

# On Decompositions, Generation Methods, and related concepts in the theory of Matching Covered Graphs

**Student:** Janmenjaya Panda

**Guide:** Nishad Kothari



Indian Institute of Technology Madras

May 24, 2024

# How this proposal came into picture?

# How this proposal came into picture?

The story began with:

# How this proposal came into picture?

## The story began with:



### When all removable edges lead to wheels: (Application: characterizing graphs that are Birkhoff-on-Neumann and PM-compact)

Marcelo Henriques de Carvalho<sup>1</sup> Nishad Kothari<sup>2</sup> Janmenjaya Panda<sup>2</sup>  
<sup>1</sup>Universidade Federal do Mato Grosso do Sul <sup>2</sup>Indian Institute of Technology Madras

#### Welcome to the show

Matchings and perfect matchings have received considerable attention in graph theory as well as in other related domains (such as but not limited to algorithms and optimization). In particular, the perfect matching problem has been studied extensively. However, there still remains many open questions pertaining to the polytope and the related graph classes – to which we do not have the answers. This research aims to define a simple graph of the study by Carvalho, Kothari, Wang, Sun (2020) on the characterization of a particular graph class, that are Birkhoff-on-Neumann as well as PM-compact. This proof emerges as an application of a conjecture which addresses the graphs: the removable edges of which ultimately lead to wheels.

#### Some mathematical jargons

- Perfect Matching** All of the covered graphs in our entire journey are undirected and bipartite. There are no edges connecting one vertex to itself. For each a graph  $G$ , a set of edges  $M \subseteq E(G)$  is called a **perfect matching** if every vertex is incident with exactly one edge in  $M$ .
- Matching covered graph** A connected undirected graph  $G$  is called **matching covered** if each edge belongs to some perfect matching.
- Removable edge** An edge  $e$  of a matching covered graph  $G$  is **removable** if  $G - e$  is also matching covered.



The result of removing  $e$  from  $G$  shows in red and single



In the figure,  $P$  every edge is removable.

#### Matching covered graphs and removable edges

- B-contracting** The term B-contracting refers to the shrinking of a degree three vertex and its two neighbors into a single vertex. Note that if a graph  $G$  has minimum degree 3 or more, then for some edge  $e$ ,  $G - e$  is not a  $B$  contraction. If  $G$  may have at most two vertices of degree three, the B-contracting of all these degree three vertices, for each a graph  $H$ , will lead to the contract of  $H$ .
- Conformal bipartite** A pair of vertex-disjoint cycles (say  $Q_1$  and  $Q_2$ ) is called a **bipartite**. A bipartite  $Q_1 \cup Q_2$  of  $G$  is called a **conformal bipartite** if  $G - V(Q_1 \cup Q_2)$  has a perfect matching. The conformal bipartite is a subgraph of  $G$  which is a bipartite graph. A graph  $G$  is called a **conformal bipartite** if it has a conformal bipartite  $Q_1 \cup Q_2$  which is a bipartite graph. For instance, the bipartite  $K_{2,2}$  has an odd conformal bipartite  $Q_1 \cup Q_2$  which is shown in red in the following figure. The perfect matching of  $G - V(Q_1 \cup Q_2)$  is shown in green in the following figure.



B-contracting vertices  $a$

B-contraction

B-contraction and Conformal bipartite



$R_{2a} = V(Q_1 \cup Q_2)$  is a PM.

$R_{2a}$  with  $Q_1 \cup Q_2$

$R_{2a}$  with  $Q_1 \cup Q_2$

#### Conjecture 1

Let  $G$  be a simple matching covered graph with minimum degree 3 or more. If there exists a removable edge  $e$  in  $G$  such that  $\text{contract}(G - e)$  is either  $K_4$  or  $K_{2,2}$  or the Murty graph  $M$  up to multiple copies joining the two vertices respectively, then

- either  $G$  is not Birkhoff-on-Neumann or  $G$  is not PM-compact, or otherwise
- $G$  is the Murty graph  $M$ .

#### Conjecture 1: An illustration and proof overview

An index-0 operation on  $R_1 \subseteq R_2$  leads to  $M$ . Here, the addition of a removable edge refers to index-0 operation. It can be shown that all other such operations lead to non-Birkhoff-on-Neumann or non-PM-compact graphs.

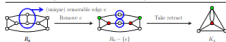


#### Conjecture 2: When all removable edges lead to wheel

Let  $G$  be a simple matching covered graph with minimum degree 3 or more. If for each removable edge  $e$ , the contract of  $G - e$  is either  $K_4$  (with multiple copies) such that it does not have a bipartite, or an odd wheel (up to multiple copies), then  $G$  is one of the following:

- the Petersen  $P_5$
- the Petersen  $P_6$
- the Myer Ladder  $M_4$
- $K_4 \cup K_4$
- an odd wheel

#### Conjecture 2: An illustration and proof overview

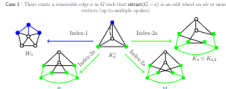


A removable edge  $e$  in  $R_1$  leading to  $K_4^*$   $G$  can be obtained from  $\text{contract}(G - e)$  either by index-0 or by index-2 operation. These operations have been explained in the diagram.

We compare the problem by dividing it into two primary cases.



Case 1: There exists a removable edge  $e$  in  $G$  such that  $\text{contract}(G - e)$  is an odd wheel or an even wheel (up to multiple copies).



Case 2: For each removable edge  $e$  in  $G$ , the graph  $\text{contract}(G - e)$  is  $K_4$  (with multiple copies) such that it does not have a bipartite.



#### Who cares...an application

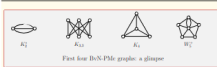
The inspiration for this conjecture arose from Tutte's well-known theorem, highlighting the important role of matchings in several domains of graph theory. Below, an application of this conjecture is presented, showing an alternative proof of the result used by Carvalho, Kothari, Wang, and Sun (2020) on some advanced theorems in Matching Theory such as brick generation theorems of Scarf and Thomassen (1986) characterizing the graphs that are Birkhoff-on-Neumann and PM-compact graphs.

#### Theorem: Birkhoff-on-Neumann graphs that are PM-compact

Let  $G$  be a matching covered graph with minimum degree 3 or more. The following are equivalent:

- $G$  is Birkhoff-on-Neumann as well as PM-compact.
- $G$  does not contain a conformal bipartite.
- $G$  is one of the following:
  - $K_4$  (with at least three multiple copies)
  - $K_{2,2}$
  - $K_4$  (up to multiple copies) such that it does not have a bipartite
  - an odd wheel (up to multiple copies)
  - $K_4 \cup K_4$
  - the Murty graph (up to multiple copies) joining the two vertices respectively

#### The graph gallery



First four Bn-PMc graphs: a glimpse

#### What these creatures: Defining Bn and PMc

In [BKP20], perfect matchings of graph  $G$  are represented as  $0 \rightarrow 1$  vectors, where 1 indicates an edge's inclusion in the matching. The vertex label of these vectors (i.e., the smallest vertex not containing their support) is called the **perfect matching polytope** of the graph  $G$  (denoted as  $\text{PMP}(G)$ ).

**Birkhoff-on-Neumann graph**: A graph  $G$  is **Birkhoff-on-Neumann** if  $\text{PMP}(G)$  is characterized with its vertices and edges.

**Perfect matching compact graph**: The polytope  $P$  (with the simple undirected graph  $G$  and its vertices in  $\mathbb{Z}^V$ ) is called a **perfect matching compact** if and only if the corresponding points belong to a convex polytope (a face of dimension 1) in  $P$ . A polytope  $P$  is compact if  $\text{PMP}(G)$  is a convex polytope. A graph  $G$  is **perfect matching compact** if  $\text{PMP}(G)$  is compact. For example,  $K_4$  (the triangle graph) has a perfect matching and hence a tetrahedron in  $\mathbb{R}^3$  with  $\text{PMP}(K_4)$  as  $K_4$ . Thus,  $K_4$  is PM-compact.

#### An alternative proof of the above theorem

Let  $\mathcal{G}$  denote the class of graphs that are Bn and PMc. By already established results,  $1 \implies 2$  holds true.  $3 \implies 2$  is easily verifiable. We aim to show  $2 \implies 3$  through induction on the number of edges. From matching theorems, one may deduce that for each a graph  $G$  (of size of undirected bipartite), there exists a removable edge (say  $e$ ) and the contract of  $G - e$  (say  $H$ ) belongs to  $\mathcal{G}$ . By the induction hypothesis,  $H$  belongs to the above mentioned families. By further results, one need not deal with  $K_4^*$  or  $K_{2,2}$ . Now we have three cases.

**Case 1**: If  $G$  has multiple edges, if its simple graph includes a degree-three vertex,  $G$  has an even conformal bipartite. Otherwise,  $K_4$ ,  $K_{2,2}$ , or an odd wheel (up to multiple copies) or  $M$ .

**Case 2**:  $G$  has removable edge  $e$  and  $H = K_4 \cup K_4$  or  $M$  (Since Conjecture 1).

**Case 3**: Similar Case 1, but  $G$  is a bipartite. For each removable edge  $e$ ,  $G - e$  (not having a bipartite) or an odd wheel (with multiple copies). Here, we include Conjecture 2.  $\square$

# How this proposal came into picture?

# How this proposal came into picture?

It was 13/12/2023 and...

# How this proposal came into picture?

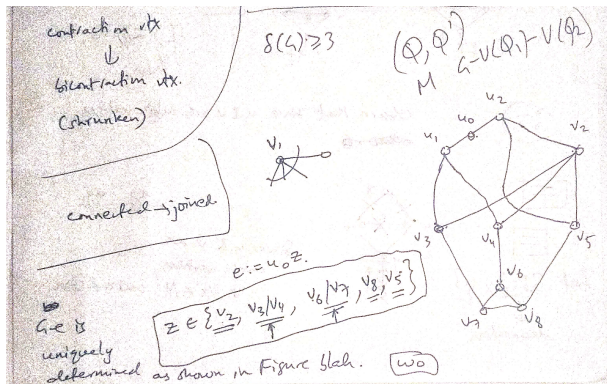
It was 13/12/2023 and...

en route writing the proof the following graph appeared:

# How this proposal came into picture?

It was 13/12/2023 and...

en route writing the proof the following graph appeared:



**Figure:** Check computationally if  $v_6$  and  $v_7$  lie in the same vertex orbit.



# Perfect matching: What is that?

# Perfect matching: What is that?

All graphs are undirected and loopless.

# Perfect matching: What is that?

All graphs are undirected and loopless.

## Perfect Matching

# Perfect matching: What is that?

All graphs are undirected and loopless.

## Perfect Matching

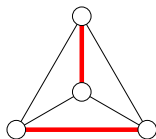
A set  $M$  of edges of a graph  $G$  is called a *perfect matching* if  $|M \cap \partial(v)| = 1$  for each vertex  $v$ .

# Perfect matching: What is that?

All graphs are undirected and loopless.

## Perfect Matching

A set  $M$  of edges of a graph  $G$  is called a *perfect matching* if  $|M \cap \partial(v)| = 1$  for each vertex  $v$ .



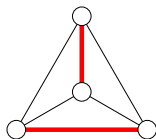
The complete graph  $K_4$   
has a perfect matching

# Perfect matching: What is that?

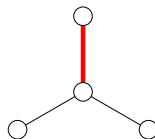
All graphs are undirected and loopless.

## Perfect Matching

A set  $M$  of edges of a graph  $G$  is called a *perfect matching* if  $|M \cap \partial(v)| = 1$  for each vertex  $v$ .



The complete graph  $K_4$   
has a perfect matching



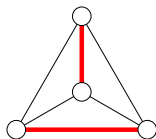
The claw  $K_{1,3}$   
does not have a perfect matching

# Perfect matching: What is that?

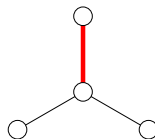
All graphs are undirected and loopless.

## Perfect Matching

A set  $M$  of edges of a graph  $G$  is called a *perfect matching* if  $|M \cap \partial(v)| = 1$  for each vertex  $v$ .



The complete graph  $K_4$   
has a perfect matching



The claw  $K_{1,3}$   
does not have a perfect matching

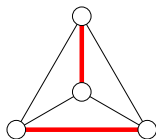
A graph is *matchable* if it has a perfect matching and

# Perfect matching: What is that?

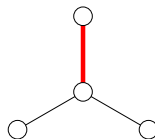
All graphs are undirected and loopless.

## Perfect Matching

A set  $M$  of edges of a graph  $G$  is called a *perfect matching* if  $|M \cap \partial(v)| = 1$  for each vertex  $v$ .



The complete graph  $K_4$   
has a perfect matching



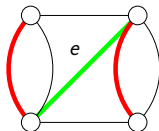
The claw  $K_{1,3}$   
does not have a perfect matching

A graph is *matchable* if it has a perfect matching and an edge of a graph is *matchable* if it participates in some perfect matching.



# Matching covered graphs

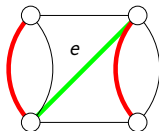
# Matching covered graphs



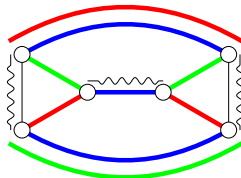
The graph  $C_4^*$  has a PM (red).

But  $e$  does not participate in any PM.

# Matching covered graphs

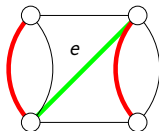


The graph  $C_4^*$  has a PM (red).  
But  $e$  does not participate in any PM.

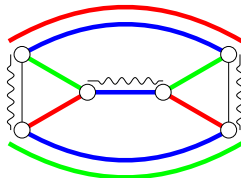


The triangular prism  $\overline{C}_6$  and  
its four PMs (in RGB and squiggle)

# Matching covered graphs



The graph  $C_4^*$  has a PM (red).  
But  $e$  does not participate in any PM.

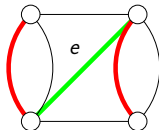


The triangular prism  $\overline{C}_6$  and  
its four PMs (in RGB and squiggle)

## Matching covered graph

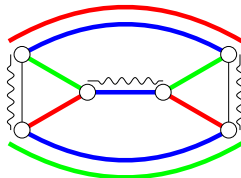
A graph  $G$  is *matching covered*, if

# Matching covered graphs



The graph  $C_4^*$  has a PM (red).

But  $e$  does not participate in any PM.



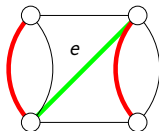
The triangular prism  $\overline{C}_6$  and  
its four PMs (in RGB and squiggle)

## Matching covered graph

A graph  $G$  is *matching covered*, if

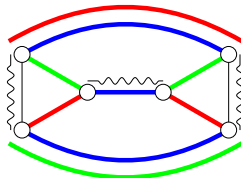
1.  $G$  is connected and  $|V(G)| \geq 2$

# Matching covered graphs



The graph  $C_4^*$  has a PM (red).

But  $e$  does not participate in any PM.



The triangular prism  $\overline{C}_6$  and  
its four PMs (in RGB and squiggle)

## Matching covered graph

A graph  $G$  is *matching covered*, if

1.  $G$  is connected and  $|V(G)| \geq 2$
2. each edge of  $G$  belongs to some perfect matching.

# But, why matching covered graphs?

# But, why matching covered graphs?

Apart from a reduced dimension for perfect matching polytope, matching covered graphs are well respected creatures.



# But, why matching covered graphs?

Apart from a reduced dimension for perfect matching polytope, matching covered graphs are well respected creatures.

- ▶ Counting number of PMs

# But, why matching covered graphs?

Apart from a reduced dimension for perfect matching polytope, matching covered graphs are well respected creatures.

- ▶ Counting number of PMs
- ▶ Rich theory

# But, why matching covered graphs?

Apart from a reduced dimension for perfect matching polytope, matching covered graphs are well respected creatures.

- ▶ Counting number of PMs
- ▶ Rich theory
- ▶ A lot of well known open problems, such as Barnette's conjecture, Berge-Fulkerson conjecture, and so on

# But, why matching covered graphs?

Apart from a reduced dimension for perfect matching polytope, matching covered graphs are well respected creatures.

- ▶ Counting number of PMs
- ▶ Rich theory
- ▶ A lot of well known open problems, such as Barnette's conjecture, Berge-Fulkerson conjecture, and so on

# What is this proposal about?

# What is this proposal about?

To implement:

# What is this proposal about?

To implement:

- ▶ Canonical partition

# What is this proposal about?

To implement:

- ▶ Canonical partition
- ▶ Tight cut decomposition



# What is this proposal about?

To implement:

- ▶ Canonical partition
- ▶ Tight cut decomposition
- ▶ Notable Families of bricks and braces

# What is this proposal about?

To implement:

- ▶ Canonical partition
- ▶ Tight cut decomposition
- ▶ Notable Families of bricks and braces
- ▶ Dependency relation and removable classes

# What is this proposal about?

To implement:

- ▶ Canonical partition
- ▶ Tight cut decomposition
- ▶ Notable Families of bricks and braces
- ▶ Dependency relation and removable classes
- ▶ Ear decompositions

# What is this proposal about?

To implement:

- ▶ Canonical partition
- ▶ Tight cut decomposition
- ▶ Notable Families of bricks and braces
- ▶ Dependency relation and removable classes
- ▶ Ear decompositions
- ▶ Brick and brace generation

# What is this proposal about?

To implement:

- ▶ Canonical partition
- ▶ Tight cut decomposition
- ▶ Notable Families of bricks and braces
- ▶ Dependency relation and removable classes
- ▶ Ear decompositions
- ▶ Brick and brace generation

in the theory of matching covered graphs.

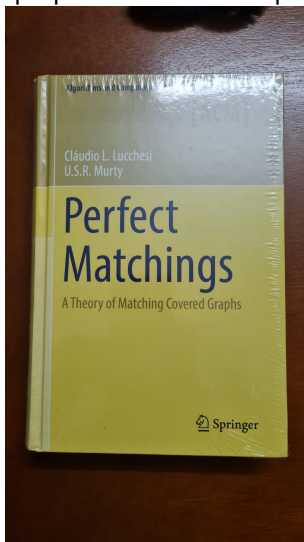
# Inspiration

# Inspiration

This proposal has been inspired by:

# Inspiration

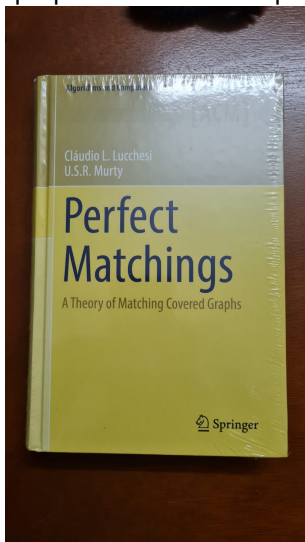
This proposal has been inspired by:





# Inspiration

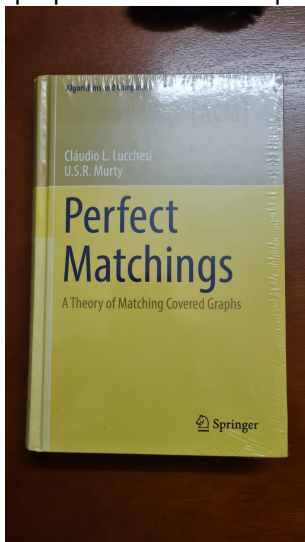
This proposal has been inspired by:



"PERFECT MATCHINGS:

# Inspiration

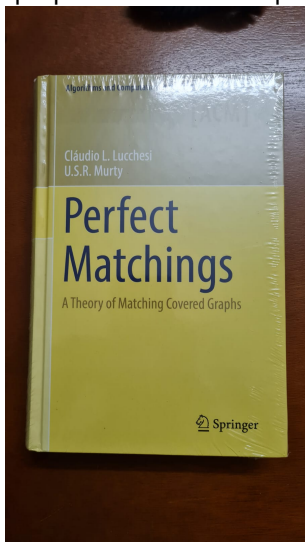
This proposal has been inspired by:



"PERFECT MATCHINGS:  
A THEORY OF MATCHING  
COVERED GRAPHS"

# Inspiration

This proposal has been inspired by:



"PERFECT MATCHINGS:  
A THEORY OF MATCHING  
COVERED GRAPHS"  
Lucchesi and Murty, 2024

# How to check if a graph is matching covered?

# How to check if a graph is matching covered?

To recall:

# How to check if a graph is matching covered?

To recall: A connected nontrivial graph is *matching covered* if each edge participate in some perfect matching.

# How to check if a graph is matching covered?

To recall: A connected nontrivial graph is *matching covered* if each edge participate in some perfect matching.

Matching covered graph

# How to check if a graph is matching covered?

To recall: A connected nontrivial graph is *matching covered* if each edge participate in some perfect matching.

## Matching covered graph

Given a matchable graph  $G$  and a perfect matching  $M$ .



# How to check if a graph is matching covered?

To recall: A connected nontrivial graph is *matching covered* if each edge participate in some perfect matching.

## Matching covered graph

Given a matchable graph  $G$  and a perfect matching  $M$ .  
 $G$  is matching covered, if and only if

# How to check if a graph is matching covered?

To recall: A connected nontrivial graph is *matching covered* if each edge participate in some perfect matching.

## Matching covered graph (DIY)

Given a matchable graph  $G$  and a perfect matching  $M$ .  
 $G$  is matching covered, if and only if for each edge  $ab$  not in  $M$ ,

# How to check if a graph is matching covered?

To recall: A connected nontrivial graph is *matching covered* if each edge participate in some perfect matching.

## Matching covered graph (DIY)

Given a matchable graph  $G$  and a perfect matching  $M$ .  
 $G$  is matching covered, if and only if for each edge  $ab$  not in  $M$ , there exists an  $M$ -alternating  $ab$ -path in  $G$  starting and ending with edges in  $M$ .

# How to check if a graph is matching covered?

To recall: A connected nontrivial graph is *matching covered* if each edge participate in some perfect matching.

## Matching covered graph (DIY)

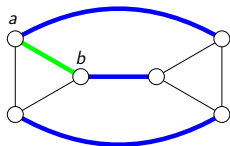
Given a matchable graph  $G$  and a perfect matching  $M$ .  
 $G$  is matching covered, if and only if for each edge  $ab$  not in  $M$ , there exists an  $M$ -alternating  $ab$ -path in  $G$  starting and ending with edges in  $M$ .

# How to check if a graph is matching covered?

To recall: A connected nontrivial graph is *matching covered* if each edge participate in some perfect matching.

## Matching covered graph (DIY)

Given a matchable graph  $G$  and a perfect matching  $M$ .  
 $G$  is matching covered, if and only if for each edge  $ab$  not in  $M$ , there exists an  $M$ -alternating  $ab$ -path in  $G$  starting and ending with edges in  $M$ .



A matchable graph  $G$  and a PM  $M$

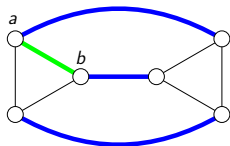
Is  $ab$  matchable?

# How to check if a graph is matching covered?

To recall: A connected nontrivial graph is *matching covered* if each edge participate in some perfect matching.

## Matching covered graph (DIY)

Given a matchable graph  $G$  and a perfect matching  $M$ .  
 $G$  is matching covered, if and only if for each edge  $ab$  not in  $M$ , there exists an  $M$ -alternating  $ab$ -path in  $G$  starting and ending with edges in  $M$ .



A matchable graph  $G$  and a PM  $M$

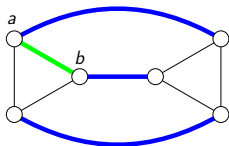
Is  $ab$  matchable?

# How to check if a graph is matching covered?

To recall: A connected nontrivial graph is *matching covered* if each edge participate in some perfect matching.

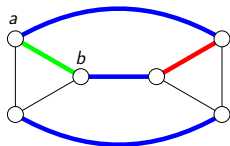
## Matching covered graph (DIY)

Given a matchable graph  $G$  and a perfect matching  $M$ .  
 $G$  is matching covered, if and only if for each edge  $ab$  not in  $M$ , there exists an  $M$ -alternating  $ab$ -path in  $G$  starting and ending with edges in  $M$ .



A matchable graph  $G$  and a PM  $M$

Is  $ab$  matchable?



Yes

The req.  $M$ -alternating path in  $G - ab$

# Maximal barriers and canonical Partition



# Maximal barriers and canonical Partition

- ▶ It turns out that maximal barriers and canonical partition are useful in finding tight cuts.

# Maximal barriers and canonical Partition

- It turns out that maximal barriers and canonical partition are useful in finding tight cuts.

## Barrier in a matchable graph

For a matchable graph  $G$ , a subset  $B$  of the vertex set  $V$  is a barrier if

$$|B| = o(G - B)$$

where  $o(H)$  denotes the  $\#$  of odd components in graph  $H$ .

# Maximal barriers and canonical Partition

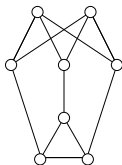
- It turns out that maximal barriers and canonical partition are useful in finding tight cuts.

## Barrier in a matchable graph

For a matchable graph  $G$ , a subset  $B$  of the vertex set  $V$  is a barrier if

$$|B| = o(G - B)$$

where  $o(H)$  denotes the  $\#$  of odd components in graph  $H$ .



An example of a barrier

- For a graph  $G$ , a barrier  $B$  is a maximal barrier if  $C$  is not a barrier for each  $C$  such that  $B \subset C \subseteq V$ .

# Maximal barriers and canonical Partition

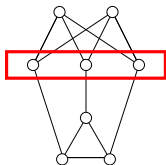
- It turns out that maximal barriers and canonical partition are useful in finding tight cuts.

## Barrier in a matchable graph

For a matchable graph  $G$ , a subset  $B$  of the vertex set  $V$  is a barrier if

$$|B| = o(G - B)$$

where  $o(H)$  denotes the  $\#$  of odd components in graph  $H$ .



An example of a barrier

# Maximal barriers and canonical Partition

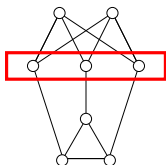
- It turns out that maximal barriers and canonical partition are useful in finding tight cuts.

## Barrier in a matchable graph

For a matchable graph  $G$ , a subset  $B$  of the vertex set  $V$  is a barrier if

$$|B| = o(G - B)$$

where  $o(H)$  denotes the  $\#$  of odd components in graph  $H$ .



An example of a barrier

- For a graph  $G$ , a barrier  $B$  is a maximal barrier if  $C$  is not a barrier for each  $C$  such that  $B \subset C \subseteq V$ .

# Maximal barriers and canonical Partition

## Canonical Partition Theorem; Kotzig 1959

# Maximal barriers and canonical Partition

## Canonical Partition Theorem; Kotzig 1959

The maximal barriers of a matching covered graph  $G$  partition its vertex set, and this partition is called its *canonical partition*.



# Maximal barriers and canonical Partition

## Canonical Partition Theorem; Kotzig 1959

The maximal barriers of a matching covered graph  $G$  partition its vertex set, and this partition is called its *canonical partition*.

- ▶ But, how to compute them in polytime?

## Canonical Partition Theorem; Kotzig 1959

The maximal barriers of a matching covered graph  $G$  partition its vertex set, and this partition is called its *canonical partition*.

- ▶ But, how to compute them in polytime?
- ▶ We use the following theorem:

# Maximal barriers and canonical Partition

## Canonical Partition Theorem; Kotzig 1959

The maximal barriers of a matching covered graph  $G$  partition its vertex set, and this partition is called its *canonical partition*.

- ▶ But, how to compute them in polytime?
- ▶ We use the following theorem:

Theorem: Lucchesi Murty, 2024

# Maximal barriers and canonical Partition

## Canonical Partition Theorem; Kotzig 1959

The maximal barriers of a matching covered graph  $G$  partition its vertex set, and this partition is called its *canonical partition*.

- ▶ But, how to compute them in polytime?
- ▶ We use the following theorem:

## Theorem: Lucchesi Murty, 2024

Let  $u$  and  $v$  be any two vertices in a matchable graph  $G$ . Then the graph  $G - u - v$  is not matchable if and only if there is a barrier of  $G$  which contains both  $u$  and  $v$ .

# Tight cut

# Tight cut

Before the tight cut decomposition,

# Tight cut

Before the tight cut decomposition, we shall know, what a tight cut is.

# Tight cut

Before the tight cut decomposition, we shall know, what a tight cut is. Before that, what is a cut?



Before the tight cut decomposition, we shall know, what a tight cut is. Before that, what is a cut?

## Cut

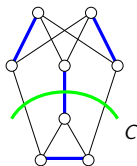
A subset  $C$  of the edge set  $E$  of a graph  $G$  is called a *cut* if  $C = \partial(X)$ , for nonempty subset  $X$  of the vertex set  $V$ .

# cut

Before the tight cut decomposition, we shall know, what a tight cut is. Before that, what is a cut?

## Cut

A subset  $C$  of the edge set  $E$  of a graph  $G$  is called a *cut* if  $C = \partial(X)$ , for nonempty subset  $X$  of the vertex set  $V$ .



An example of a cut  $C$

# Tight cut

Before the tight cut decomposition, we shall know, what a tight cut is:

# Tight cut

Before the tight cut decomposition, we shall know, what a tight cut is:

## Tight cut

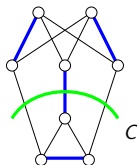
A cut  $C$  of a matching covered graph  $G$  is tight if  $|M \cap C| = 1$ , for every perfect matching  $M$  of  $G$ .

# Tight cut

Before the tight cut decomposition, we shall know, what a tight cut is:

## Tight cut

A cut  $C$  of a matching covered graph  $G$  is tight if  $|M \cap C| = 1$ , for every perfect matching  $M$  of  $G$ .



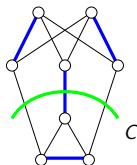
An example of a tight cut  $C$

# Tight cut

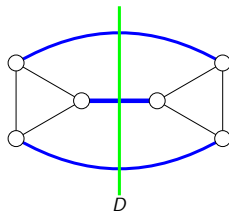
Before the tight cut decomposition, we shall know, what a tight cut is:

## Tight cut

A cut  $C$  of a matching covered graph  $G$  is tight if  $|M \cap C| = 1$ , for every perfect matching  $M$  of  $G$ .



An example of a tight cut  $C$



$D$ : A cut that is not tight

# Cuts and cut-contractions

## Cuts and cut-contractions



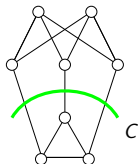
## Cuts and cut-contractions

Given any cut  $C := \partial(X)$  of a connected graph  $G$ , where  $X$  is a nonempty proper subset of  $V$ , we refer to the two graphs  $G/(X \rightarrow x)$  and  $G/(\bar{X} \rightarrow x)$  as the  $C$ -contractions of  $G$ , where  $\bar{X} := V - X$ .

# Cuts and cut-contractions

## Cuts and cut-contractions

Given any cut  $C := \partial(X)$  of a connected graph  $G$ , where  $X$  is a nonempty proper subset of  $V$ , we refer to the two graphs  $G/(X \rightarrow x)$  and  $G/(\bar{X} \rightarrow x)$  as the  $C$ -contractions of  $G$ , where  $\bar{X} := V - X$ .



The graph  $G$  with the cut  $C$  leads to two  $C$ -contractions, that are —  $K_4$  and  $K_{3,3}$ .

# Towards the tight cut decomposition

# Towards the tight cut decomposition

A small DIY for all of you:

# Towards the tight cut decomposition

A small DIY for all of you:

## Getting smaller MCGs thru tight cuts

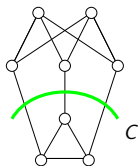
Let  $G$  be a graph and  $C$  be a tight cut of it. Suppose that  $G_1$  and  $G_2$  are the  $C$ -contractions of  $G$ . The graph  $G$  is matching covered iff each of  $G_1$  and  $G_2$  are matching covered.

# Towards the tight cut decomposition

A small DIY for all of you:

## Getting smaller MCGs thru tight cuts

Let  $G$  be a graph and  $C$  be a tight cut of it. Suppose that  $G_1$  and  $G_2$  are the  $C$ -contractions of  $G$ . The graph  $G$  is matching covered iff each of  $G_1$  and  $G_2$  are matching covered.



The graph  $G$  shown has a non-trivial tight cut  $C$ , that leads to two  $C$ -contractions, that are —  $K_4$  and  $K_{3,3}$ , both of which are free of tight cuts.

## Bricks and braces

For a matching covered graph that is free of non-trivial tight cuts, it is called a *brick* if it is non-bipartite and a *brace* if it is bipartite.

For example:  $K_4$  is a brick and  $K_{3,3}$  is a brace.

A natural question.



# A natural question.

Are the results of two TCD procedure unique?

A matching covered graph might have several tight cuts. Will they lead to different list of bricks and braces if we follow two different sequence of tight cut decomposition procedure?

# Lovász Tight cut decomposition theorem

# Lovász Tight cut decomposition theorem

The unique decomposition theorem: Lovász 1987

# Lovász Tight cut decomposition theorem

## The unique decomposition theorem: Lovász 1987

Any two applications of the tight cut decomposition procedure to a matching covered graph  $G$  produce the same list of bricks and braces, up to multiple edges.

# Lovász Tight cut decomposition theorem

## The unique decomposition theorem: Lovász 1987

Any two applications of the tight cut decomposition procedure to a matching covered graph  $G$  produce the same list of bricks and braces, up to multiple edges.

- ▶ The list underlying simple graphs of the bricks and the braces of a matching covered graph are its invariant.

# Why is it interesting?

# Why is it interesting?

- ▶ Several properties concerning problems in matching theory are preserved under tight cut decomposition thus reducing the problems to bricks and braces.
- ▶ For example, pfaffian orientations are useful in counting the number of PMs.

# Why is it interesting?

- ▶ Several properties concerning problems in matching theory are preserved under tight cut decomposition thus reducing the problems to bricks and braces.
- ▶ For example, pfaffian orientations are useful in counting the number of PMs.

## A cool result

A matching covered graph is pfaffian iff each of its bricks and braces is pfaffian.



# Enough! How to implement these?

# Enough! How to implement these?

## The implementation problem

# Enough! How to implement these?

## The implementation problem

Given a matching covered graph; either

1. deduce that it is a brick or a brace, or
2. find a nontrivial tight cut.

# Enough! How to implement these?

## The implementation problem

Given a matching covered graph; either

1. deduce that it is a brick or a brace, or
2. find a nontrivial tight cut.

It turns out that finding arbitrary tight cuts (if any) in a matching covered graph is not an easy task.

# How to do in polytime?

# How to do in polytime?

## Edmonds-Lovász-Pulleyblank 1982

If a matching covered graph has a nontrivial tight cut, then it has a non trivial tight cut that is either a barrier cut, or a two-separation cut.

# How to do in polytime?

## Edmonds-Lovász-Pulleyblank 1982

If a matching covered graph has a nontrivial tight cut, then it has a non trivial tight cut that is either a barrier cut, or a two-separation cut.

- ▶ We will see some example subsequently.

# How to do in polytime?

## Edmonds-Lovász-Pulleyblank 1982

If a matching covered graph has a nontrivial tight cut, then it has a non trivial tight cut that is either a barrier cut, or a two-separation cut.

- ▶ We will see some example subsequently.
- ▶ Note that, there might be cuts that are neither a barrier cut nor a 2-separation cut.



# How to do in polytime?

## Edmonds-Lovász-Pulleyblank 1982

If a matching covered graph has a nontrivial tight cut, then it has a non trivial tight cut that is either a barrier cut, or a two-separation cut.

- ▶ We will see some example subsequently.
- ▶ Note that, there might be cuts that are neither a barrier cut nor a 2-separation cut. Again, we will see examples as we go on.

# What is a barrier cut?

To recall:

# What is a barrier cut?

To recall: For a matchable graph  $G$ , a subset  $B$  of the vertex set  $V$  is a barrier if

$$|B| = o(G - B)$$

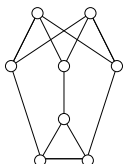
where  $o(H)$  denotes the # of odd components in graph  $H$ .

# What is a barrier cut?

To recall: For a matchable graph  $G$ , a subset  $B$  of the vertex set  $V$  is a barrier if

$$|B| = o(G - B)$$

where  $o(H)$  denotes the # of odd components in graph  $H$ .



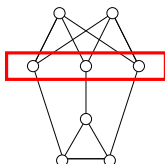
The set shown in red constitutes a barrier; the corresponding nontrivial odd comp. is shown in blue. The tight cut  $C$  shown in green is a nontrivial barrier cut.

# What is a barrier cut?

To recall: For a matchable graph  $G$ , a subset  $B$  of the vertex set  $V$  is a barrier if

$$|B| = o(G - B)$$

where  $o(H)$  denotes the # of odd components in graph  $H$ .



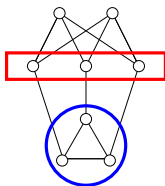
The set shown in red constitutes a barrier; the corresponding nontrivial odd comp. is shown in blue. The tight cut  $C$  shown in green is a nontrivial barrier cut.

# What is a barrier cut?

To recall: For a matchable graph  $G$ , a subset  $B$  of the vertex set  $V$  is a barrier if

$$|B| = o(G - B)$$

where  $o(H)$  denotes the # of odd components in graph  $H$ .



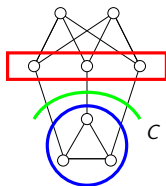
The set shown in red constitutes a barrier; the corresponding nontrivial odd comp. is shown in blue. The tight cut  $C$  shown in green is a nontrivial barrier cut.

# What is a barrier cut?

To recall: For a matchable graph  $G$ , a subset  $B$  of the vertex set  $V$  is a barrier if

$$|B| = o(G - B)$$

where  $o(H)$  denotes the # of odd components in graph  $H$ .

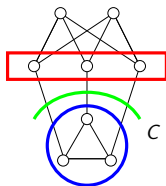


# What is a barrier cut?

To recall: For a matchable graph  $G$ , a subset  $B$  of the vertex set  $V$  is a barrier if

$$|B| = o(G - B)$$

where  $o(H)$  denotes the # of odd components in graph  $H$ .



The set shown in red constitutes a barrier; the corresponding nontrivial odd comp. is shown in blue. The tight cut  $C$  shown in green is a nontrivial barrier cut.



# What if there are no nontrivial barrier cuts?

# What if there are no nontrivial barrier cuts?

## Bicritical graph

A matching covered graph is *bicritical* if it is free of nontrivial barriers,

# What if there are no nontrivial barrier cuts?

## Bicritical graph

A matching covered graph is *bicritical* if it is free of nontrivial barriers, that is —  $\emptyset$  and singleton vertex set are the only barriers of a bicritical graph.

Clearly, bricks are bicritical, but is each bicritical graph a brick?

# What if there are no nontrivial barrier cuts?

## Bicritical graph

A matching covered graph is *bicritical* if it is free of nontrivial barriers, that is —  $\emptyset$  and singleton vertex set are the only barriers of a bicritical graph.

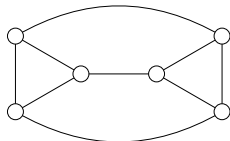
Clearly, bricks are bicritical, but is each bicritical graph a brick?

# What if there are no nontrivial barrier cuts?

## Bicritical graph

A matching covered graph is *bicritical* if it is free of nontrivial barriers, that is —  $\emptyset$  and singleton vertex set are the only barriers of a bicritical graph.

Clearly, bricks are bicritical, but is each bicritical graph a brick?



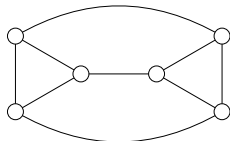
A bicritical graph that is a brick

# What if there are no nontrivial barrier cuts?

## Bicritical graph

A matching covered graph is *bicritical* if it is free of nontrivial barriers, that is —  $\emptyset$  and singleton vertex set are the only barriers of a bicritical graph.

Clearly, bricks are bicritical, but is each bicritical graph a brick?



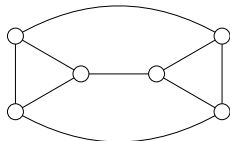
A bicritical graph that is a brick

# What if there are no nontrivial barrier cuts?

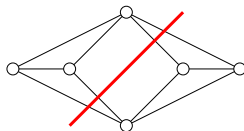
## Bicritical graph

A matching covered graph is *bicritical* if it is free of nontrivial barriers, that is —  $\emptyset$  and singleton vertex set are the only barriers of a bicritical graph.

Clearly, bricks are bicritical, but is each bicritical graph a brick?



A bicritical graph that is a brick



A bicritical graph that is not a brick

## 2-separation cut



## 2-separation cut

- ▶ Consider a matching covered graph  $G$

## 2-separation cut

- ▶ Consider a matching covered graph  $G$
- ▶ As the name suggest, we have a 2 vertex cut  $\{u, v\}$  in  $G$ .

## 2-separation cut

- ▶ Consider a matching covered graph  $G$
- ▶ As the name suggest, we have a 2 vertex cut  $\{u, v\}$  in  $G$ .
- ▶ What you can say about the # of odd components of  $G - u - v$ ?

## 2-separation cut

- ▶ Consider a matching covered graph  $G$
- ▶ As the name suggest, we have a 2 vertex cut  $\{u, v\}$  in  $G$ .
- ▶ What you can say about the # of odd components of  $G - u - v$ ?
- ▶ By Tutte's theorem, either it is 2 or 0.

## 2-separation cut

- ▶ Consider a matching covered graph  $G$
- ▶ As the name suggest, we have a 2 vertex cut  $\{u, v\}$  in  $G$ .
- ▶ What you can say about the  $\#$  of odd components of  $G - u - v$ ?
- ▶ By Tutte's theorem, either it is 2 or 0.
- ▶ If it is two and we have a nontrivial odd component, then we have a nontrivial barrier cut.

## 2-separation cut

- ▶ Consider a matching covered graph  $G$
- ▶ As the name suggest, we have a 2 vertex cut  $\{u, v\}$  in  $G$ .
- ▶ What you can say about the  $\#$  of odd components of  $G - u - v$ ?
- ▶ By Tutte's theorem, either it is 2 or 0.
- ▶ If it is two and we have a nontrivial odd component, then we have a nontrivial barrier cut.
- ▶ If it is zero, then we will get a bunch of even components (at least two).

## 2-separation cut

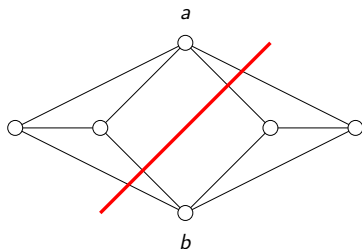
- ▶ Consider a matching covered graph  $G$
- ▶ As the name suggest, we have a 2 vertex cut  $\{u, v\}$  in  $G$ .
- ▶ What you can say about the  $\#$  of odd components of  $G - u - v$ ?
- ▶ By Tutte's theorem, either it is 2 or 0.
- ▶ If it is two and we have a nontrivial odd component, then we have a nontrivial barrier cut.
- ▶ If it is zero, then we will get a bunch of even components (at least two).

## 2-separation cut

Bricks are free of 2 separation cuts as well. Moreover, they are 3-connected.



## 2-separation cut



The illustration of a 2-separation cut

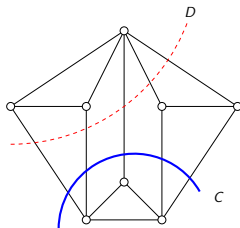
Bricks are free of 2 separation cuts as well. Moreover, they are 3-connected.

# Are tight cuts always a barrier cut or a 2-sep cut?

- ▶ No,

# Are tight cuts always a barrier cut or a 2-sep cut?

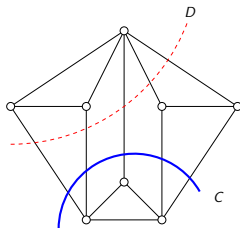
- ▶ No, not all tight cuts are either barrier cuts or 2-separation cuts.



**Figure:** The cut  $C$  (shown in bold blue) is neither a barrier cut nor a 2-separation cut, but  $D$  (shown in dashed red) is a 2-separation cut

# Are tight cuts always a barrier cut or a 2-sep cut?

- ▶ No, not all tight cuts are either barrier cuts or 2-separation cuts.

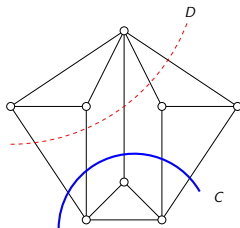


**Figure:** The cut  $C$  (shown in bold blue) is neither a barrier cut nor a 2-separation cut, but  $D$  (shown in dashed red) is a 2-separation cut

- ▶ Finding an arbitrary tight cut is difficult, but finding a barrier cut or a 2-separation cut is polytime computable.

# Are tight cuts always a barrier cut or a 2-sep cut?

- ▶ No, not all tight cuts are either barrier cuts or 2-separation cuts.



**Figure:** The cut  $C$  (shown in bold blue) is neither a barrier cut nor a 2-separation cut, but  $D$  (shown in dashed red) is a 2-separation cut

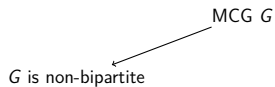
- ▶ Finding an arbitrary tight cut is difficult, but finding a barrier cut or a 2-separation cut is polytime computable.
- ▶ The ELP theorem guarantees the existence of either a nontrivial barrier or a non trivial 2-separation cut, if at all the matching covered graph has a nontrivial tight cut.

# Implementing tight cut decomposition procedure

# Implementing tight cut decomposition procedure

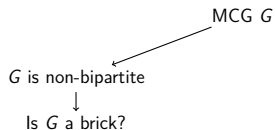
MCG  $G$

# Implementing tight cut decomposition procedure

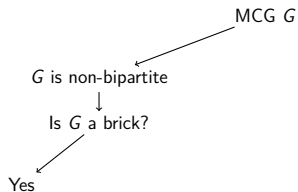




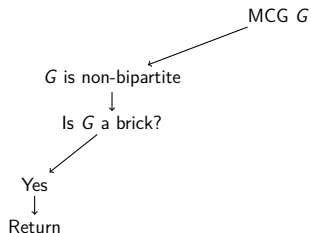
# Implementing tight cut decomposition procedure



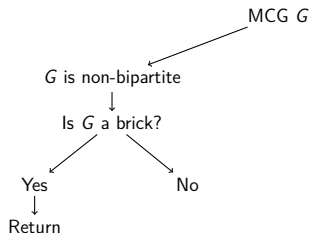
# Implementing tight cut decomposition procedure



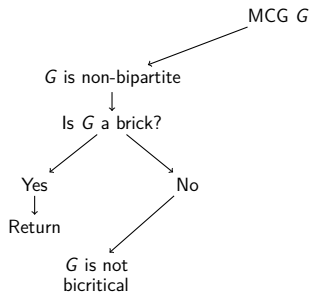
# Implementing tight cut decomposition procedure



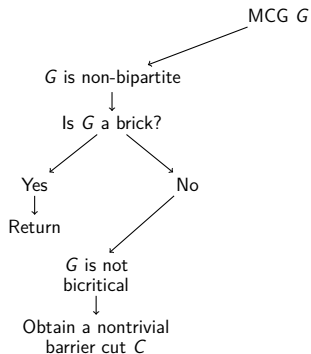
# Implementing tight cut decomposition procedure



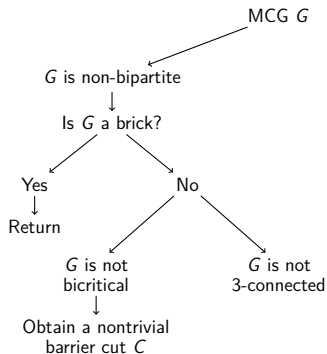
# Implementing tight cut decomposition procedure



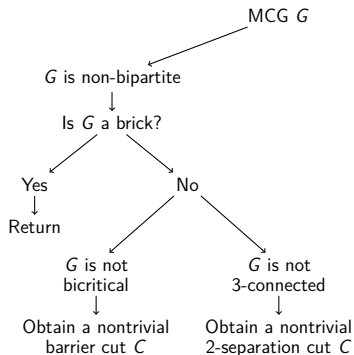
# Implementing tight cut decomposition procedure



# Implementing tight cut decomposition procedure

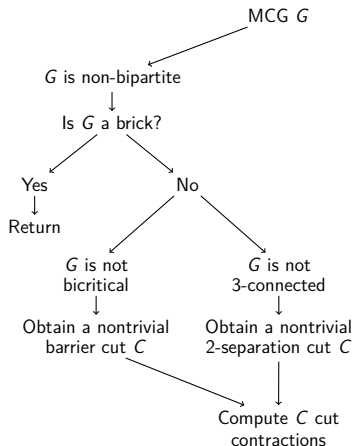


# Implementing tight cut decomposition procedure

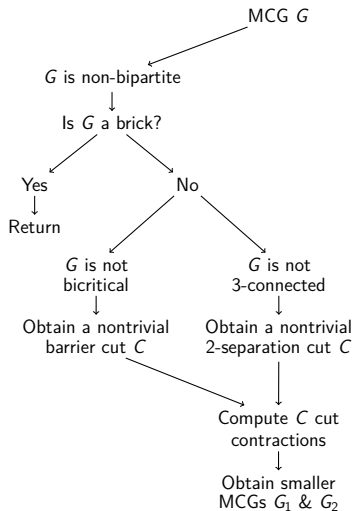




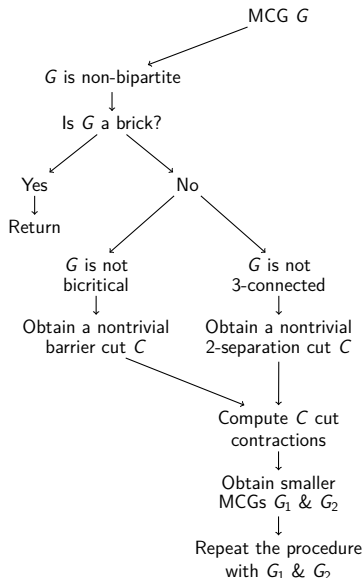
# Implementing tight cut decomposition procedure



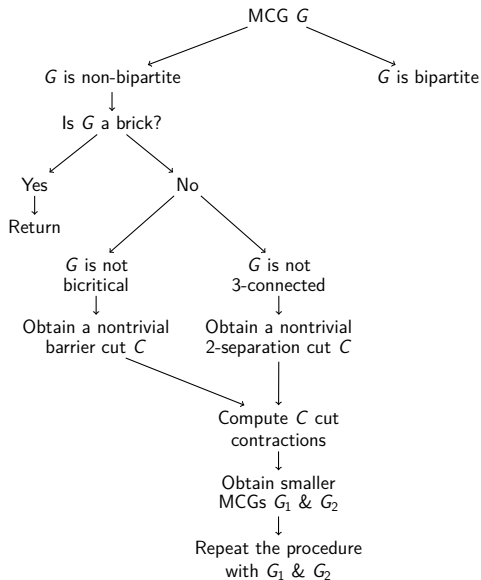
# Implementing tight cut decomposition procedure



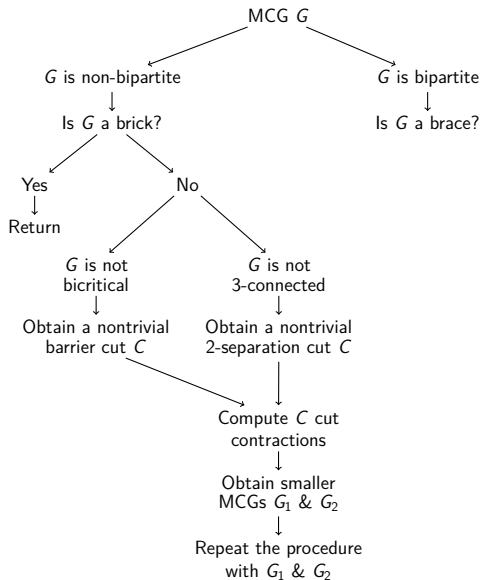
# Implementing tight cut decomposition procedure



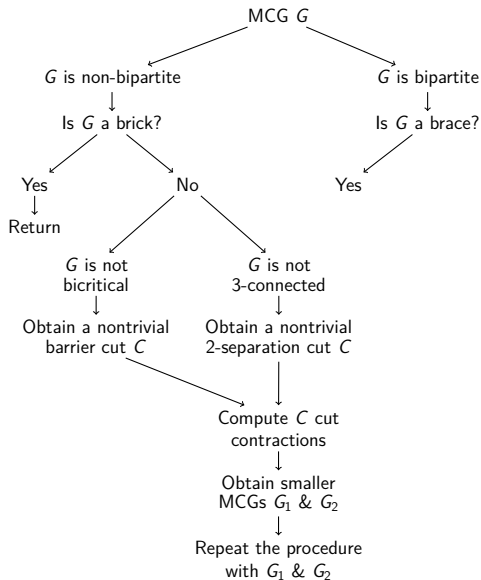
# Implementing tight cut decomposition procedure



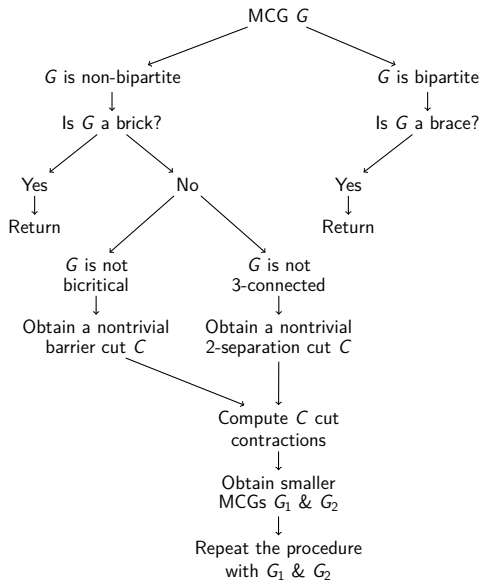
# Implementing tight cut decomposition procedure



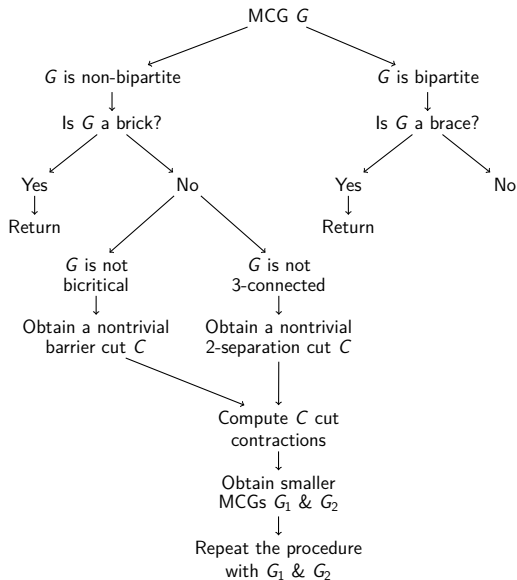
# Implementing tight cut decomposition procedure



# Implementing tight cut decomposition procedure

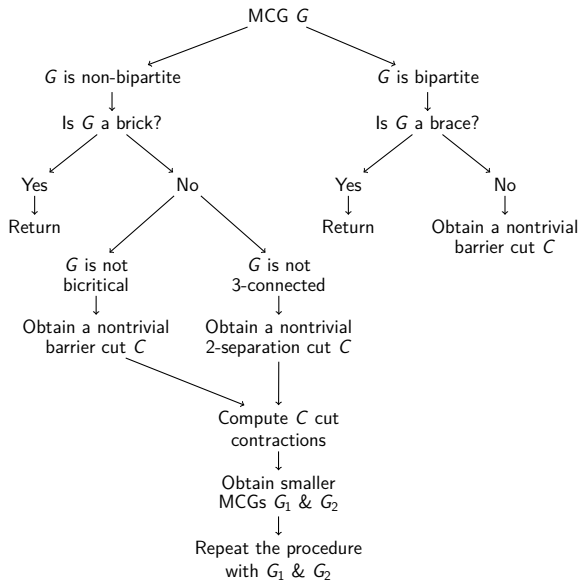


# Implementing tight cut decomposition procedure

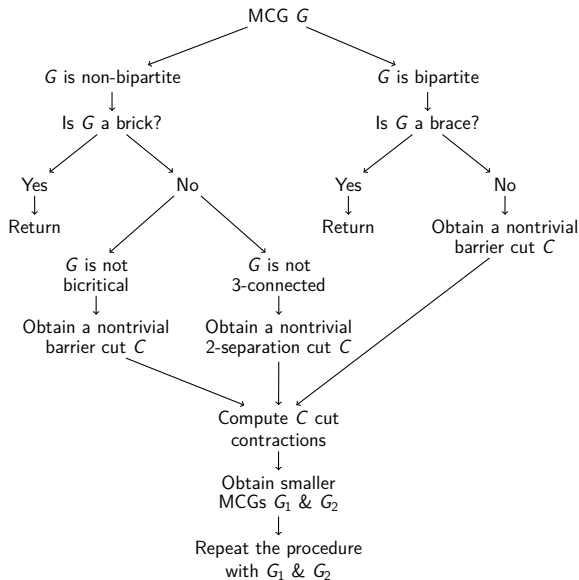




# Implementing tight cut decomposition procedure



# Implementing tight cut decomposition procedure



# Notable families of bricks and braces

# Notable families of bricks and braces

These families play significant roles in the works of McCuaig and Norine and Thomas.

- ▶ Circular Ladder graph

# Notable families of bricks and braces

These families play significant roles in the works of McCuaig and Norine and Thomas.

- ▶ Circular Ladder graph
- ▶ Wheel graph

# Notable families of bricks and braces

These families play significant roles in the works of McCuaig and Norine and Thomas.

- ▶ Circular Ladder graph
- ▶ Wheel graph
- ▶ Möbius Ladder graph

# Notable families of bricks and braces

These families play significant roles in the works of McCuaig and Norine and Thomas.

- ▶ Circular Ladder graph
- ▶ Wheel graph
- ▶ Möbius Ladder graph
- ▶ Biwheel graph

# Notable families of bricks and braces

These families play significant roles in the works of McCuaig and Norine and Thomas.

- ▶ Circular Ladder graph
- ▶ Wheel graph
- ▶ Möbius Ladder graph
- ▶ Biwheel graph
- ▶ Truncated Biwheel graph



# Notable families of bricks and braces

These families play significant roles in the works of McCuaig and Norine and Thomas.

- ▶ Circular Ladder graph
- ▶ Wheel graph
- ▶ Möbius Ladder graph
- ▶ Biwheel graph
- ▶ Truncated Biwheel graph
- ▶ Staircase graph

# Notable families of bricks and braces

These families play significant roles in the works of McCuaig and Norine and Thomas.

- ▶ Circular Ladder graph
- ▶ Wheel graph
- ▶ Möbius Ladder graph
- ▶ Biwheel graph
- ▶ Truncated Biwheel graph
- ▶ Staircase graph

## Notable families: Circular Ladder graph

# Notable families: Circular Ladder graph

Circular Ladder graph

# Notable families: Circular Ladder graph

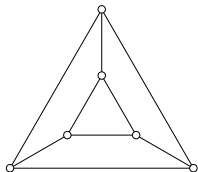
## Circular Ladder graph

The *Circular ladder graph*, aka *Prism graph*  $\mathbb{P}_{2n}$ , for  $n \geq 3$ , is the graph obtained from two disjoint cycles

$$u_1 u_2 u_3 \dots u_n u_1 \text{ and } v_1 v_2 \dots v_n v_1$$

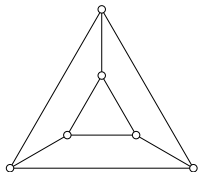
of length  $n$  by the addition of the  $n$  edges  $u_i v_i$ ,  $i = 1, 2, \dots, n$ .

# Notable families: Circular Ladder graph

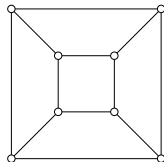


(a) CircularLadderGraph(3)

# Notable families: Circular Ladder graph

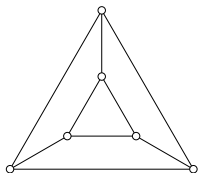


(a) CircularLadderGraph(3)

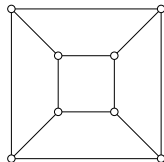


(b) CircularLadderGraph(4)

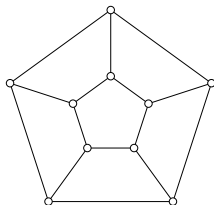
# Notable families: Circular Ladder graph



(a) CircularLadderGraph(3)



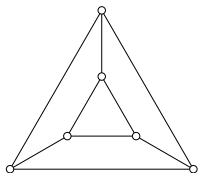
(b) CircularLadderGraph(4)



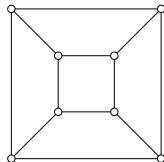
(c) CircularLadderGraph(5)



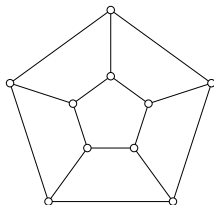
# Notable families: Circular Ladder graph



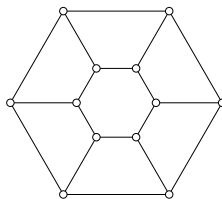
(a) CircularLadderGraph(3)



(b) CircularLadderGraph(4)

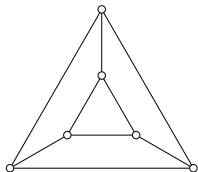


(c) CircularLadderGraph(5)

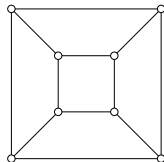


(d) CircularLadderGraph(6)

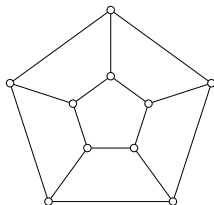
# Notable families: Circular Ladder graph



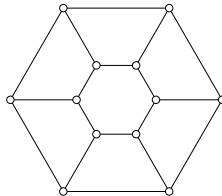
(a) CircularLadderGraph(3)



(b) CircularLadderGraph(4)



(c) CircularLadderGraph(5)



(d) CircularLadderGraph(6)

Figure: The family of circular ladder graphs

# Notable families: Wheel graph

# Notable families: Wheel graph

Wheel graph

# Notable families: Wheel graph

## Wheel graph

The *Wheel graph*  $\mathbb{W}_n$ , for  $n \geq 4$ , is the graph obtained from a cycle

$$v_1 v_2 \dots v_{n-1} v_1$$

of length  $n - 1$ , called the *rim* of  $\mathbb{W}_n$ , by the addition a universal vertex, called *hub*  $h$ .

# Notable families: Wheel graph

## Wheel graph

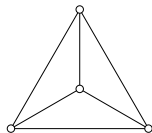
The *Wheel graph*  $\mathbb{W}_n$ , for  $n \geq 4$ , is the graph obtained from a cycle

$$v_1 v_2 \dots v_{n-1} v_1$$

of length  $n - 1$ , called the *rim* of  $\mathbb{W}_n$ , by the addition a universal vertex, called *hub*  $h$ . Note that wheels on an even number of vertices, aka  $W_{2k}$  for some  $k \geq 2$ , are matching covered — in fact, bricks.

# Notable families: Wheel graph

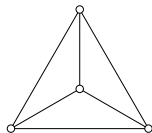
# Notable families: Wheel graph



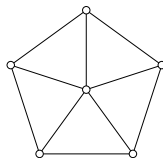
(a)  $\text{WheelGraph}(4)$



# Notable families: Wheel graph

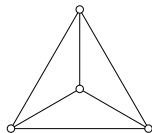


(a) WheelGraph(4)

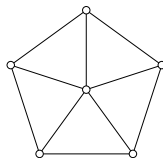


(b) WheelGraph(6)

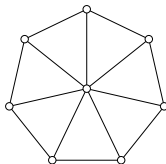
# Notable families: Wheel graph



(a) WheelGraph(4)

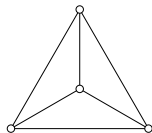


(b) WheelGraph(6)

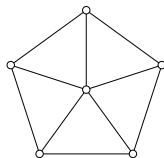


(c) WheelGraph(8)

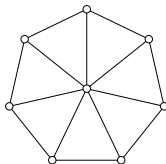
# Notable families: Wheel graph



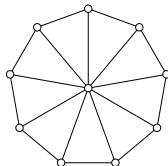
(a) WheelGraph(4)



(b) WheelGraph(6)

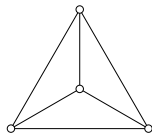


(c) WheelGraph(8)

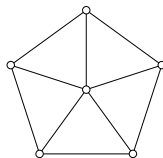


(d) WheelGraph(10)

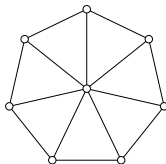
# Notable families: Wheel graph



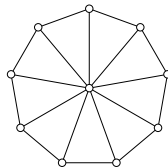
(a) WheelGraph(4)



(b) WheelGraph(6)



(c) WheelGraph(8)



(d) WheelGraph(10)

Figure: The family of wheel graphs, that are matching covered

# Notable families: Möbius Ladder graph

# Notable families: Möbius Ladder graph

Möbius Ladder graph

# Notable families: Möbius Ladder graph

## Möbius Ladder graph

The *Möbius ladder graph*  $\mathbb{M}_{2n}$ , for  $n \geq 2$ , is the graph obtained from a cycle

$$v_1 v_2 \dots v_{2n} v_1$$

of length  $2n$  by the addition of the  $n$  chords  $v_i v_{i+n}$ , for  $1 \leq i \leq n$ , joining antipodal pairs of vertices of the cycle.

# Notable families: Möbius Ladder graph

## Möbius Ladder graph

The *Möbius ladder graph*  $\mathbb{M}_{2n}$ , for  $n \geq 2$ , is the graph obtained from a cycle

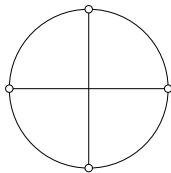
$$v_1 v_2 \dots v_{2n} v_1$$

of length  $2n$  by the addition of the  $n$  chords  $v_i v_{i+n}$ , for  $1 \leq i \leq n$ , joining antipodal pairs of vertices of the cycle.



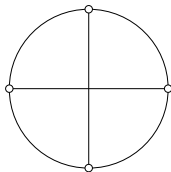
# Notable families: Möbius Ladder graph

# Notable families: Möbius Ladder graph

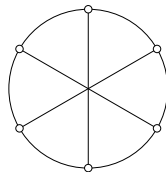


(a) MöbiusLadderGraph(2)

# Notable families: Möbius Ladder graph

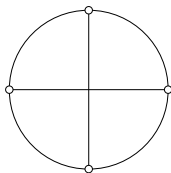


(a) MöbiusLadderGraph(2)

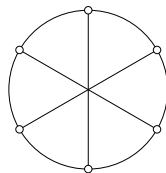


(b) MöbiusLadderGraph(3)

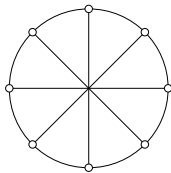
# Notable families: Möbius Ladder graph



(a) MöbiusLadderGraph(2)

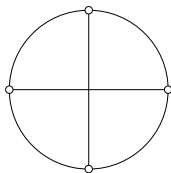


(b) MöbiusLadderGraph(3)

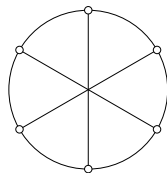


(c) MöbiusLadderGraph(4)

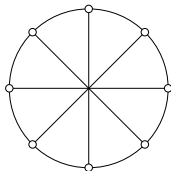
# Notable families: Möbius Ladder graph



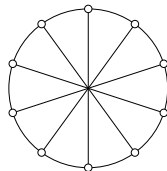
(a) MöbiusLadderGraph(2)



(b) MöbiusLadderGraph(3)

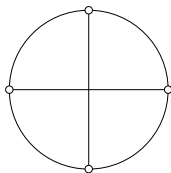


(c) MöbiusLadderGraph(4)

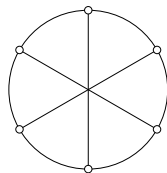


(d) MöbiusLadderGraph(5)

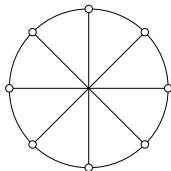
# Notable families: Möbius Ladder graph



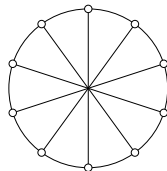
(a) MöbiusLadderGraph(2)



(b) MöbiusLadderGraph(3)



(c) MöbiusLadderGraph(4)



(d) MöbiusLadderGraph(5)

Figure: The family of möbius ladder graphs

# Notable families: Biwheel graph

# Notable families: Biwheel graph

Biwheel graph



# Notable families: Biwheel graph

## Biwheel graph

The *Biwheel graph*  $\mathbb{B}_{2n}$ , for  $n \geq 4$ , is the bipartite graph obtained from a cycle

$$v_1 v_2 \dots v_{2n-2} v_1$$

of length  $2n - 2$ , called the *rim* of  $\mathbb{B}_{2n}$ , by the addition of two vertices,  $h_1$  and  $h_2$ , called the *hubs* of  $\mathbb{B}_{2n}$ , and by the addition of edges  $h_1 v_1, h_1 v_3, \dots, h_1 v_{2n-3}$  and edges  $h_2 v_2, h_2 v_4, \dots, h_2 v_{2n-2}$ .

# Notable families: Biwheel graph

## Biwheel graph

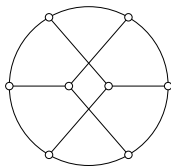
The *Biwheel graph*  $\mathbb{B}_{2n}$ , for  $n \geq 4$ , is the bipartite graph obtained from a cycle

$$v_1 v_2 \dots v_{2n-2} v_1$$

of length  $2n - 2$ , called the *rim* of  $\mathbb{B}_{2n}$ , by the addition of two vertices,  $h_1$  and  $h_2$ , called the *hubs* of  $\mathbb{B}_{2n}$ , and by the addition of edges  $h_1 v_1, h_1 v_3, \dots, h_1 v_{2n-3}$  and edges  $h_2 v_2, h_2 v_4, \dots, h_2 v_{2n-2}$ .

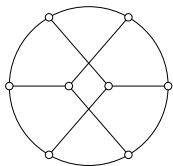
# Notable families: Biwheel graph

# Notable families: Biwheel graph

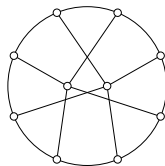


(a) BiwheelGraph(8)

# Notable families: Biwheel graph

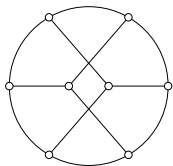


(a) BiwheelGraph(8)

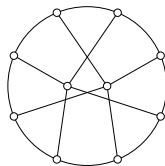


(b) BiwheelGraph(10)

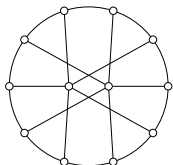
# Notable families: Biwheel graph



(a) BiwheelGraph(8)

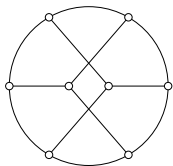


(b) BiwheelGraph(10)

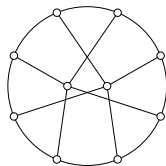


(c) BiwheelGraph(12)

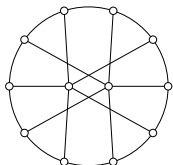
# Notable families: Biwheel graph



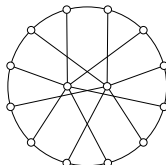
(a) BiwheelGraph(8)



(b) BiwheelGraph(10)

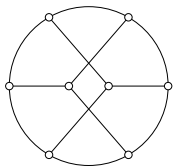


(c) BiwheelGraph(12)

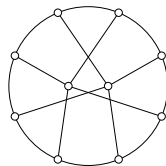


(d) BiwheelGraph(14)

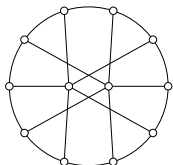
# Notable families: Biwheel graph



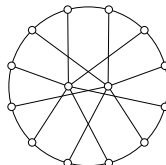
(a) BiwheelGraph(8)



(b) BiwheelGraph(10)



(c) BiwheelGraph(12)



(d) BiwheelGraph(14)

Figure: The family of biwheel graphs



# Notable families: Truncated Biwheel graph

# Notable families: Truncated Biwheel graph

Truncated Biwheel graph

# Notable families: Truncated Biwheel graph

## Truncated Biwheel graph

The *Truncated biwheel graph*  $\mathbb{T}_{2n}$ , for  $n \geq 3$ , is the graph obtained from a path  $v_1 v_2 \dots v_{2n-2}$  of length  $2n - 3$ , by the addition of two vertices,  $h_1$  and  $h_2$ , and by the addition of edges  $h_1 v_1, h_1 v_3, \dots, h_1 v_{2n-3}$ , edges  $h_2 v_2, h_2 v_4, \dots, h_2 v_{2n-2}$  and edges  $h_1 v_{2n-2}$  and  $h_2 v_1$ .

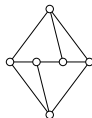
# Notable families: Truncated Biwheel graph

## Truncated Biwheel graph

The *Truncated biwheel graph*  $\mathbb{T}_{2n}$ , for  $n \geq 3$ , is the graph obtained from a path  $v_1 v_2 \dots v_{2n-2}$  of length  $2n - 3$ , by the addition of two vertices,  $h_1$  and  $h_2$ , and by the addition of edges  $h_1 v_1, h_1 v_3, \dots, h_1 v_{2n-3}$ , edges  $h_2 v_2, h_2 v_4, \dots, h_2 v_{2n-2}$  and edges  $h_1 v_{2n-2}$  and  $h_2 v_1$ .

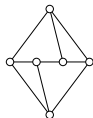
# Notable families: Truncated Biwheel graph

# Notable families: Truncated Biwheel graph

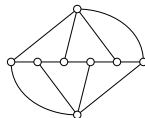


(a) `TruncatedBiwheelGraph(3)`

# Notable families: Truncated Biwheel graph

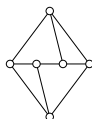


(a) TruncatedBiwheelGraph(3)

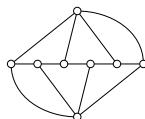


(b) TruncatedBiwheelGraph(4)

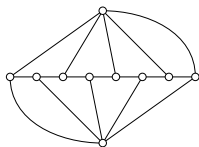
# Notable families: Truncated Biwheel graph



(a) TruncatedBiwheelGraph(3)



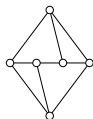
(b) TruncatedBiwheelGraph(4)



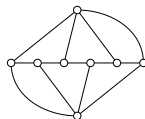
(c) TruncatedBiwheelGraph(5)



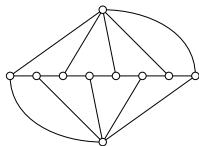
# Notable families: Truncated Biwheel graph



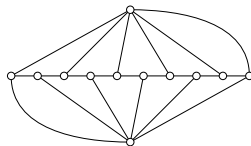
(a) TruncatedBiwheelGraph(3)



(b) TruncatedBiwheelGraph(4)

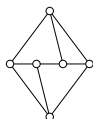


(c) TruncatedBiwheelGraph(5)

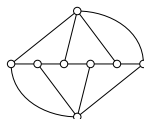


(d) TruncatedBiwheelGraph(6)

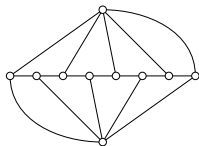
# Notable families: Truncated Biwheel graph



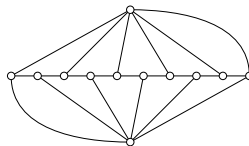
(a) TruncatedBiwheelGraph(3)



(b) TruncatedBiwheelGraph(4)



(c) TruncatedBiwheelGraph(5)



(d) TruncatedBiwheelGraph(6)

Figure: The family of truncated biwheel graphs

# Notable families: Staircase graph

# Notable families: Staircase graph

Staircase graph

# Notable families: Staircase graph

## Staircase graph

Consider the Ladder graph  $\mathbb{L}_{2n-2}$  obtained from two disjoint paths  $u_1 u_2 \dots u_{n-1}$  and  $v_1 v_2 \dots v_{n-1}$  by adding, for  $1 \leq i \leq n-1$ , an edge joining  $u_i$  and  $v_i$ . For  $n \geq 3$ , the *Staircase graph*  $\mathbb{S}_{2n}$  is the graph obtained from  $\mathbb{L}_{2n-2}$  by adding two new vertices  $x$  and  $y$ , and then joining  $x$  to  $u_1$  and  $v_1$ , the vertex  $y$  to  $u_{n-1}$  and  $v_{n-1}$ , and  $x$  and  $y$  to each other.

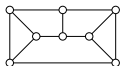
# Notable families: Staircase graph

## Staircase graph

Consider the Ladder graph  $\mathbb{L}_{2n-2}$  obtained from two disjoint paths  $u_1 u_2 \dots u_{n-1}$  and  $v_1 v_2 \dots v_{n-1}$  by adding, for  $1 \leq i \leq n-1$ , an edge joining  $u_i$  and  $v_i$ . For  $n \geq 3$ , the *Staircase graph*  $\mathbb{S}_{2n}$  is the graph obtained from  $\mathbb{L}_{2n-2}$  by adding two new vertices  $x$  and  $y$ , and then joining  $x$  to  $u_1$  and  $v_1$ , the vertex  $y$  to  $u_{n-1}$  and  $v_{n-1}$ , and  $x$  and  $y$  to each other.

# Notable families: Staircase graph

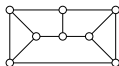
# Notable families: Staircase graph



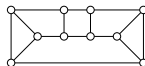
(a) StaircaseGraph(4)



# Notable families: Staircase graph

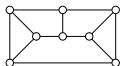


(a) StaircaseGraph(4)

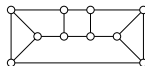


(b) StaricaseGraph(5)

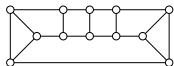
# Notable families: Staircase graph



(a) StaircaseGraph(4)

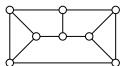


(b) StaircaseGraph(5)

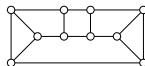


(c) StaircaseGraph(6)

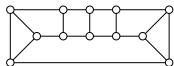
# Notable families: Staircase graph



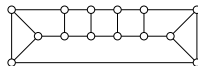
(a) StaircaseGraph(4)



(b) StaircaseGraph(5)

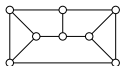


(c) StaircaseGraph(6)

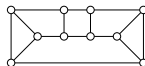


(d) StaircaseGraph(7)

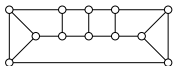
# Notable families: Staircase graph



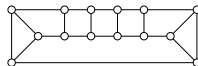
(a) StaircaseGraph(4)



(b) StaircaseGraph(5)



(c) StaircaseGraph(6)



(d) StaircaseGraph(7)

Figure: The family of staircase graphs

# An overview

# An overview

Module	Method	Status
--------	--------	--------

# An overview

Module		Method	Status
A	Fundamentals	M_alternating_tree_mark()	Implemented
		is_matching_covered()	Implemented
		is_bicritical()	Implemented

# An overview

Module		Method	Status
A	Fundamentals	M_alternating_tree_mark()	Implemented
		is_matching_covered()	Implemented
		is_bicritical()	Implemented
B	Canonical Partition	maximal_barrier()	Implemented
		canonical_partition()	Implemented



# An overview

Module		Method	Status
A	Fundamentals	M_alternating_tree_mark()	Implemented
		is_matching_covered()	Implemented
		is_bicritical()	Implemented
B	Canonical Partition	maximal_barrier()	Implemented
		canonical_partition()	Implemented
C	Tight cut decomposition	is_brick()	Implemented
		is_brace()	Implemented
		tight_cut_decomposition()	Implemented
		bricks_and_braces()	Implemented
		number_of_bricks()	Implemented
		number_of_braces()	Implemented
		number_of_petersen_bricks()	Implemented

## An overview (continued)

## An overview (continued)

Module	Method	Status
--------	--------	--------

## An overview (continued)

Module		Method	Status
D	Notable Families	MöbiusLadderGraph()	Implemented
		StaircaseGraph()	Implemented
		BiwheelGraph()	Ongoing
		TruncatedBiwheelGraph()	Implemented

## An overview (continued)

Module		Method	Status
D	Notable Families	MöbiusLadderGraph()	Implemented
		StaircaseGraph()	Implemented
		BiwheelGraph()	Ongoing
		TruncatedBiwheelGraph()	Implemented
E	Dependency relation	is_removable_edge()	Implemented
		removable_edges()	Implemented
		is_removable_doubleton()	Implemented
		removable_doubletons()	Implemented

## An overview (continued)

## An overview (continued)

Module	Method	Status
--------	--------	--------

## An overview (continued)

Module		Method	Status
F	Ear decomposition	retract()	Ongoing
		matching_covered_ear_	To be done
		decomposition()	To be done
		optimal_ear()	To be done
		optimal_ear_	To be done
		decomposition()	



## An overview (continued)

Module		Method	Status
F	Ear decomposition	retract()	Ongoing
		matching_covered_ear_decomposition()	To be done
		optimal_ear()	To be done
		optimal_ear_decomposition()	To be done
G	Brick and brace generation	is_strictly_thin_edges()	To be done
		mccuaig_brace_decomposition()	To be done
		norine_thomas_brick_decomposition()	To be done

# What's next?

# What's next?

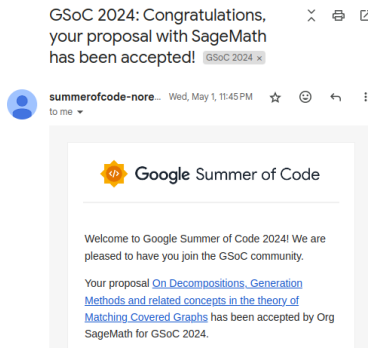
- ▶ One major goal is to make these implementations accessible for everyone.

# What's next?

- ▶ One major goal is to make these implementations accessible for everyone.
- ▶ Proposal to SageMath (via Google Summer of Code 2024).

# What's next?

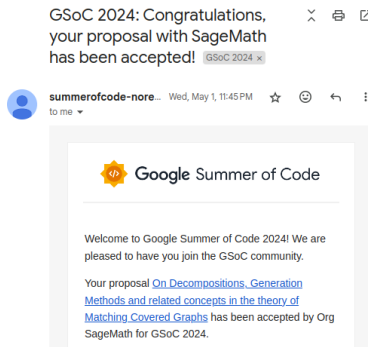
- ▶ One major goal is to make these implementations accessible for everyone.



- ▶ Proposal to SageMath (via Google Summer of Code 2024).

# What's next?

- ▶ One major goal is to make these implementations accessible for everyone.



- ▶ Should be accessible by everyone from 26th of August 2024.

# Yes!

Yes! you folks made my day.

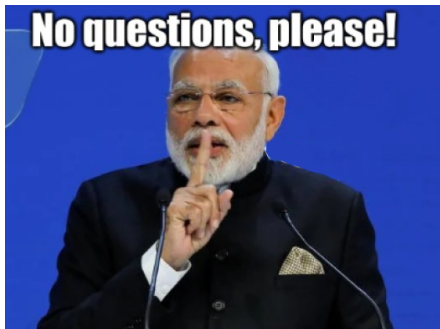




Yes! you folks made my day.



Yes! you folks made my day.



Yes! you folks made my day.



Yes! you folks made my day.



thank you!

