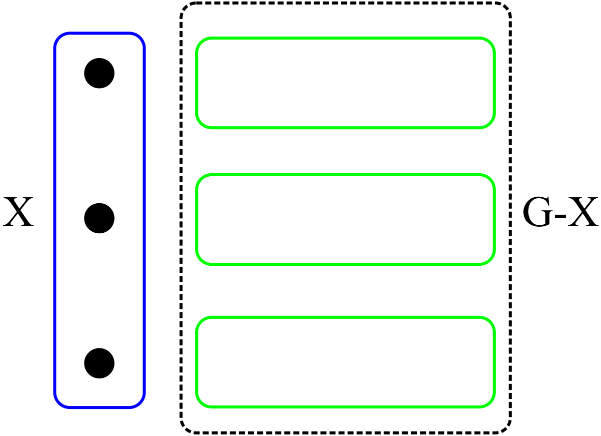
$$|X| = \text{\# of odd connected components in } G - X$$



Barriers

Let $\Phi(G) > 0$. A set $X \subset V(G)$ is a **barrier** if

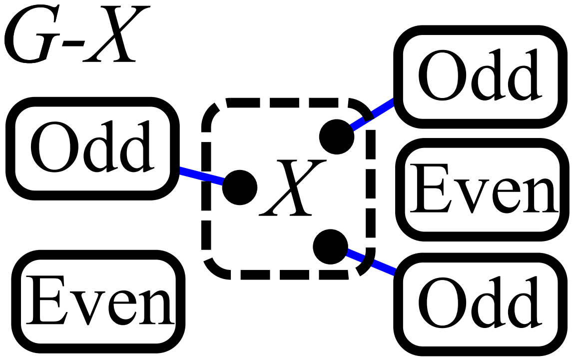
$$|X| = \text{\# of odd connected components in } G - X$$



Barriers

Let $\Phi(G) > 0$. A set $X \subset V(G)$ is a **barrier** if

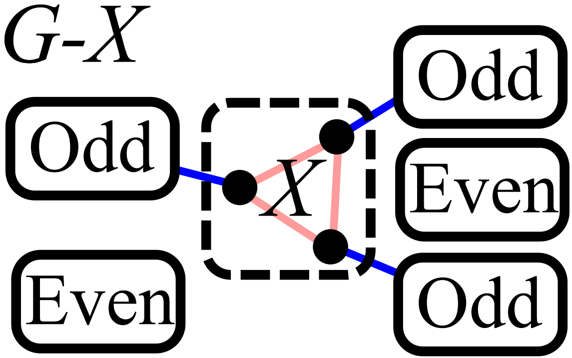
$$|X| = \text{\# of odd connected components in } G - X$$



Barriers

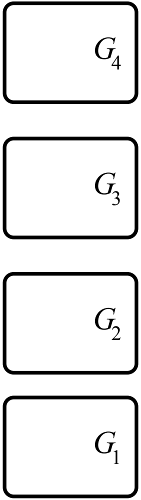
Let $\Phi(G) > 0$. A set $X \subset V(G)$ is a **barrier** if

$$|X| = \text{\# of odd connected components in } G - X$$



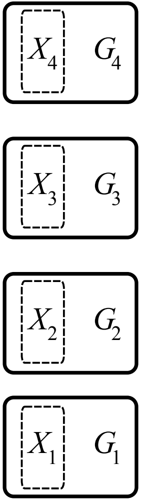
Spires

Let $G_1, \ldots, G_k$ be chambers with barriers $X_1, \ldots, X_k$ where X_i is of maximum size in G_i.



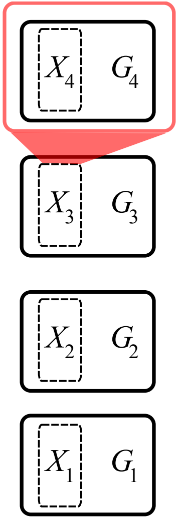
Spires

Let $G_1, \ldots, G_k$ be chambers with barriers $X_1, \ldots, X_k$ where X_i is of maximum size in G_i.



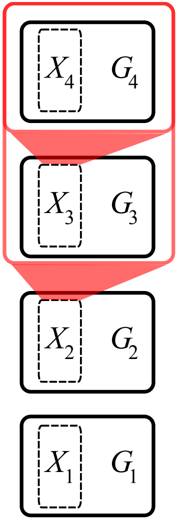
Spires

Let $G_1, \ldots, G_k$ be chambers with barriers $X_1, \ldots, X_k$ where X_i is of maximum size in G_i.



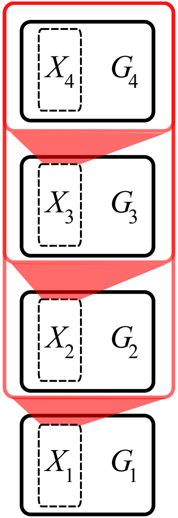
Spires

Let $G_1, \ldots, G_k$ be chambers with barriers $X_1, \ldots, X_k$ where X_i is of maximum size in G_i.



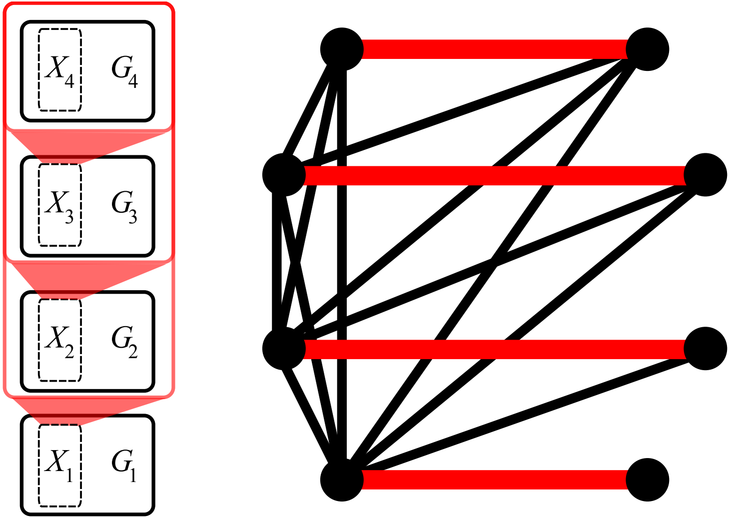
Spires

Let $G_1, \ldots, G_k$ be chambers with barriers $X_1, \ldots, X_k$ where X_i is of maximum size in G_i.



Spires

Let $G_1, \ldots, G_k$ be chambers with barriers $X_1, \ldots, X_k$ where X_i is of maximum size in G_i.



p-Extremal Graphs are Spires

Theorem (Hartke, Stolee, West, Yancey '12) If G is p-extremal, then G is a spire of chambers $G_1, \ldots, G_k$, with barriers $X_i \subseteq V(G_i)$ of maximum size.

p-Extremal Graphs are Spires

Theorem (Hartke, Stolee, West, Yancey '12) If G is p-extremal, then G is a spire of chambers $G_1, \dots, G_k$, with barriers $X_i \subseteq V(G_i)$ of maximum size. In addition:

1. $p = \Phi(G) = \prod_{i=1}^{k} \Phi(G_i)$. Let $p_i = \Phi(G_i)$.

p-Extremal Graphs are Spires

Theorem (Hartke, Stolee, West, Yancey '12) If G is p-extremal, then G is a spire of chambers $G_1, \ldots, G_k$, with barriers $X_i \subseteq V(G_i)$ of maximum size. In addition:

1. $p = \Phi(G) = \prod_{i=1}^{k} \Phi(G_i)$. Let $p_i = \Phi(G_i)$.
2. If $p_i = 1$, then $H_i \cong K_2$.

p-Extremal Graphs are Spires

Theorem (Hartke, Stolee, West, Yancey '12) If G is p-extremal, then G is a spire of chambers $G_1, \ldots, G_k$, with barriers $X_i \subseteq V(G_i)$ of maximum size. In addition:

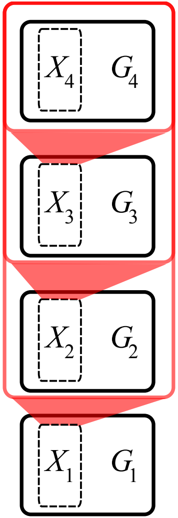
1. $p = \Phi(G) = \prod_{i=1}^{k} \Phi(G_i)$. Let $p_i = \Phi(G_i)$.
2. If $p_i = 1$, then $H_i \cong K_2$.
3. There are at most $N_{p_i} = O(\sqrt{p_i})$ vertices in G_i, and $c(G_i) \leq c_{p_i}$.

p-Extremal Graphs are Spires

Theorem (Hartke, Stolee, West, Yancey '12) If G is p-extremal, then G is a spire of chambers $G_1, \dots, G_k$, with barriers $X_i \subseteq V(G_i)$ of maximum size. In addition:

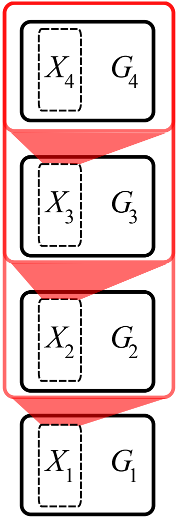
1. $p = \Phi(G) = \prod_{i=1}^{k} \Phi(G_i)$. Let $p_i = \Phi(G_i)$.
2. If $p_i = 1$, then $H_i \cong K_2$.
3. There are at most $N_{p_i} = O(\sqrt{p_i})$ vertices in G_i, and $c(G_i) \leq c_{p_i}$.
4. $c_p = c(G) \leq \sum_{i=1}^{k} c(G_i)$ with equality if and only if $\frac{|X_i|}{|V(G_i)|} = \frac{1}{2}$ for all $i < k$.

Order of Chambers



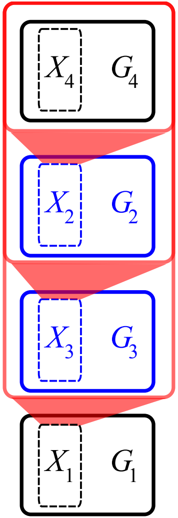
$$\frac{|X_4|}{|V(G_4)|} \leq \frac{|X_3|}{|V(G_3)|} \leq \frac{|X_2|}{|V(G_2)|} \leq \frac{|X_1|}{|V(G_1)|}$$

Order of Chambers



$$\frac{|X_4|}{|V(G_4)|} \leq \frac{|X_3|}{|V(G_3)|} = \frac{|X_2|}{|V(G_2)|} \leq \frac{|X_1|}{|V(G_1)|}$$

Order of Chambers



$$\frac{|X_4|}{|V(G_4)|} \leq \frac{|X_2|}{|V(G_2)|} = \frac{|X_3|}{|V(G_3)|} \leq \frac{|X_1|}{|V(G_1)|}$$

Finiteness

Characterizing p-extremal graphs becomes finite for each fixed p.

Finiteness

Characterizing p-extremal graphs becomes finite for each fixed p.

If G has p perfect matchings and $c = c(G)$, then

$$N_p \leq 3 + \sqrt{16p - 8c - 23}$$

is an upper bound on the maximum size of a p-extremal chamber.

Finiteness

Characterizing p-extremal graphs becomes finite for each fixed p.

If G has p perfect matchings and $c = c(G)$, then

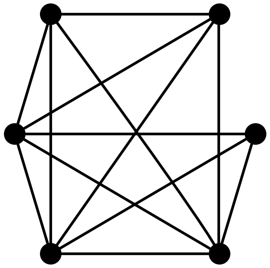
$$N_p \leq 3 + \sqrt{16p - 8c - 23}$$

is an upper bound on the maximum size of a p-extremal chamber.

For $p \leq 10$, $N_p \leq 12$ and `geng` can enumerate all possible graphs.

Example: $p = 7$

Theorem (HSWY, '12) For even n with $n \geq 6$, the unique 7-extremal graph has $\frac{n^2}{4} + 3$ edges and is a spire with $k = n/2 - 2$ chambers $G_1, \ldots, G_k$ are given by $G_i = K_2$ for $i < k$ and G_k given below.



Example: Generating Graphs by Vertex Additions

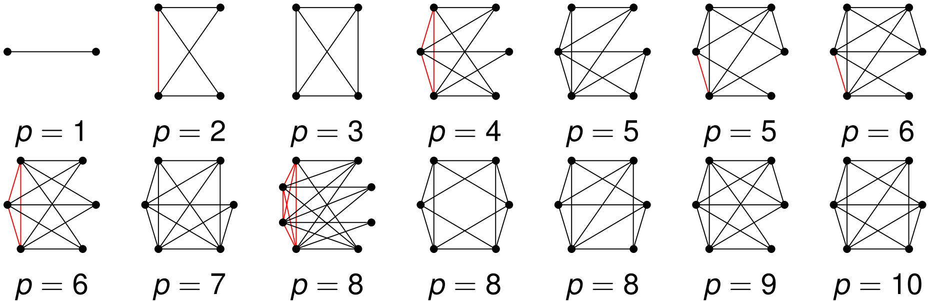
Let's generate all graphs of order n by adding vertices one-by-one.

Augmentation: Add a vertex adjacent to a set $S \subset V(G)$.

IMPORTANT: Only one augmentation per orbit!

Deletion: Select a vertex $v \in V(G)$ to delete, $G' = G - v$.

Extremal Chambers for $p \leq 10$

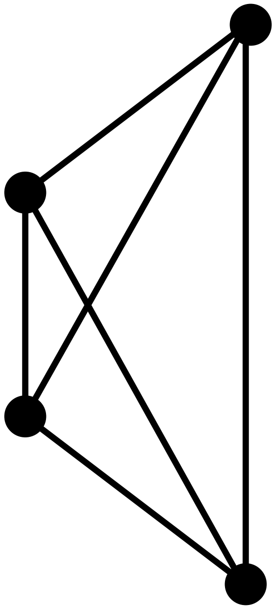


Values of c_p for $p \leq 10$

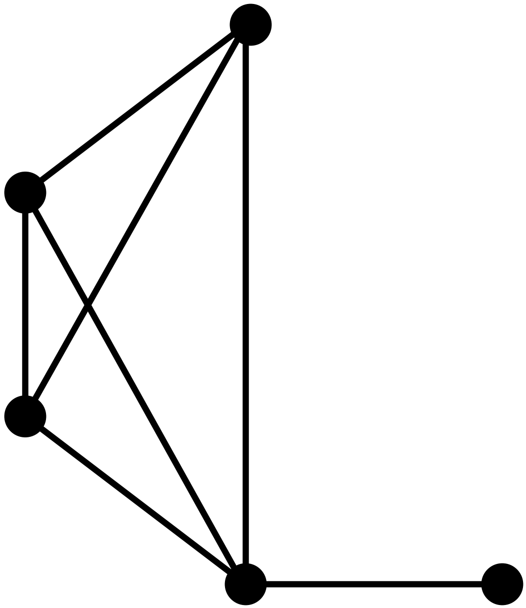
p	1	2	3	4	5	6	7	8	9	10
n_p	2	4	4	6	6	6	6	6	6	6
c_p	0	1	2	2	2	3	3	3	4	4
N_p	2	4	6	8	8	10	10	12	12	12
	Dudek & Schmitt						HSWY			

Table: Known values of n_p and c_p.

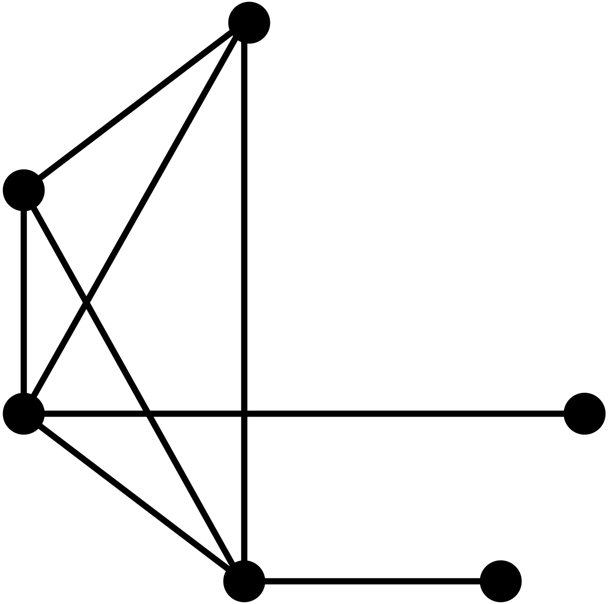
Φ Not Monotonic for Vertex Augmentations



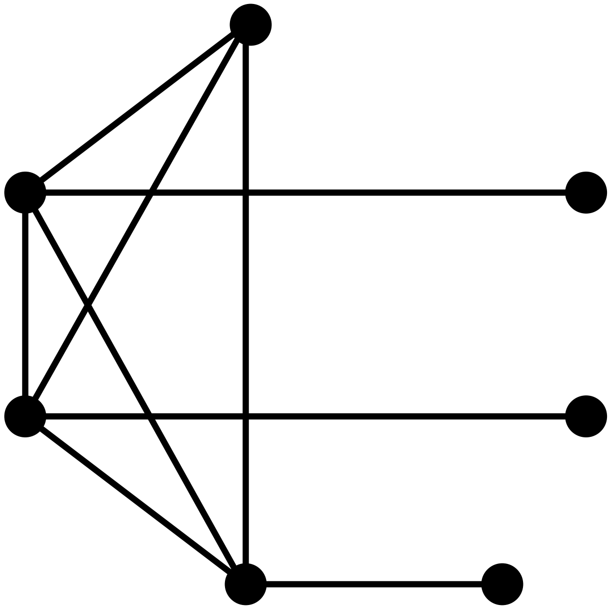
Φ Not Monotonic for Vertex Augmentations



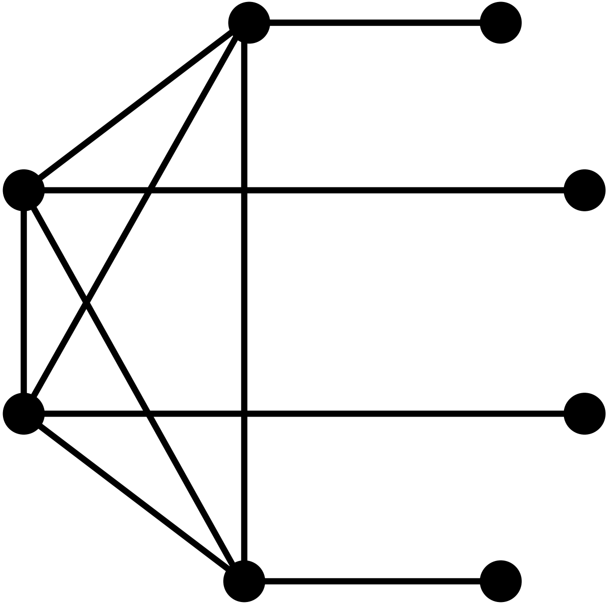
Φ Not Monotonic for Vertex Augmentations



Φ Not Monotonic for Vertex Augmentations



Φ Not Monotonic for Vertex Augmentations



Focus on Chambers

Since the structure theorem only depends on combinations of chambers, we can generate **chambers** of **maximum excess**.

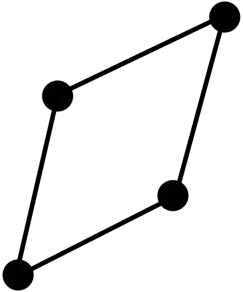
Focus on Chambers

Since the structure theorem only depends on combinations of chambers, we can generate **chambers** of **maximum excess**.

We break chambers into the **extendable** and **forbidden** edges.

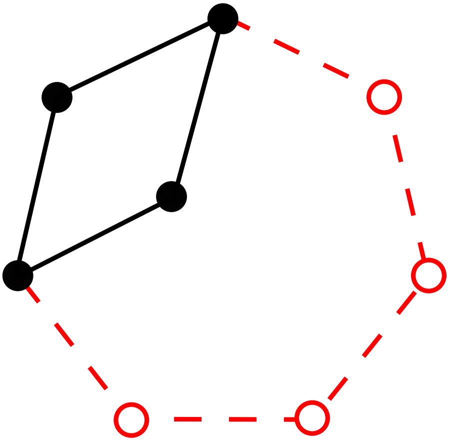
Extendable Graphs are 2-connected

A graph is **2-connected** if no single vertex deletion disconnects the graph.



Extendable Graphs are 2-connected

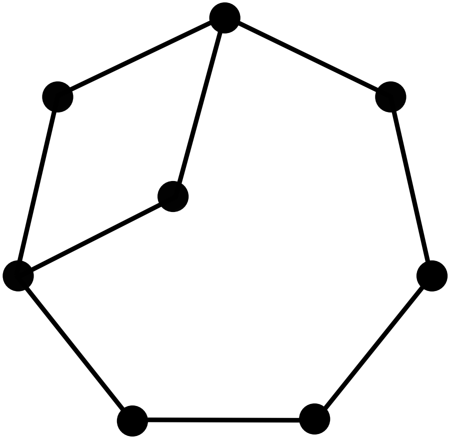
A graph is **2-connected** if no single vertex deletion disconnects the graph.



2-connected graphs can be built by **ear augmentations**.

Extendable Graphs are 2-connected

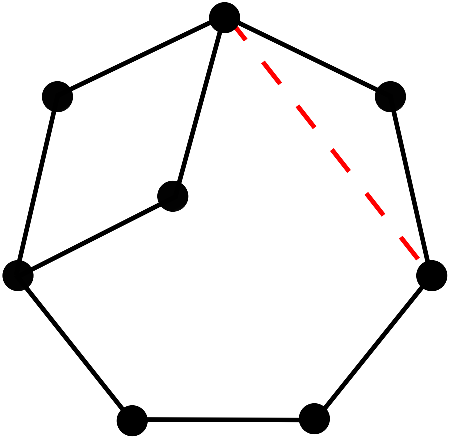
A graph is **2-connected** if no single vertex deletion disconnects the graph.



2-connected graphs can be built by **ear augmentations**.

Extendable Graphs are 2-connected

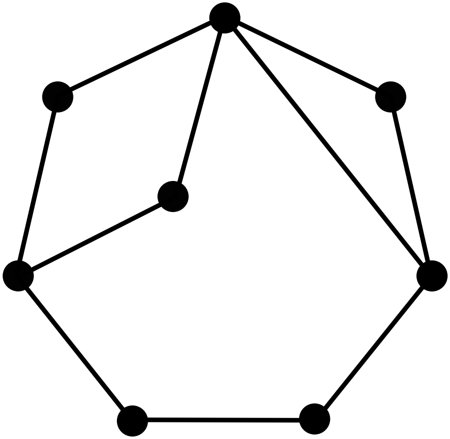
A graph is **2-connected** if no single vertex deletion disconnects the graph.



2-connected graphs can be built by **ear augmentations**.

Extendable Graphs are 2-connected

A graph is **2-connected** if no single vertex deletion disconnects the graph.



2-connected graphs can be built by **ear augmentations**.

The Structure of Extendable Edges

A connected graph is **extendable** if all edges are extendable

The Structure of Extendable Edges

A connected graph is **extendable** if all edges are extendable

Theorem (Lovász Two-Ears Theorem) If H is a extendable graph, there is a **graded ear decomposition**

$$H_0 \subset H_1 \subset H_2 \subset \cdots \subset H_k$$

such that

The Structure of Extendable Edges

A connected graph is **extendable** if all edges are extendable

Theorem (Lovász Two-Ears Theorem) If H is a extendable graph, there is a **graded ear decomposition**

$$H_0 \subset H_1 \subset H_2 \subset \cdots \subset H_k$$

such that

1. $H_0 \cong C_{2\ell}$ for some ℓ and $H_k = H$.

The Structure of Extendable Edges

A connected graph is **extendable** if all edges are extendable

Theorem (Lovász Two-Ears Theorem) If H is a extendable graph, there is a **graded ear decomposition**

$$H_0 \subset H_1 \subset H_2 \subset \cdots \subset H_k$$

such that

1. $H_0 \cong C_{2\ell}$ for some ℓ and $H_k = H$.
2. Each H_i is extendable.

The Structure of Extendable Edges

A connected graph is **extendable** if all edges are extendable

Theorem (Lovász Two-Ears Theorem) If H is a extendable graph, there is a **graded ear decomposition**

$$H_0 \subset H_1 \subset H_2 \subset \cdots \subset H_k$$

such that

1. $H_0 \cong C_{2\ell}$ for some ℓ and $H_k = H$.
2. Each H_i is extendable.
3. Each ear augmentation $H_i \subset H_{i+1}$ uses one or two ears **of even order**.

The Structure of Extendable Edges

A connected graph is **extendable** if all edges are extendable

Theorem (Lovász Two-Ears Theorem) If H is a extendable graph, there is a **graded ear decomposition**

$$H_0 \subset H_1 \subset H_2 \subset \cdots \subset H_k$$

such that

1. $H_0 \cong C_{2\ell}$ for some ℓ and $H_k = H$.
2. Each H_i is extendable.
3. Each ear augmentation $H_i \subset H_{i+1}$ uses one or two ears **of even order**.

Graphs which appear "between" two extendable graphs in a two-ear augmentation are *almost extendable* graphs.

Example: Generating Graphs by Ear Augmentations

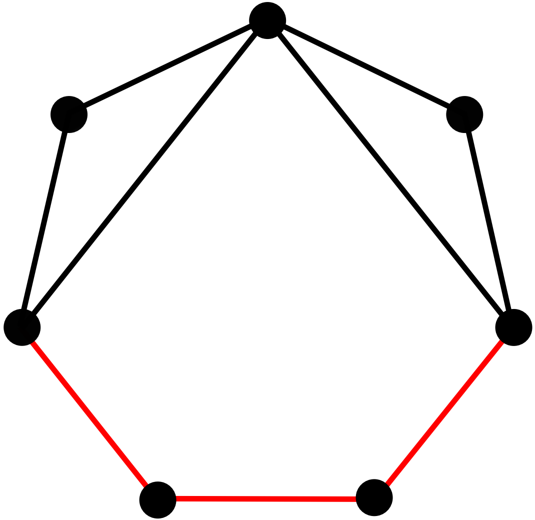
Let's generate all graphs of order n by adding vertices one-by-one.

Initialization: Let G be a cycle.

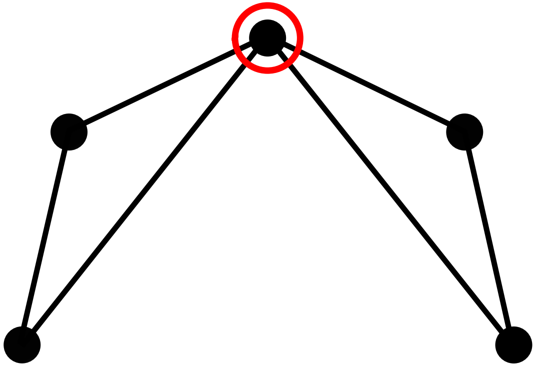
Augmentation: Let $x, y \in V(G)$ be distinct vertices and ℓ a length.
Add an ear of length ℓ between x and y.

Deletion: Select an ear to delete, such that G remains 2-connected.

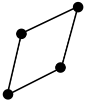
Generating with Ear Augmentations

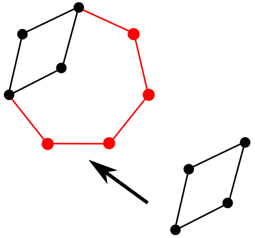


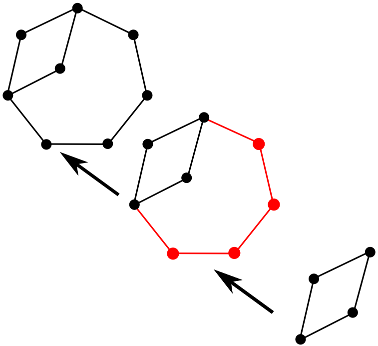
Generating with Ear Augmentations

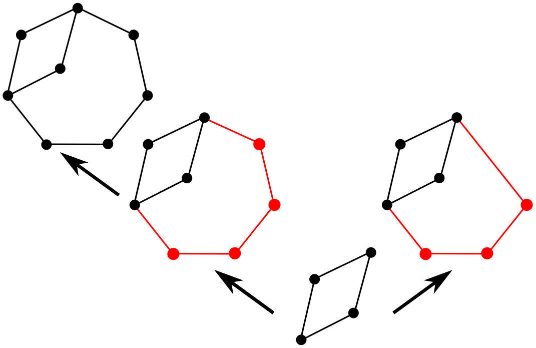


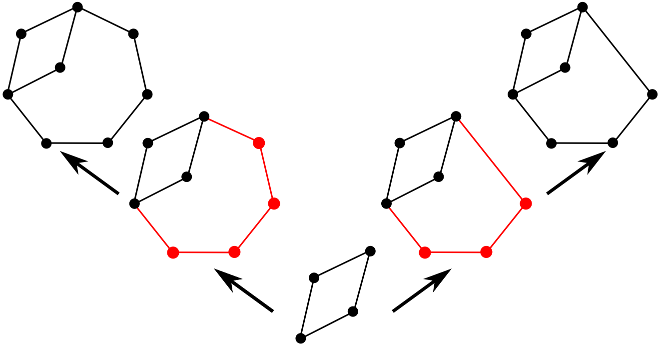
Generating with Ear Augmentations

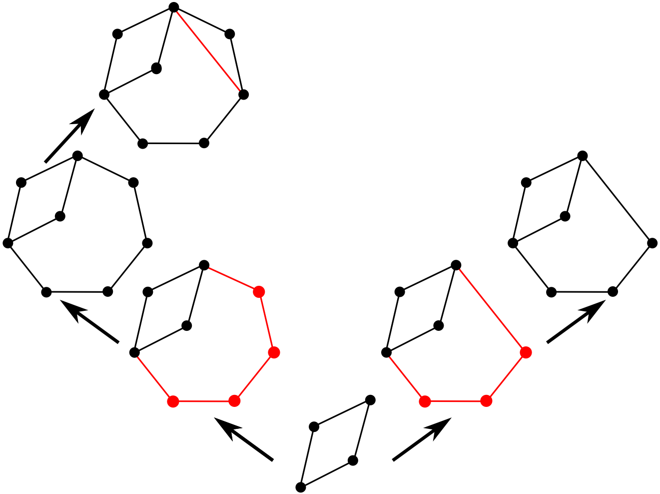


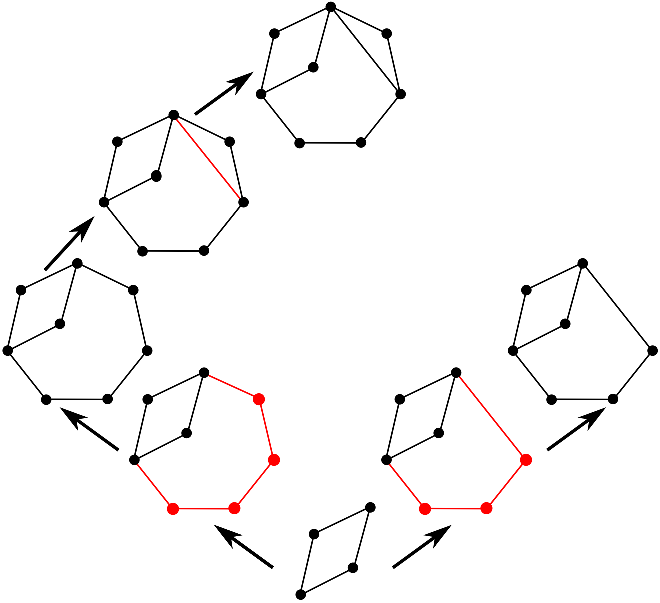


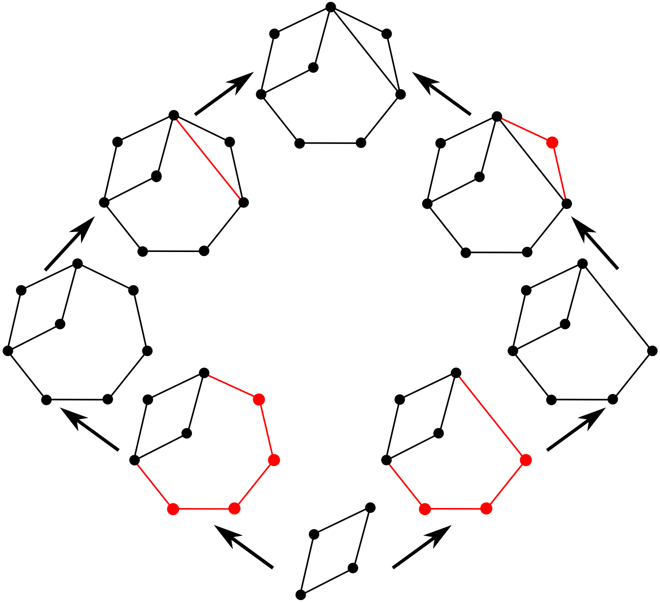


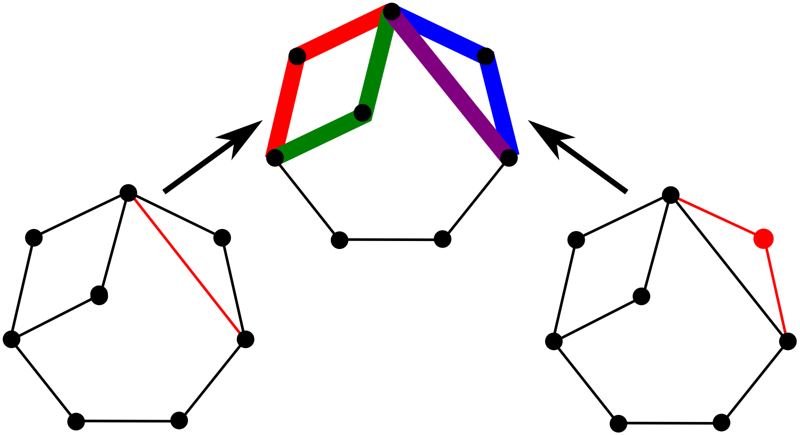


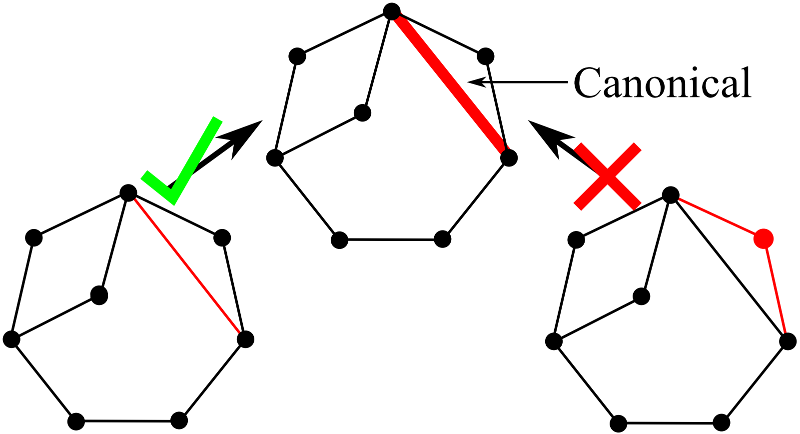











Canonical

Canonical Deletion by Filtering

Let S be the set of ears of G. Filter S until $|S| = 1$ by the following conditions:

Canonical Deletion by Filtering

Let S be the set of ears of G. Filter S until $|S| = 1$ by the following conditions:

1. Remove ears $\varepsilon \in S$ such that $G - \varepsilon$ is not 2-connected.

Canonical Deletion by Filtering

Let S be the set of ears of G. Filter S until $|S| = 1$ by the following conditions:

1. Remove ears $\varepsilon \in S$ such that $G - \varepsilon$ is not 2-connected.
2. If G is almost-extendable, then remove ears $\varepsilon \in S$ such that $G - \varepsilon$ is not extendable.

Canonical Deletion by Filtering

Let S be the set of ears of G. Filter S until $|S| = 1$ by the following conditions:

1. Remove ears $\varepsilon \in S$ such that $G - \varepsilon$ is not 2-connected.
2. If G is almost-extendable, then remove ears $\varepsilon \in S$ such that $G - \varepsilon$ is not extendable.
3. Among ears in S, minimize their length.

Canonical Deletion by Filtering

Let S be the set of ears of G. Filter S until $|S| = 1$ by the following conditions:

1. Remove ears $\varepsilon \in S$ such that $G - \varepsilon$ is not 2-connected.
2. If G is almost-extendable, then remove ears $\varepsilon \in S$ such that $G - \varepsilon$ is not extendable.
3. Among ears in S, minimize their length.
4. Among ears in S, minimize the degrees of their endpoints.

Canonical Deletion by Filtering

Let S be the set of ears of G. Filter S until $|S| = 1$ by the following conditions:

1. Remove ears $\varepsilon \in S$ such that $G - \varepsilon$ is not 2-connected.
2. If G is almost-extendable, then remove ears $\varepsilon \in S$ such that $G - \varepsilon$ is not extendable.
3. Among ears in S, minimize their length.
4. Among ears in S, minimize the degrees of their endpoints.
5. Compute a canonical labeling ℓ, and set

$$\varepsilon = \text{argmin}_{\varepsilon \in S}\{n(G)\ell(\varepsilon_1) + \ell(\varepsilon_2)\}.$$

Canonical Deletion by Filtering

Let S be the set of ears of G. Filter S until $|S| = 1$ by the following conditions:

1. Remove ears $\varepsilon \in S$ such that $G - \varepsilon$ is not 2-connected.
2. If G is almost-extendable, then remove ears $\varepsilon \in S$ such that $G - \varepsilon$ is not extendable.
3. Among ears in S, minimize their length.
4. Among ears in S, minimize the degrees of their endpoints.
5. Compute a canonical labeling ℓ, and set

$$\varepsilon = \text{argmin}_{\varepsilon \in S}\{n(G)\ell(\varepsilon_1) + \ell(\varepsilon_2)\}.$$

The ear ε is the canonical deletion.

Adding in the Forbidden Edges

We can generate the extendable edges by ear augmentations.

Adding in the Forbidden Edges

We can generate the extendable edges by ear augmentations.

Adding an edge between two vertices in the same barrier adds a forbidden edge.

Adding in the Forbidden Edges

We can generate the extendable edges by ear augmentations.

Adding an edge between two vertices in the same barrier adds a forbidden edge.

However, this changes the *other* barriers.

Adding in the Forbidden Edges

We can generate the extendable edges by ear augmentations.

Adding an edge between two vertices in the same barrier adds a forbidden edge.

However, this changes the *other* barriers.

We can assume that all barriers in a p-extremal graph are **cliques**.

Adding in the Forbidden Edges

We can generate the extendable edges by ear augmentations.

Adding an edge between two vertices in the same barrier adds a forbidden edge.

However, this changes the *other* barriers.

We can assume that all barriers in a p-extremal graph are **cliques**.

If H is extendable, then let $\mathcal{E}(H)$ be the collection of supergraphs G where all edges in $E(G) \setminus E(H)$ are forbidden.

The Extremal Two-Ears Theorem

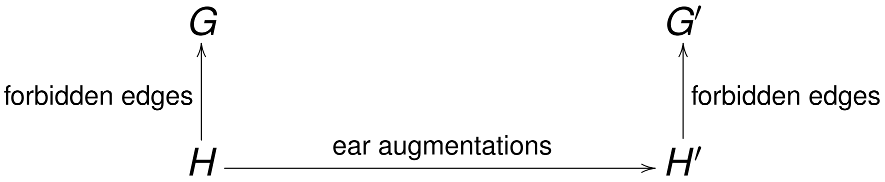
Lemma. Let H be a 1-extendable graph on n vertices with $\Phi(H) = q$. Let H' be a 1-extendable supergraph of H built from H by a graded ear decomposition. Let $\Phi(H') = p > q$ and $N = n(H')$. Choose $G \in \mathcal{E}(H)$ and $G' \in \mathcal{E}(H')$ with the maximum number of edges in each set. Then,

$$c(G') \leq c(G) + 2(p - q) - \frac{1}{4}(N - n)(n - 2).$$

The Extremal Two-Ears Theorem

Lemma. Let H be a 1-extendable graph on n vertices with $\Phi(H) = q$. Let H' be a 1-extendable supergraph of H built from H by a graded ear decomposition. Let $\Phi(H') = p > q$ and $N = n(H')$. Choose $G \in \mathcal{E}(H)$ and $G' \in \mathcal{E}(H')$ with the maximum number of edges in each set. Then,

$$c(G') \leq c(G) + 2(p-q) - \frac{1}{4}(N-n)(n-2).$$



The Full Search Algorithm (In Parts)

Begin with p, c, N. Generate all chambers G with p perfect matchings, $c(G) \geq c$, and $n(G) \leq N$.

The Full Search Algorithm (In Parts)

Begin with p, c, N. Generate all chambers G with p perfect matchings, $c(G) \geq c$, and $n(G) \leq N$.

1. Start with an even cycle H of order at most N.

The Full Search Algorithm (In Parts)

Begin with p, c, N. Generate all chambers G with p perfect matchings, $c(G) \geq c$, and $n(G) \leq N$.

1. Start with an even cycle H of order at most N.
2. Add ear augmentations to H that match canonical deletions.

The Full Search Algorithm (In Parts)

Begin with p, c, N. Generate all chambers G with p perfect matchings, $c(G) \geq c$, and $n(G) \leq N$.

1. Start with an even cycle H of order at most N.
2. Add ear augmentations to H that match canonical deletions.
3. If $\Phi(H) > p$, then backtrack.

The Full Search Algorithm (In Parts)

Begin with p, c, N. Generate all chambers G with p perfect matchings, $c(G) \geq c$, and $n(G) \leq N$.

1. Start with an even cycle H of order at most N.
2. Add ear augmentations to H that match canonical deletions.
3. If $\Phi(H) > p$, then backtrack.
4. Maintain and update list of barriers on H.

The Full Search Algorithm (In Parts)

Begin with p, c, N. Generate all chambers G with p perfect matchings, $c(G) \geq c$, and $n(G) \leq N$.

1. Start with an even cycle H of order at most N.
2. Add ear augmentations to H that match canonical deletions.
3. If $\Phi(H) > p$, then backtrack.
4. Maintain and update list of barriers on H.
5. Find maximum chambers G by adding forbidden edges to H.

The Full Search Algorithm (In Parts)

Begin with p, c, N. Generate all chambers G with p perfect matchings, $c(G) \geq c$, and $n(G) \leq N$.

1. Start with an even cycle H of order at most N.
2. Add ear augmentations to H that match canonical deletions.
3. If $\Phi(H) > p$, then backtrack.
4. Maintain and update list of barriers on H.
5. Find maximum chambers G by adding forbidden edges to H.
6. If $c(G) + 2(p - \Phi(H)) < c$, then backtrack.

The Full Search Algorithm (In Parts)

Begin with p, c, N. Generate all chambers G with p perfect matchings, $c(G) \geq c$, and $n(G) \leq N$.

1. Start with an even cycle H of order at most N.
2. Add ear augmentations to H that match canonical deletions.
3. If $\Phi(H) > p$, then backtrack.
4. Maintain and update list of barriers on H.
5. Find maximum chambers G by adding forbidden edges to H.
6. If $c(G) + 2(p - \Phi(H)) < c$, then backtrack.
7. If $\Phi(H) = p$, then output all maximum G (with $c(G) \geq c$).

Timing

p	N_p	c_p	CPU Time
5	8	2	0.02s
6	10	3	0.04s
7	10	3	0.18s
8	12	3	0.72s
9	12	4	1.46s
10	12	4	5.95s
11	14	3	43.29s
12	14	5	44.01s
13	14	3	6.66m
14	16	4	12.17m
15	16	6	12.71m

p	N_p	c_p	CPU Time
16	16	4	2.02h
17	16	4	6.77h
18	18	5	11.75h
19	18	4	2.71d
20	18	5	4.21d
21	18	5	13.71d
22	20	5	42.84d
23	20	5	118.32d
24	20	6	209.42d
25	20	5	2.52y
26	20	5	7.21y
27	22	6	10.68y

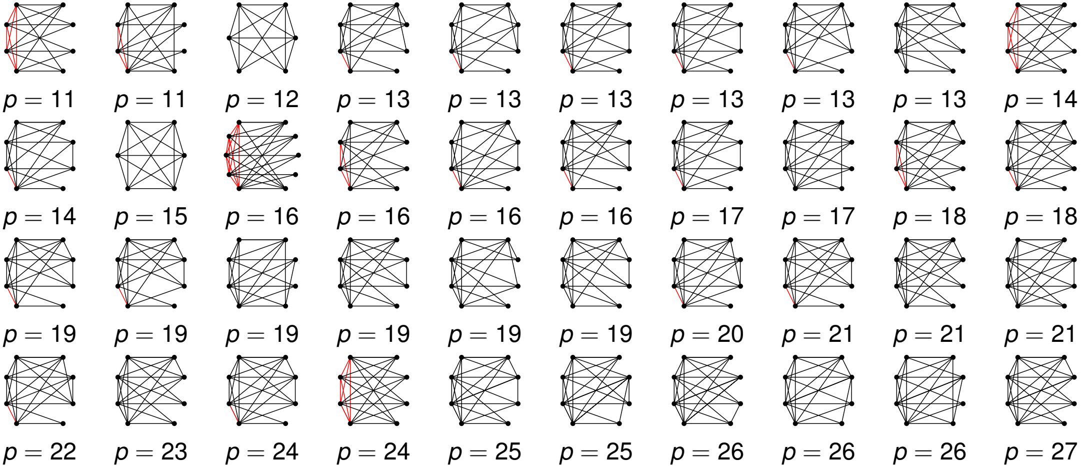
Results

p	1	2	3	4	5	6	7	8	9	10
c_p	**0**	**1**	**2**	**2**	**2**	**3**	**3**	**3**	**4**	**4**
n_p	2	4	4	6	6	6	6	6	6	6

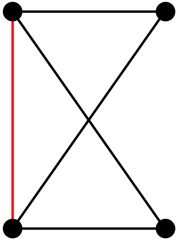
p	11	12	13	14	15	16	17	18	19	20
c_p	3	**5**	3	4	**6**	4	4	5	4	5
n_p	8	6	8	8	6	8	8	8	8	8

p	21	22	23	24	25	26	27			
c_p	5	5	5	**6**	5	5	**6**			
n_p	8	8	8	8	8	8	8			

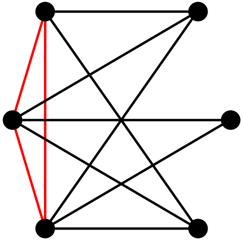
Extremal Chambers for $11 \leq p \leq 27$



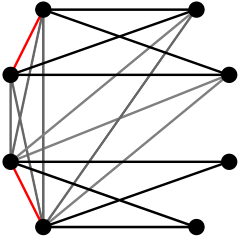
p-Extremal Configurations for $p \in \{2, 4\}$



$p = 2$
$c_2 = 1$

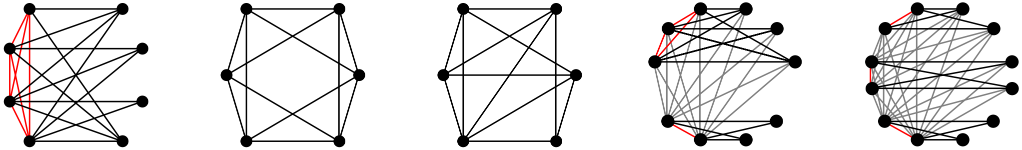


$p = 4$
$c_4 = 2$



$p = 4$
$c_4 = 2$

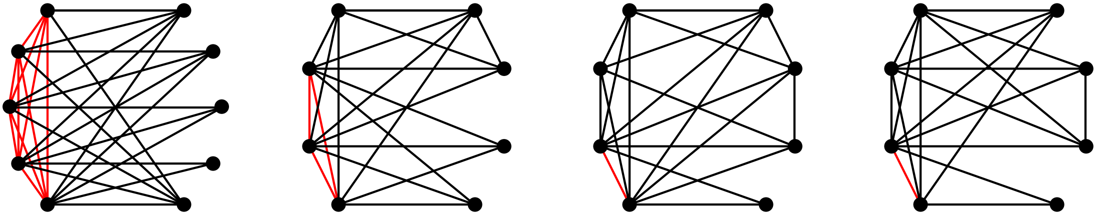
p-Extremal Configurations for $p = 8$



$p = 8$

$c_8 = 3$

p-Extremal Configurations for $p = 16$



$p = 16$

$c_{16} = 4$

Open Problems!

1. Compute more values of c_p.

Open Problems!

1. Compute more values of c_p.
2. Show a growing lower bound on c_p.

Open Problems!

1. Compute more values of c_p.
2. Show a growing lower bound on c_p.
3. Show a logarithmic(?) upper bound on N_p^*.

Open Problems!

1. Compute more values of c_p.
2. Show a growing lower bound on c_p.
3. Show a logarithmic(?) upper bound on N_p^*.

Conjecture (Hartke, Stolee, West, Yancey, '12) Let p, k, t be integers so that $k \in \{1, \ldots, 2t\}$ and

$$k(2t-1)!! \leq p < (k+1)(2t-1)!!$$

and set $C_p = t^2 - t + k - 1$. **Always** $c_p \leq C_p$.

Open Problems!

1. Compute more values of c_p.
2. Show a growing lower bound on c_p.
3. Show a logarithmic(?) upper bound on N_p^*.

Conjecture (Hartke, Stolee, West, Yancey, '12) Let p, k, t be integers so that $k \in \{1, \ldots, 2t\}$ and

$$k(2t-1)!! \leq p < (k+1)(2t-1)!!$$

and set $C_p = t^2 - t + k - 1$. **Always $c_p \leq C_p$.**

If the conjecture holds, then $c_p \leq O\left(\left(\frac{\log p}{\log \log p}\right)^2\right)$.

If you learned ANYTHING...

If you learned ANYTHING...

...then it should be that

pairing structural theorems with specialized algorithms can be

very effective!

Generating p-extremal graphs

Derrick Stolee

Iowa State University
dstolee@iastate.edu
http://www.math.iastate.edu/dstolee/

December 9, 2013
ISU MECS Seminar