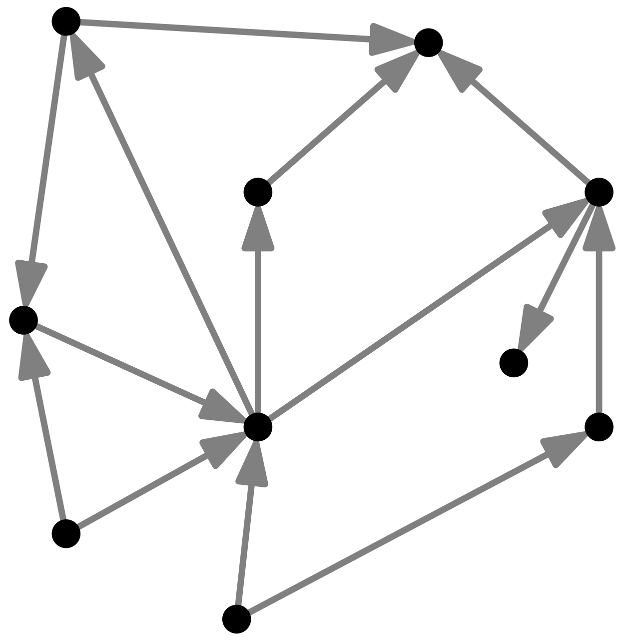
Parameterized and Approximation Algorithms for the Maximum Bimodal Subgraph Problem

GD 2023, Palermo

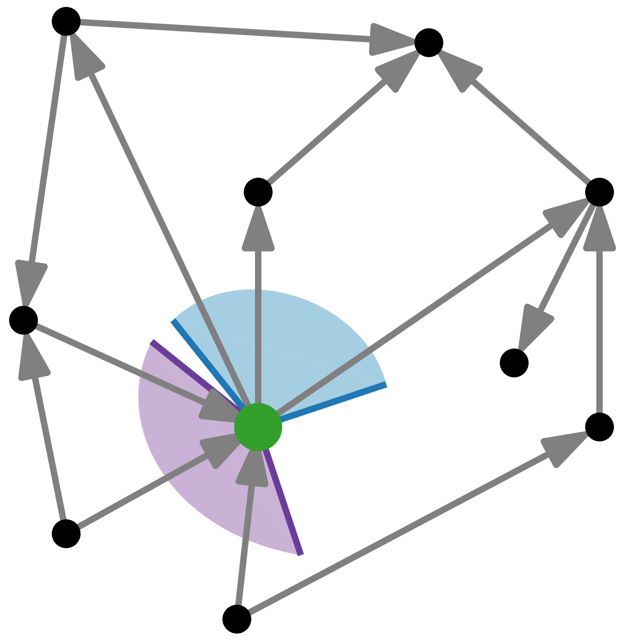
Walter Didimo[1] Fedor V. Fomin[2] Petr A. Golovach[2] Tanmay Inamdar[2] Stephen Kobourov[3] **M. Diana Sieper[4]**

[1] University of Perugia, Italy
[2] University of Bergen, Norway
[3] University of Arizona, USA
[4] University of Würzburg, Germany

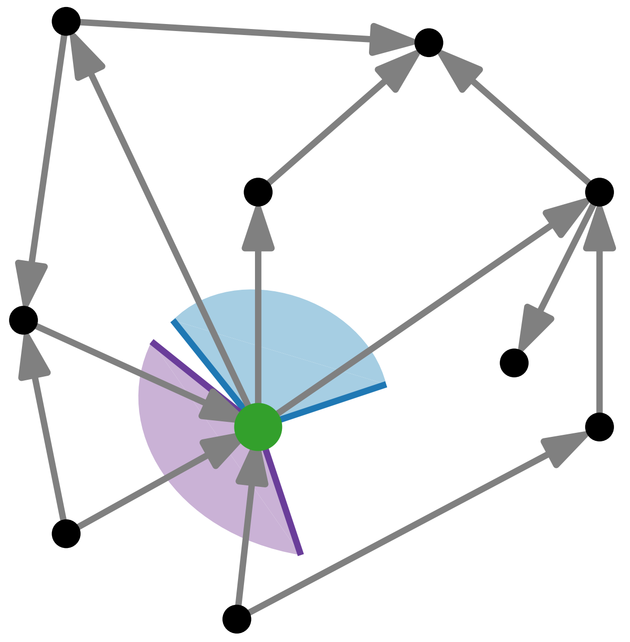
Bimodality



Bimodality

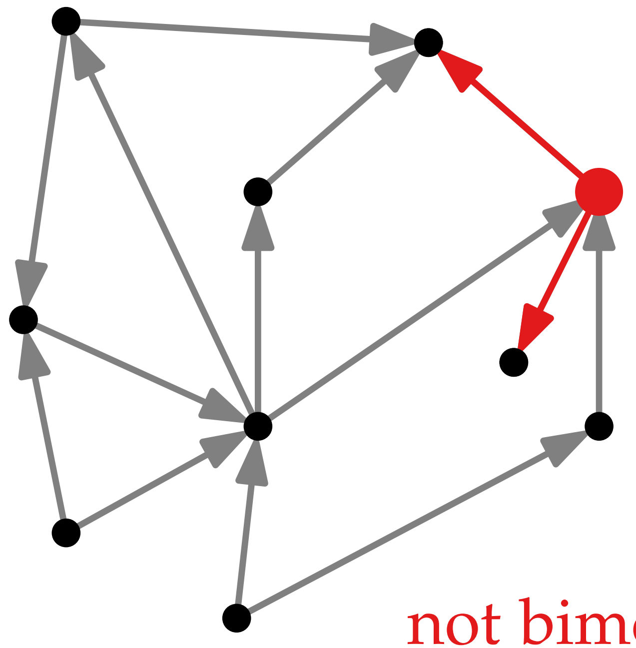


Bimodality



Bimodal vertex: All outgoing (incoming) edges are consecutive.

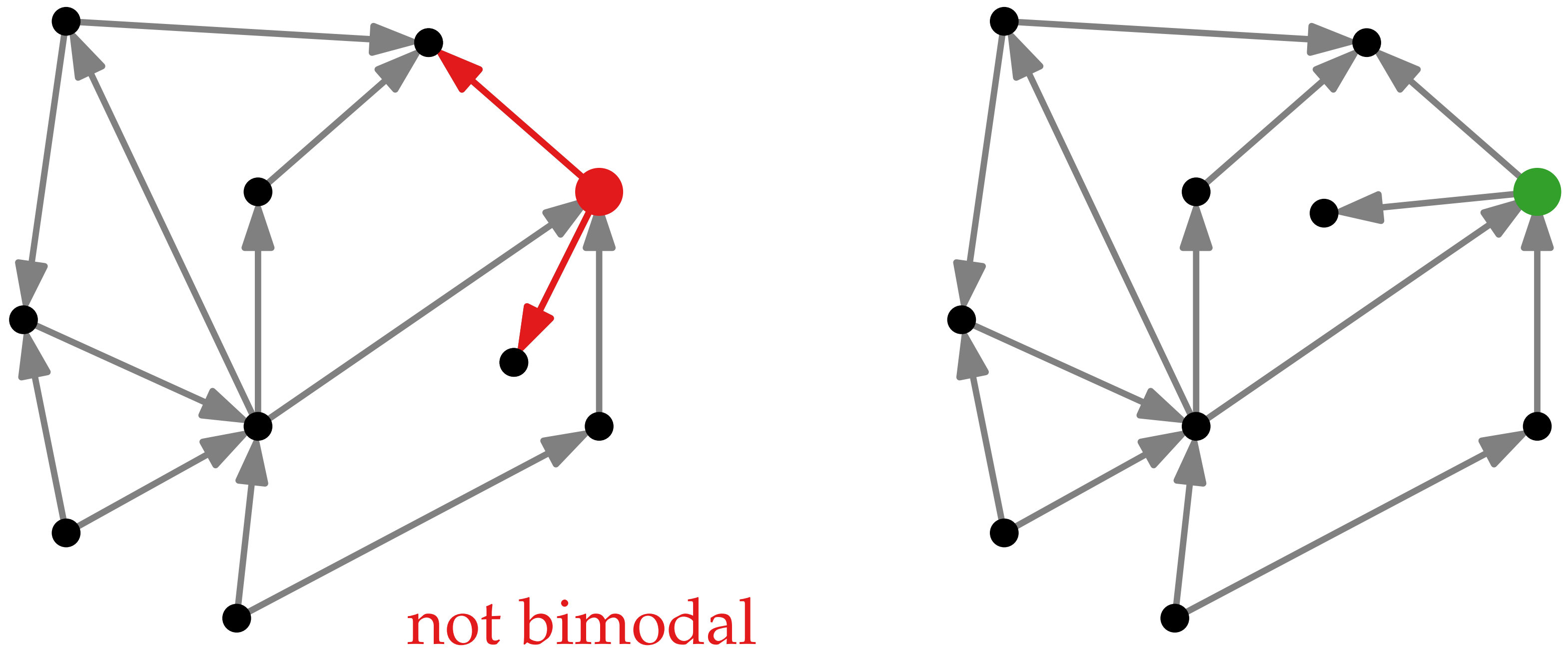
Bimodality



Bimodal vertex: All outgoing (incoming) edges are consecutive.

Bimodal Graph: Every vertex is bimodal.

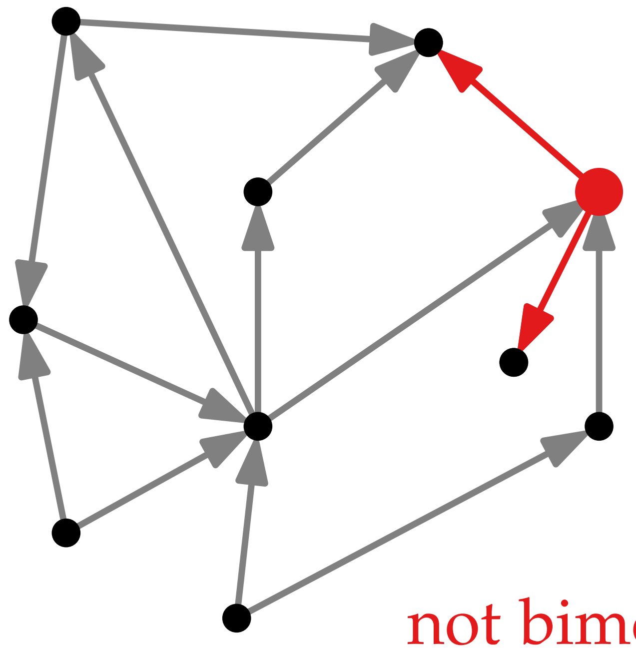
Bimodality



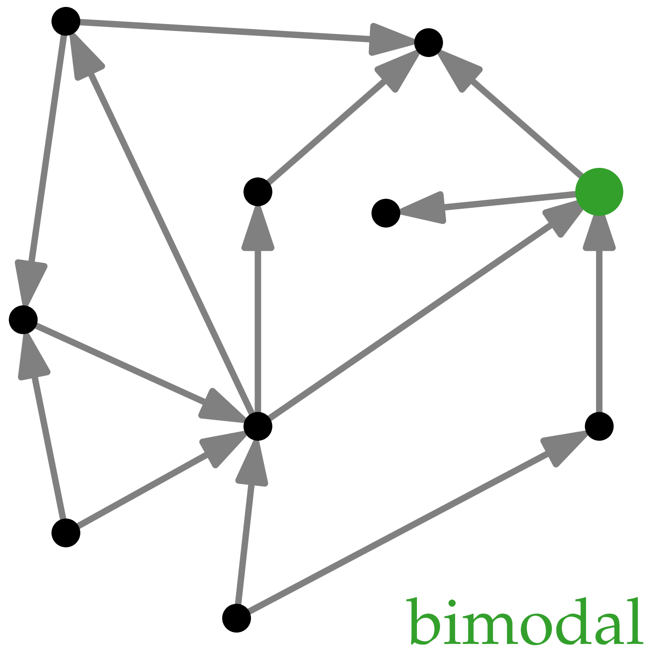
Bimodal vertex: All outgoing (incoming) edges are consecutive.

Bimodal Graph: Every vertex is bimodal.

Bimodality



not bimodal

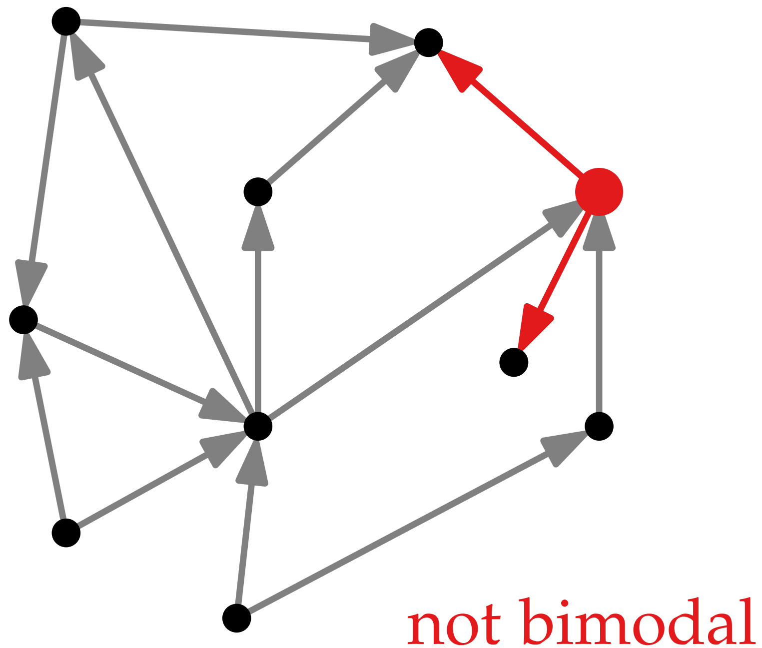


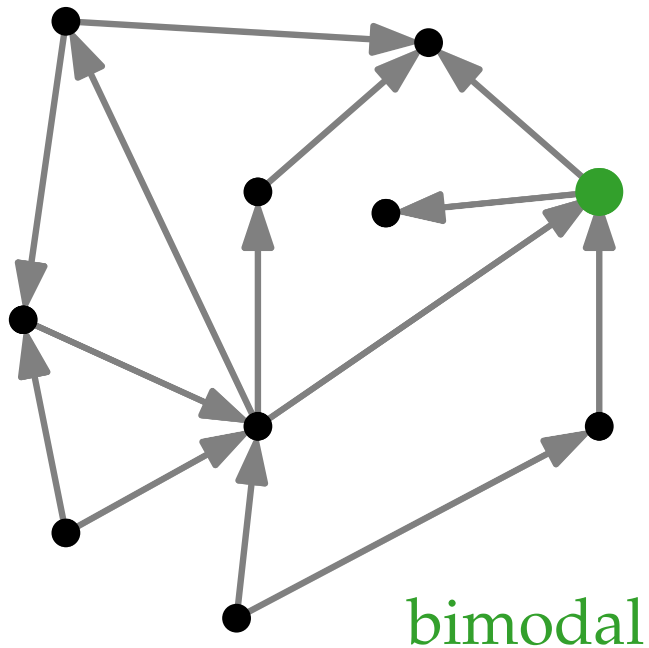
bimodal

Bimodal vertex: All outgoing (incoming) edges are consecutive.

Bimodal Graph: Every vertex is bimodal.

Bimodality





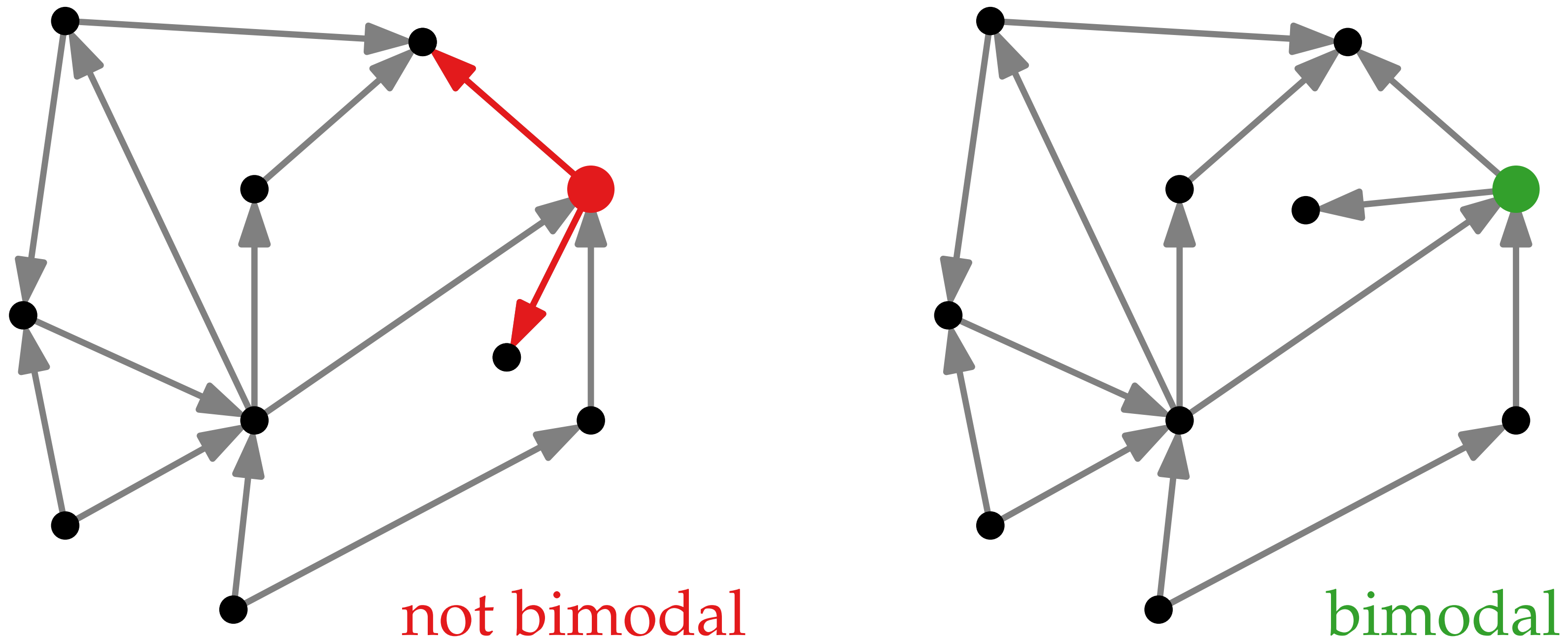
Embedding important!

$\rightarrow$ assume plane graphs

Bimodal vertex: All outgoing (incoming) edges are consecutive.

Bimodal Graph: Every vertex is bimodal.

Bimodality



Embedding important!

$\rightarrow$ assume plane graphs

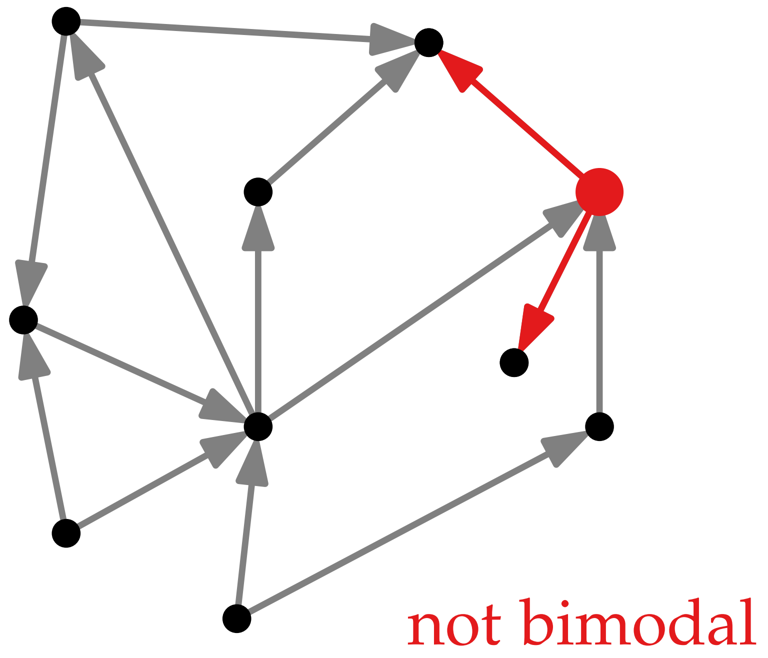
Motivation:

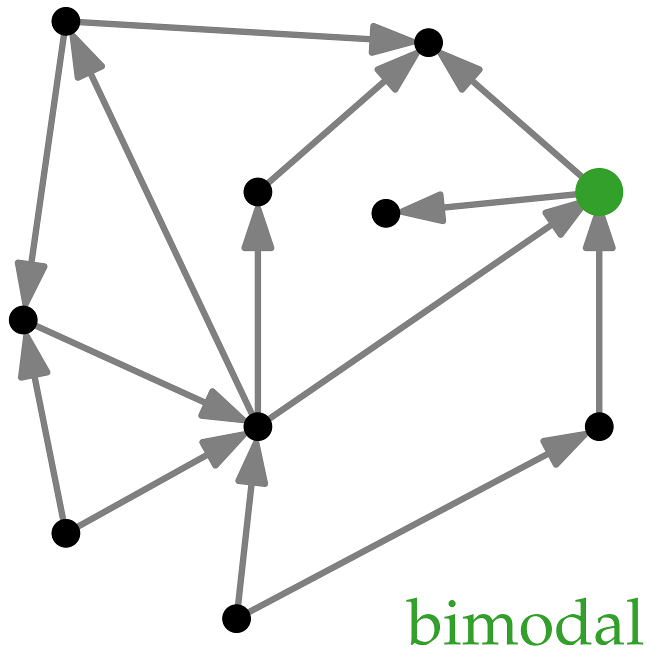
Necessary criterion for Upward Planarity, Level Planarity, …

Bimodal vertex: All outgoing (incoming) edges are consecutive.

Bimodal Graph: Every vertex is bimodal.

Bimodality





Embedding important!

→ assume plane graphs

Motivation:

Necessary criterion for Upward Planarity, Level Planarity, ...

Sufficient criterion for *L*-Drawings.

Bimodal vertex: All outgoing (incoming) edges are consecutive.

Bimodal Graph: Every vertex is bimodal.

Maximum Bimodal Subgraph Problem (MBS)

Given: Plane directed graph G

Maximum Bimodal Subgraph Problem (MBS)

Given: Plane directed graph G

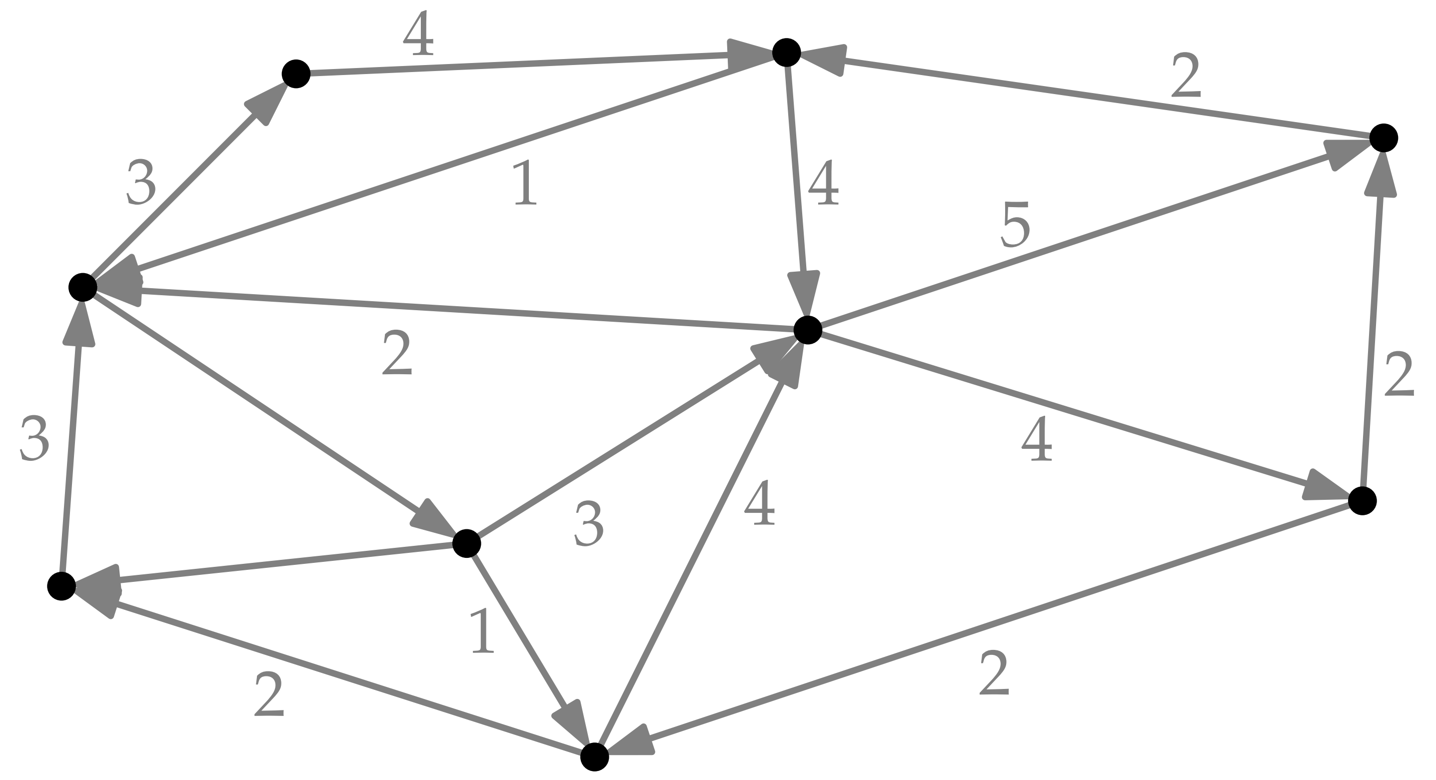
Wanted: Subgraph G' of G such that G' is bimodal and has the maximum number of edges among all bimodal subgraphs of G.

Maximum Weighted Bimodal Subgraph Problem (MWBS)

Given: Plane directed graph G with rational edge weights
Wanted: Subgraph G' of G such that G' is bimodal and has the maximum weight ~~number of edges~~ among all bimodal subgraphs of G.

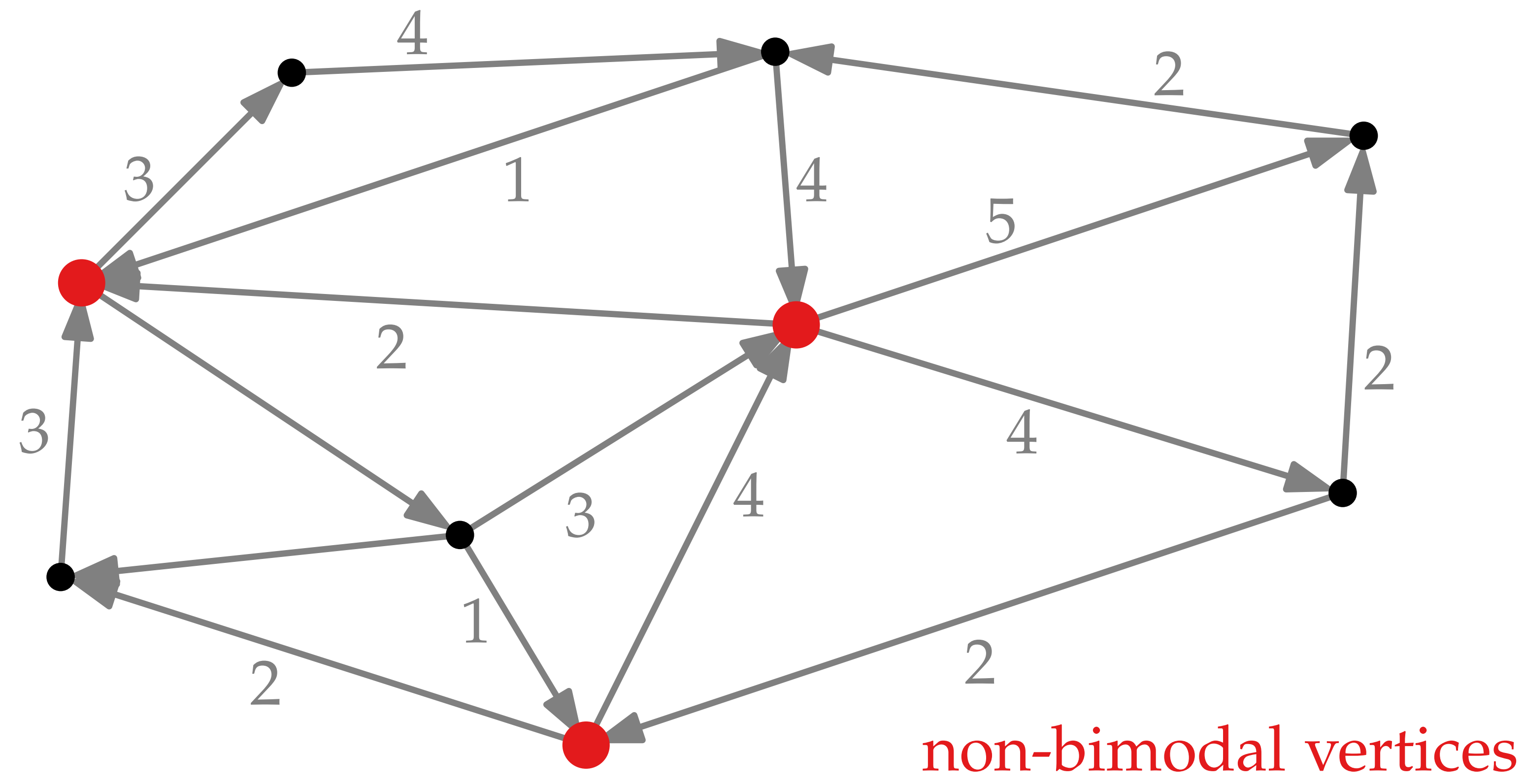
Maximum Weighted Bimodal Subgraph Problem (MWBS)

Given: Plane directed graph G with rational edge weights
Wanted: Subgraph G' of G such that G' is bimodal and has the maximum weight ~~number of edges~~ among all bimodal subgraphs of G.



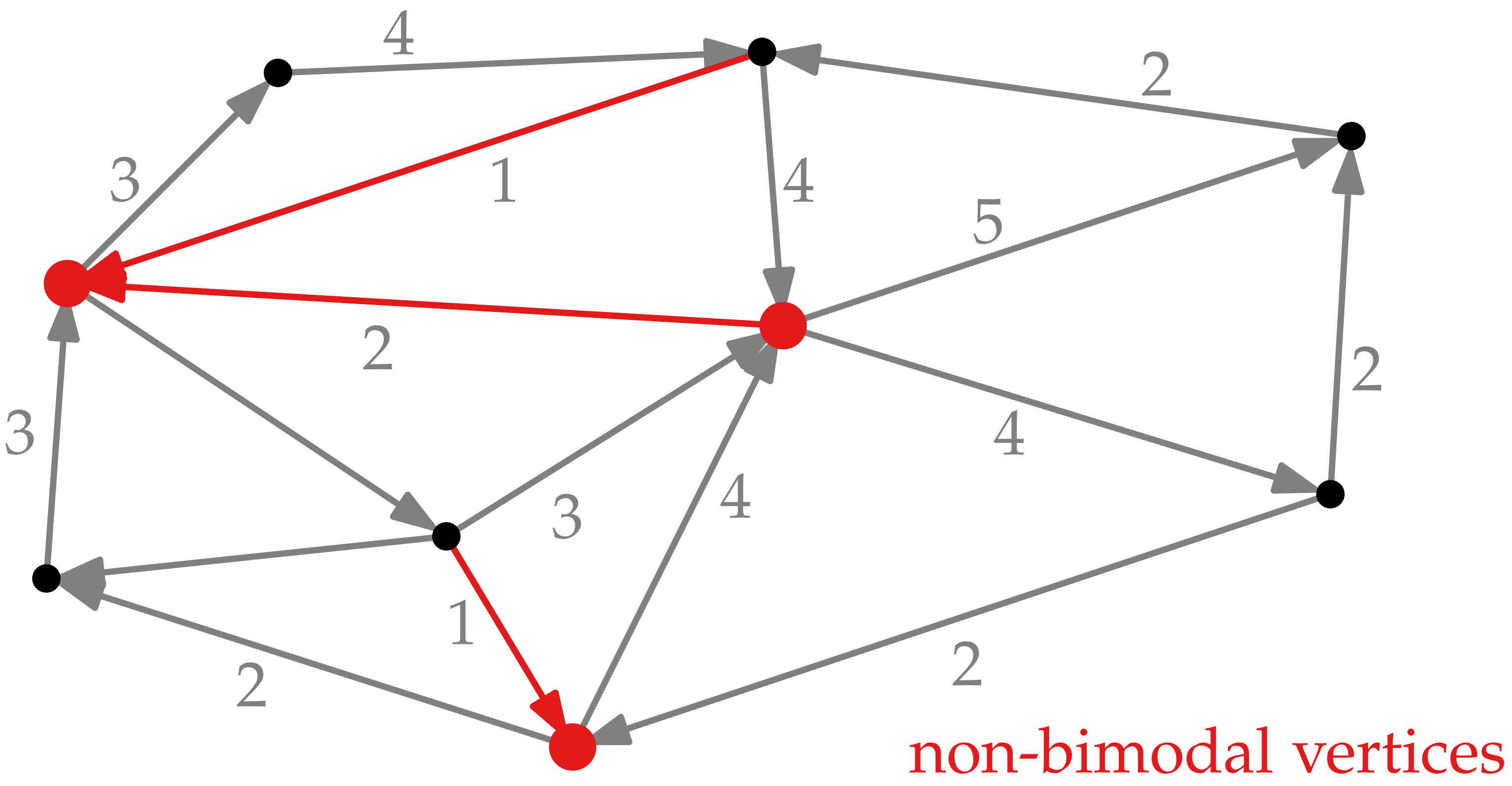
Maximum Weighted Bimodal Subgraph Problem (MWBS)

Given: Plane directed graph G with rational edge weights
Wanted: Subgraph G' of G such that G' is bimodal and has the maximum weight ~~number of edges~~ among all bimodal subgraphs of G.



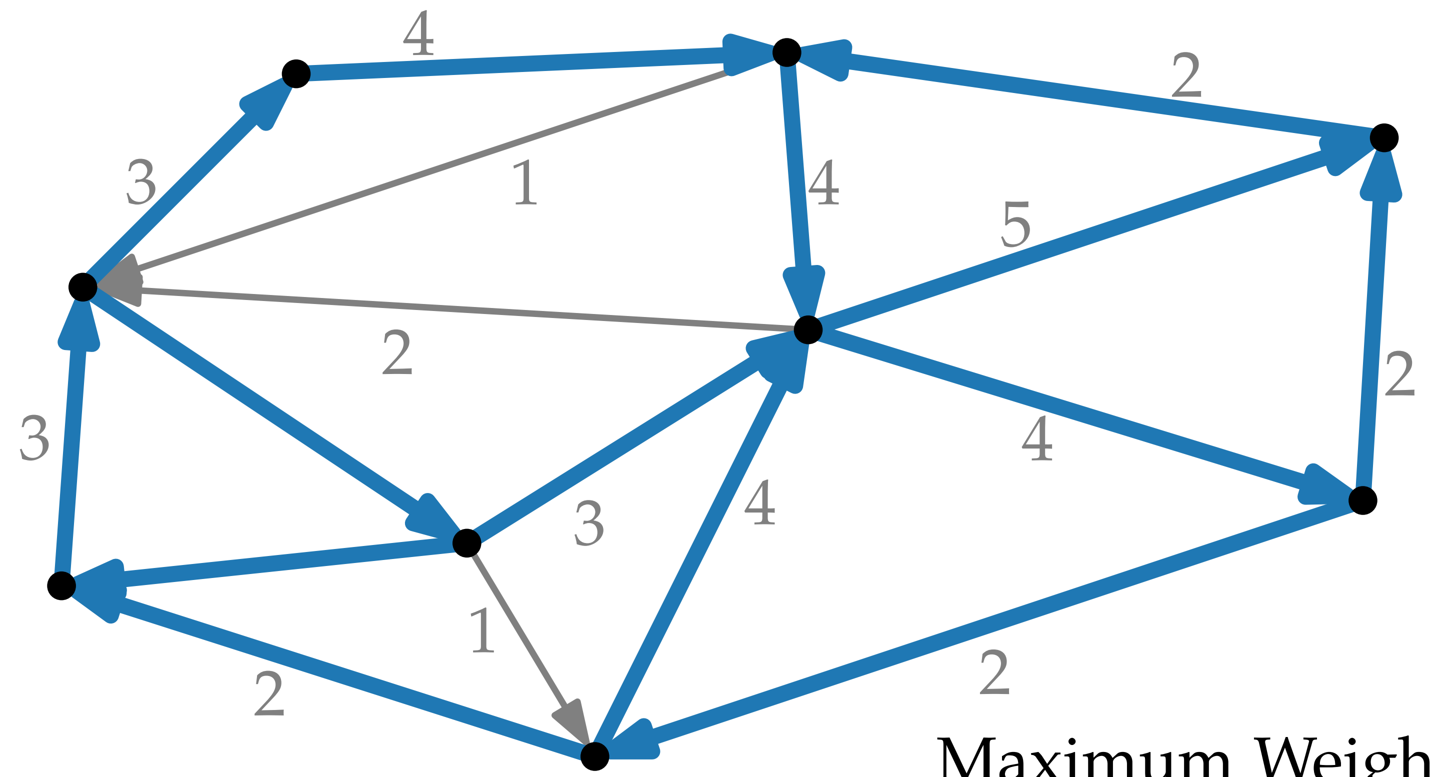
Maximum Weighted Bimodal Subgraph Problem (MWBS)

Given: Plane directed graph G with rational edge weights
Wanted: Subgraph G' of G such that G' is bimodal and has the maximum weight ~~number of edges~~ among all bimodal subgraphs of G.



Maximum Weighted Bimodal Subgraph Problem (MWBS)

Given: Plane directed graph G with rational edge weights
Wanted: Subgraph G' of G such that G' is bimodal and has the maximum weight ~~number of edges~~ among all bimodal subgraphs of G.



Maximum Weighted Bimodal Subgraph

Maximum Weighted Bimodal Subgraph Problem (MWBS)

Given: Plane directed graph G with rational edge weights
Wanted: Subgraph G' of G such that G' is bimodal and has the maximum weight ~~number of edges~~ among all bimodal subgraphs of G.

Previous Results:
[Binucci, Didimo, Giordano 2008]

- M(W)BS is NP-hard
- Heuristic (2-Approximation) for MBS
- Branch-and-Bound Algorithm for MBS

Maximum Weighted Bimodal Subgraph Problem (MWBS)

Given: Plane directed graph G with rational edge weights
Wanted: Subgraph G' of G such that G' is bimodal and has the maximum weight ~~number of edges~~ among all bimodal subgraphs of G.

Previous Results:
[Binucci, Didimo, Giordano 2008]

- M(W)BS is NP-hard
- Heuristic (2-Approximation) for MBS
- Branch-and-Bound Algorithm for MBS

Our Contribution:

Maximum Weighted Bimodal Subgraph Problem (MWBS)

Given: Plane directed graph G with rational edge weights
Wanted: Subgraph G' of G such that G' is bimodal and has the maximum weight ~~number of edges~~ among all bimodal subgraphs of G.

Previous Results:
[Binucci, Didimo, Giordano 2008]

- M(W)BS is NP-hard
- Heuristic (2-Approximation) for MBS
- Branch-and-Bound Algorithm for MBS

Our Contribution:

- Parameter: Branchwidth (/Treewidth)

Maximum Weighted Bimodal Subgraph Problem (MWBS)

Given: Plane directed graph G with rational edge weights
Wanted: Subgraph G' of G such that G' is bimodal and has the maximum weight ~~number of edges~~ among all bimodal subgraphs of G.

Previous Results:
[Binucci, Didimo, Giordano 2008]

- M(W)BS is NP-hard
- Heuristic (2-Approximation) for MBS
- Branch-and-Bound Algorithm for MBS

Our Contribution:

- Parameter: Branchwidth (/Treewidth)
  - FPT-Algorithm: running time $2^{\mathcal{O}(\text{bw}(G))} \cdot n^{\mathcal{O}(1)}$

Maximum Weighted Bimodal Subgraph Problem (MWBS)

Given: Plane directed graph G with rational edge weights
Wanted: Subgraph G' of G such that G' is bimodal and has the maximum weight ~~number of edges~~ among all bimodal subgraphs of G.

Previous Results:
[Binucci, Didimo, Giordano 2008]

- M(W)BS is NP-hard
- Heuristic (2-Approximation) for MBS
- Branch-and-Bound Algorithm for MBS

Our Contribution:

- Parameter: Branchwidth (/Treewidth)
  - FPT-Algorithm: running time $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$
- Parameter: Number b of non-bimodal vertices

Maximum Weighted Bimodal Subgraph Problem (MWBS)

Given: Plane directed graph G with rational edge weights
Wanted: Subgraph G' of G such that G' is bimodal and has the maximum weight ~~number of edges~~ among all bimodal subgraphs of G.

Previous Results:
[Binucci, Didimo, Giordano 2008]

- M(W)BS is NP-hard
- Heuristic (2-Approximation) for MBS
- Branch-and-Bound Algorithm for MBS

Our Contribution:

- Parameter: Branchwidth (/Treewidth)
  - FPT-Algorithm: running time $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$
- Parameter: Number b of non-bimodal vertices
  - FPT-Algorithm: running time $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$

Maximum Weighted Bimodal Subgraph Problem (MWBS)

Given: Plane directed graph G with rational edge weights
Wanted: Subgraph G' of G such that G' is bimodal and has the maximum weight ~~number of edges~~ among all bimodal subgraphs of G.

Previous Results:
[Binucci, Didimo, Giordano 2008]

- M(W)BS is NP-hard
- Heuristic (2-Approximation) for MBS
- Branch-and-Bound Algorithm for MBS

Our Contribution:

- Parameter: Branchwidth (/Treewidth)
  - FPT-Algorithm: running time $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$
- Parameter: Number b of non-bimodal vertices
  - FPT-Algorithm: running time $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$
  - Compression to kernel of size polynomial in b

Maximum Weighted Bimodal Subgraph Problem (MWBS)

Given: Plane directed graph G with rational edge weights
Wanted: Subgraph G' of G such that G' is bimodal and has the maximum weight ~~number of edges~~ among all bimodal subgraphs of G.

Previous Results:
[Binucci, Didimo, Giordano 2008]

- M(W)BS is NP-hard
- Heuristic (2-Approximation) for MBS
- Branch-and-Bound Algorithm for MBS

Our Contribution:

- Parameter: Branchwidth (/Treewidth)
  - FPT-Algorithm: running time $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$
- Parameter: Number b of non-bimodal vertices
  - FPT-Algorithm: running time $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$
  - Compression to kernel of size polynomial in b
- EPTAS for MWBS and for the corresponding minimization variant.

Maximum Weighted Bimodal Subgraph Problem (MWBS)

Given: Plane directed graph G with rational edge weights
Wanted: Subgraph G' of G such that G' is bimodal and has the maximum weight ~~number of edges~~ among all bimodal subgraphs of G.

Previous Results:
[Binucci, Didimo, Giordano 2008]

- M(W)BS is NP-hard
- Heuristic (2-Approximation) for MBS
- Branch-and-Bound Algorithm for MBS

Our Contribution: In this talk:

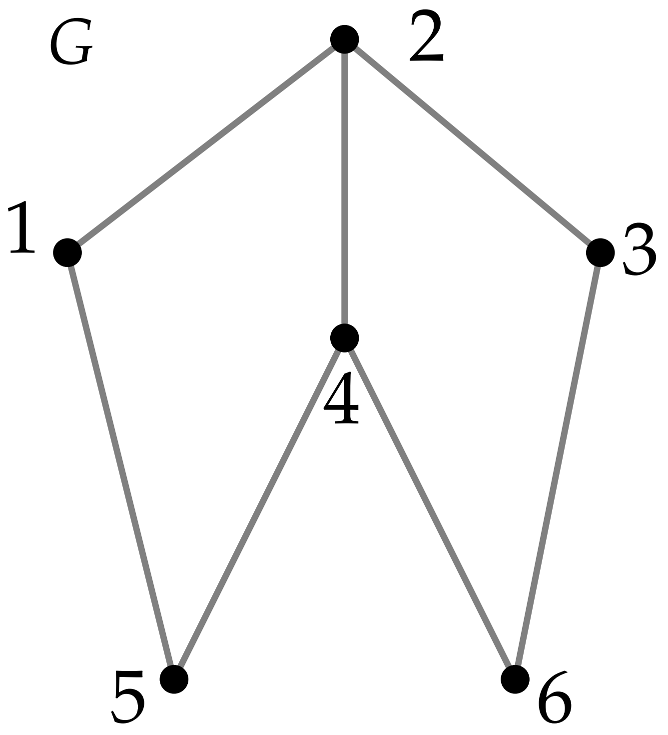
- Parameter: Branchwidth (/Treewidth)
  - FPT-Algorithm: running time $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$
- Parameter: Number b of non-bimodal vertices
  - FPT-Algorithm: running time $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$
  - Compression to kernel of size polynomial in b
- EPTAS for MWBS and for the corresponding minimization variant.

Branchwidth

Let G be a graph. A **branch decomposition** of G is an unrooted proper binary tree T whose leaves correspond bijectively to $E(G)$.

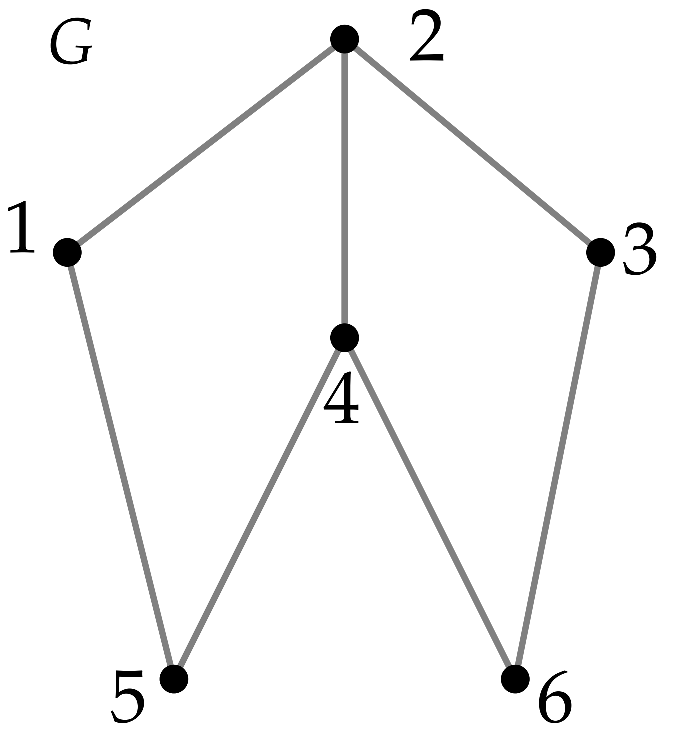
Branchwidth

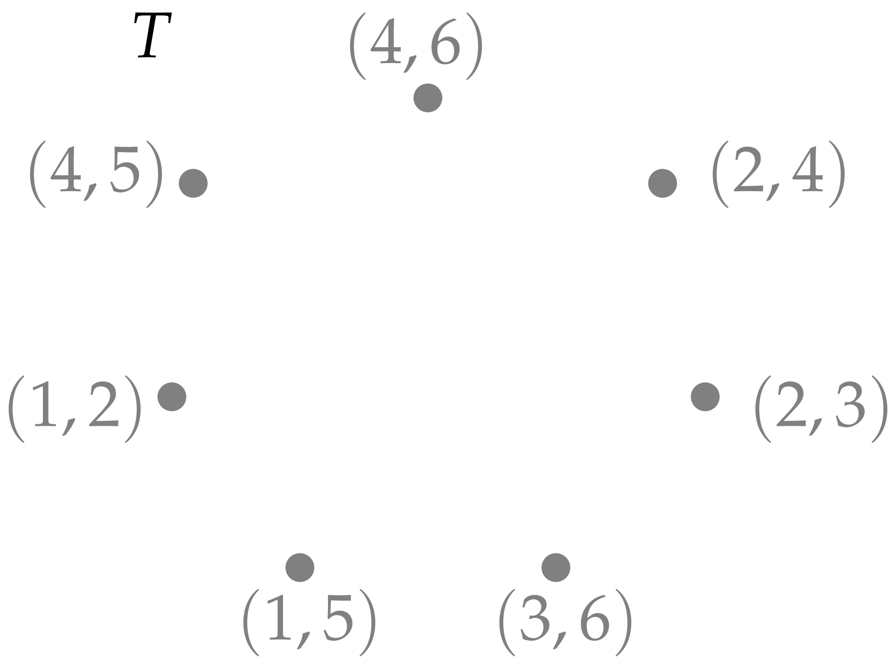
Let G be a graph. A **branch decomposition** of G is an unrooted proper binary tree T whose leaves correspond bijectively to $E(G)$.



Branchwidth

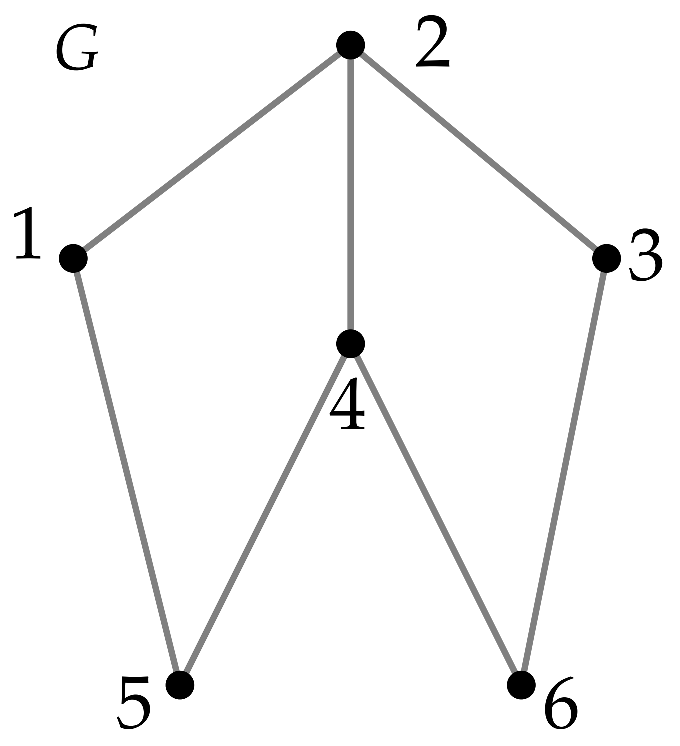
Let G be a graph. A **branch decomposition** of G is an unrooted proper binary tree T whose leaves correspond bijectively to $E(G)$.

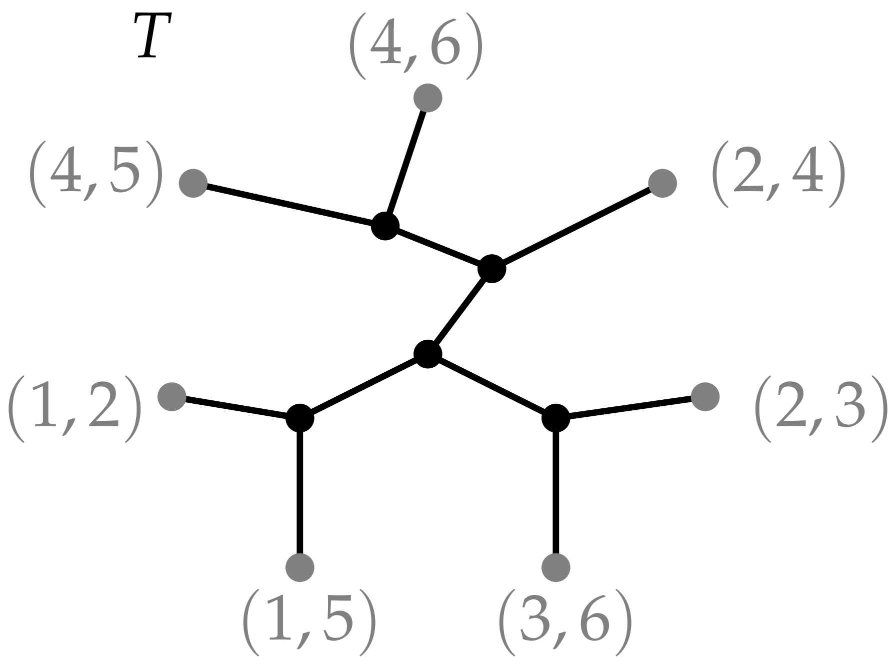




Branchwidth

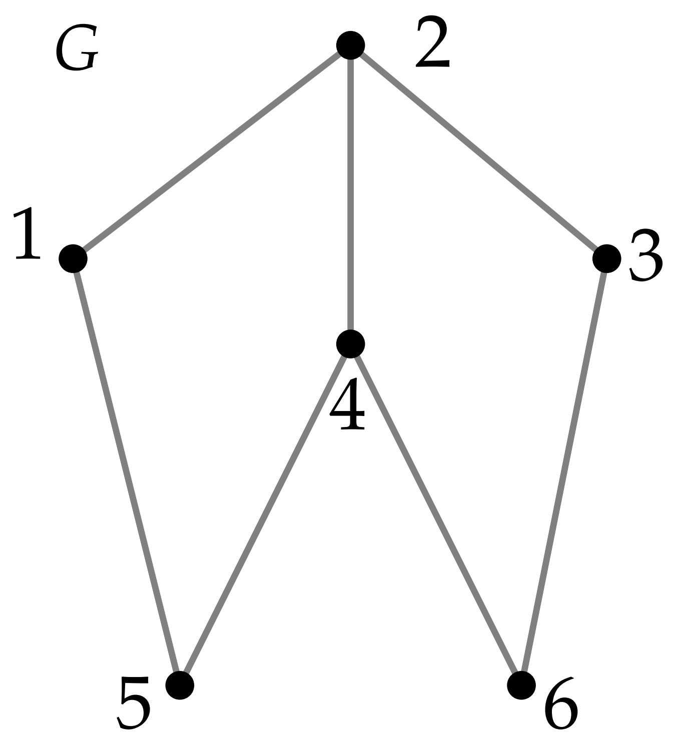
Let G be a graph. A **branch decomposition** of G is an unrooted proper binary tree T whose leaves correspond bijectively to $E(G)$.

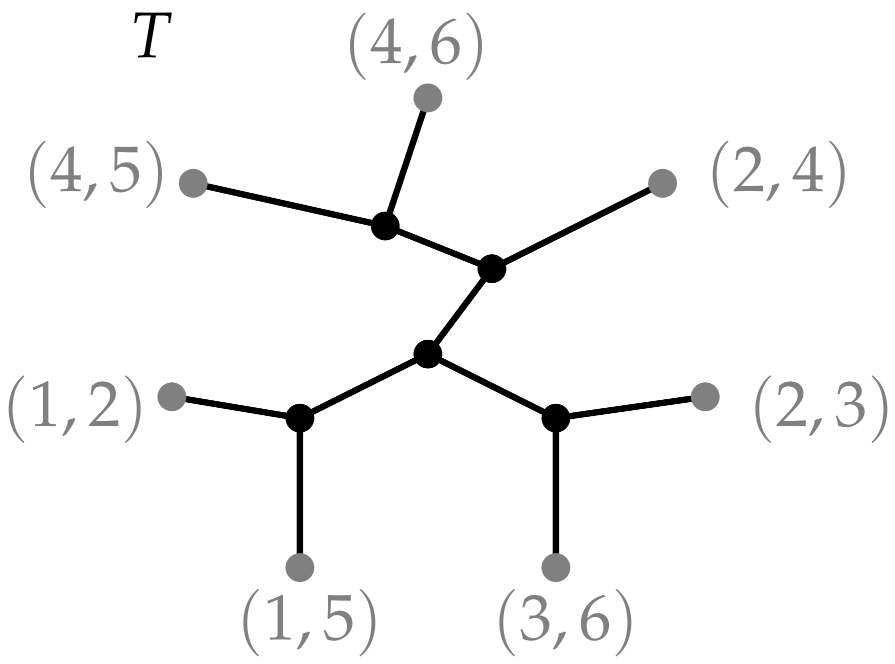




Branchwidth

Let G be a graph. A **branch decomposition** of G is an unrooted proper binary tree T whose leaves correspond bijectively to $E(G)$.

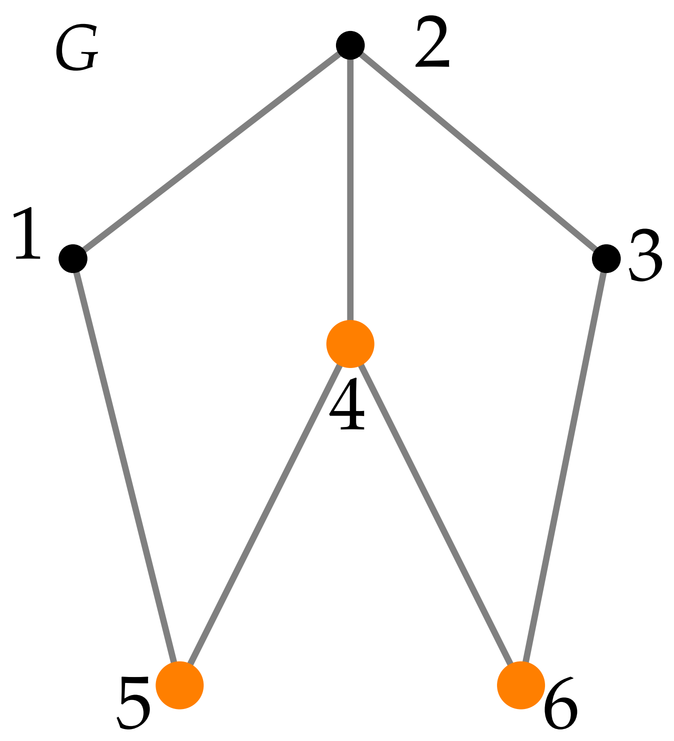


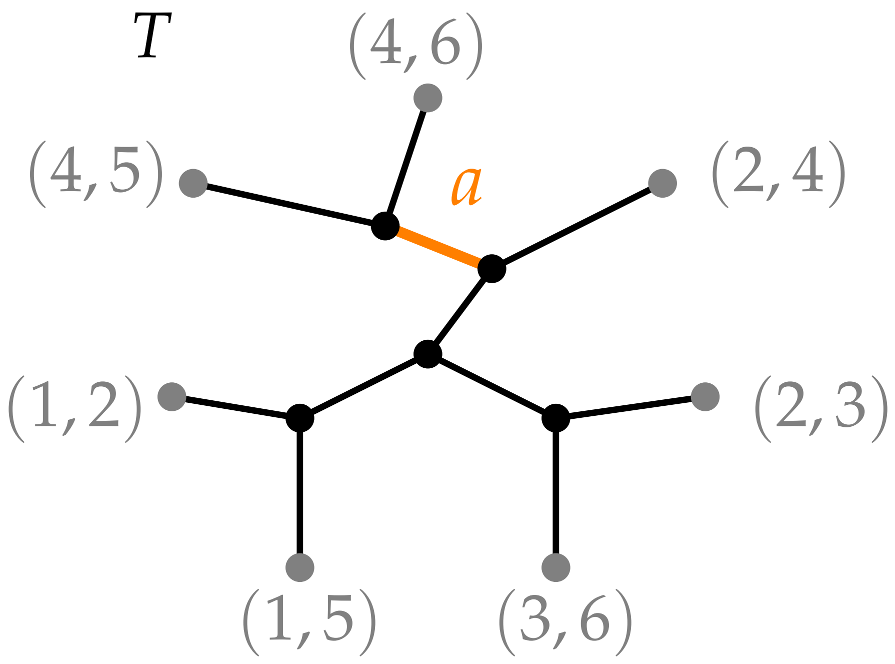


Each edge a in T splits T into two connected components and corresponds to the set of vertices in G that are adjacent to edges from both components, the **middle set** of a.

Branchwidth

Let G be a graph. A **branch decomposition** of G is an unrooted proper binary tree T whose leaves correspond bijectively to $E(G)$.

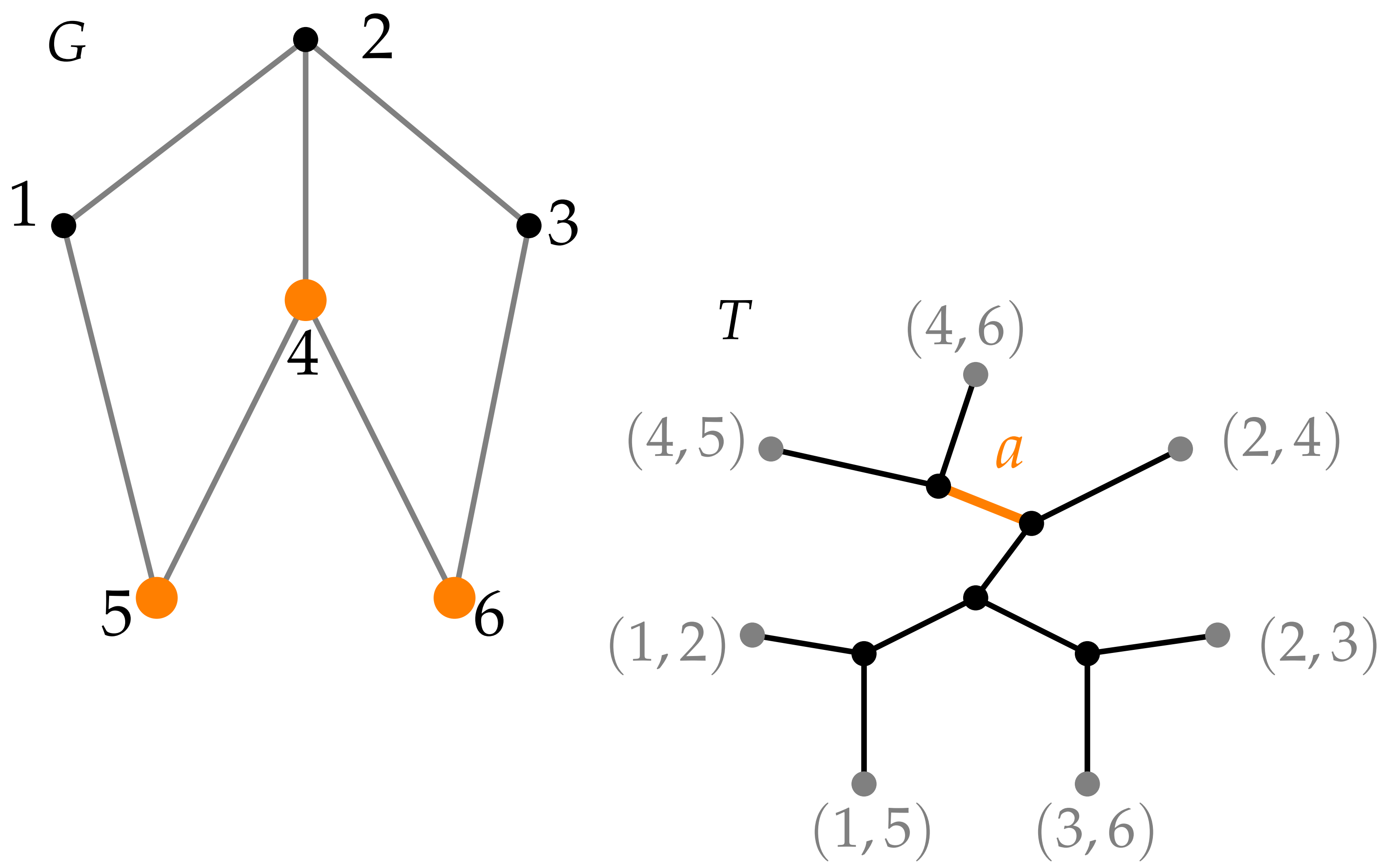




Each edge a in T splits T into two connected components and corresponds to the set of vertices in G that are adjacent to edges from both components, the **middle set** of a.

Branchwidth

Let G be a graph. A **branch decomposition** of G is an unrooted proper binary tree T whose leaves correspond bijectively to $E(G)$.

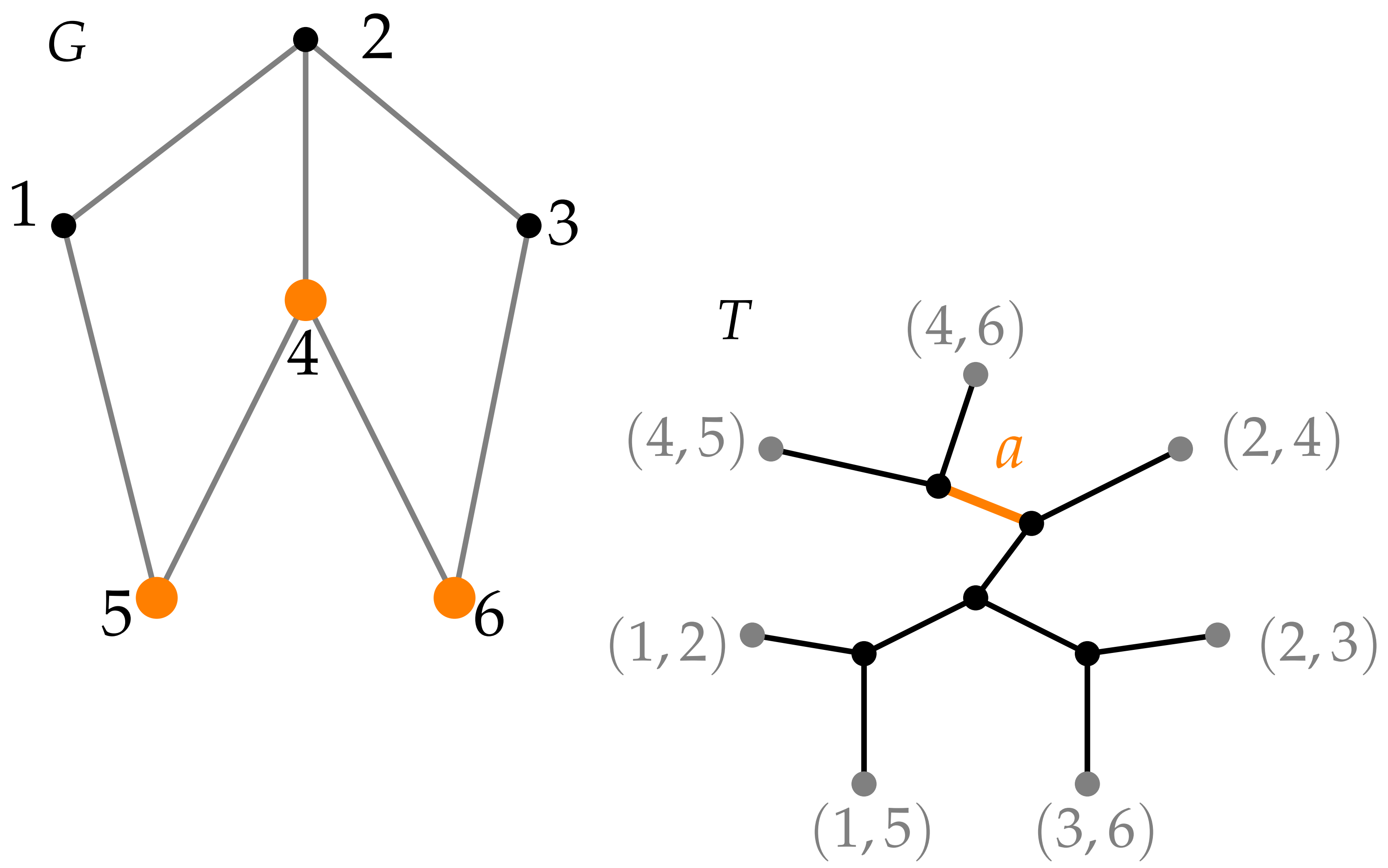


Each edge a in T splits T into two connected components and corresponds to the set of vertices in G that are adjacent to edges from both components, the **middle set** of a.

The **width** of T is the maximum size of a middle set of T.

Branchwidth

Let G be a graph. A **branch decomposition** of G is an unrooted proper binary tree T whose leaves correspond bijectively to $E(G)$.



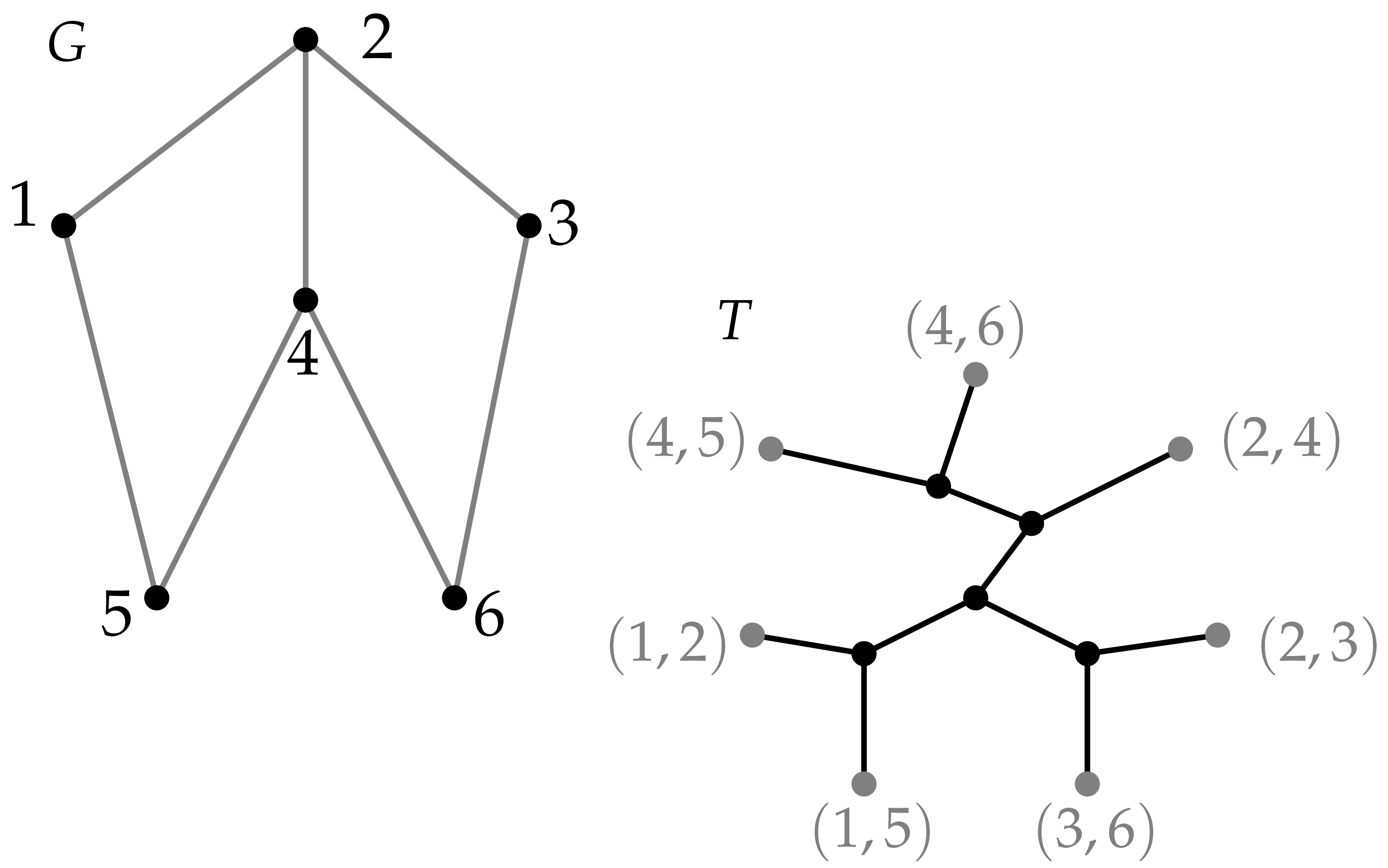
Each edge a in T splits T into two connected components and corresponds to the set of vertices in G that are adjacent to edges from both components, the **middle set** of a.

The **width** of T is the maximum size of a middle set of T.

The **branchwidth** $\mathrm{bw}(G)$ of G is the minimum width of all branch decompositions of G.

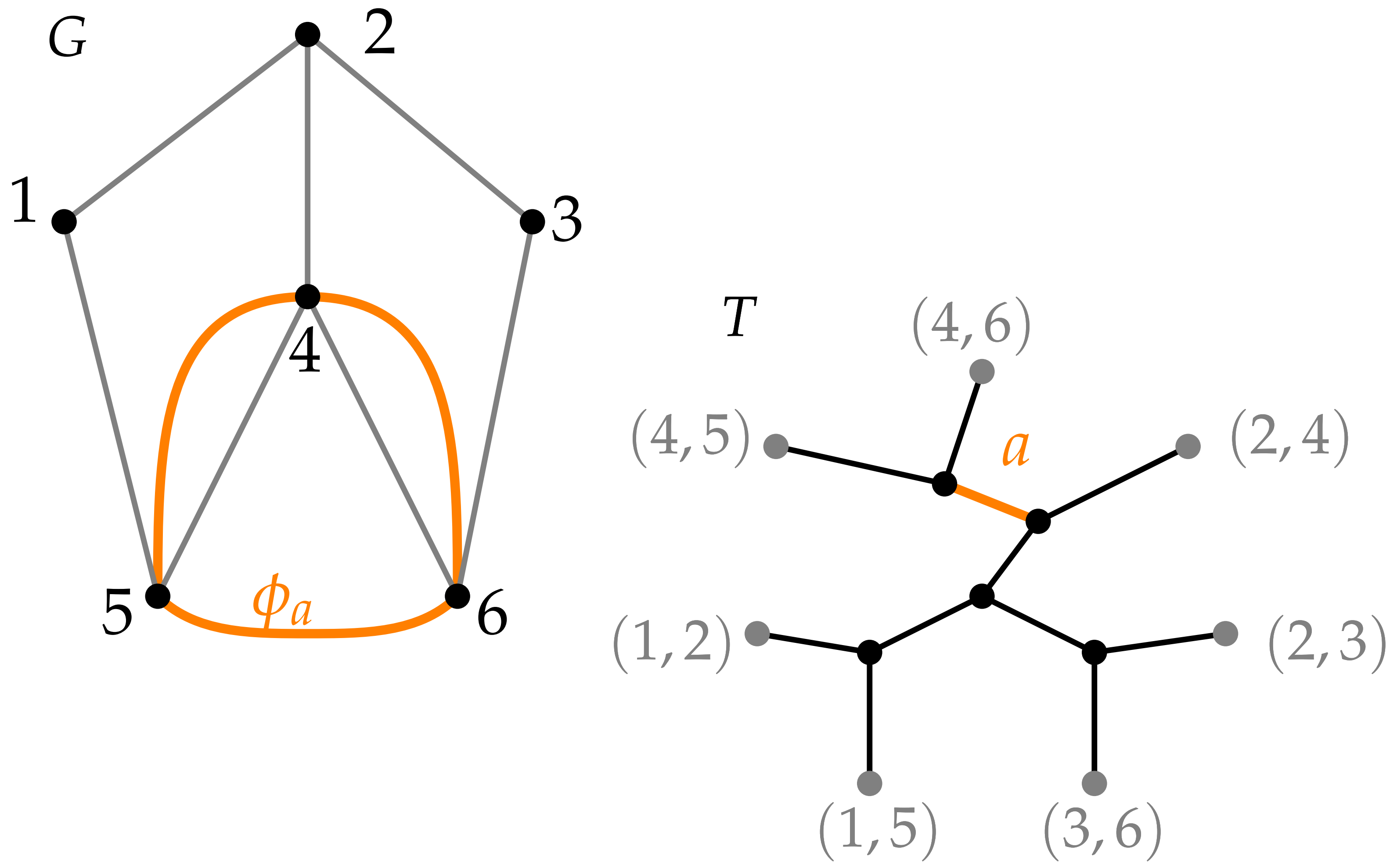
Sphere-Cut Decomposition

Let G be a connected planar graph embedded in the sphere. A **Sphere-Cut Decomposition** of G is a branch decomposition T together with simple closed curves ϕ_a for every edge $a \in T$, such that ϕ_a:



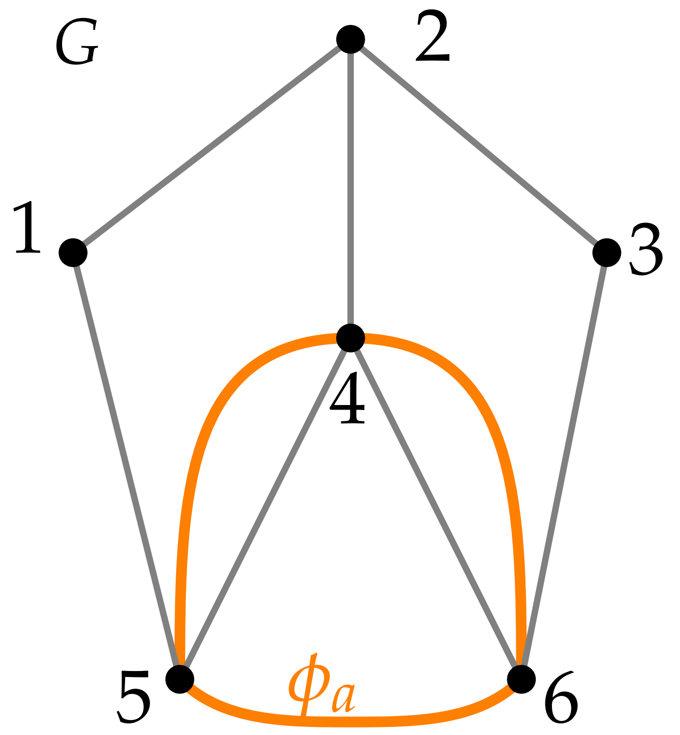
Sphere-Cut Decomposition

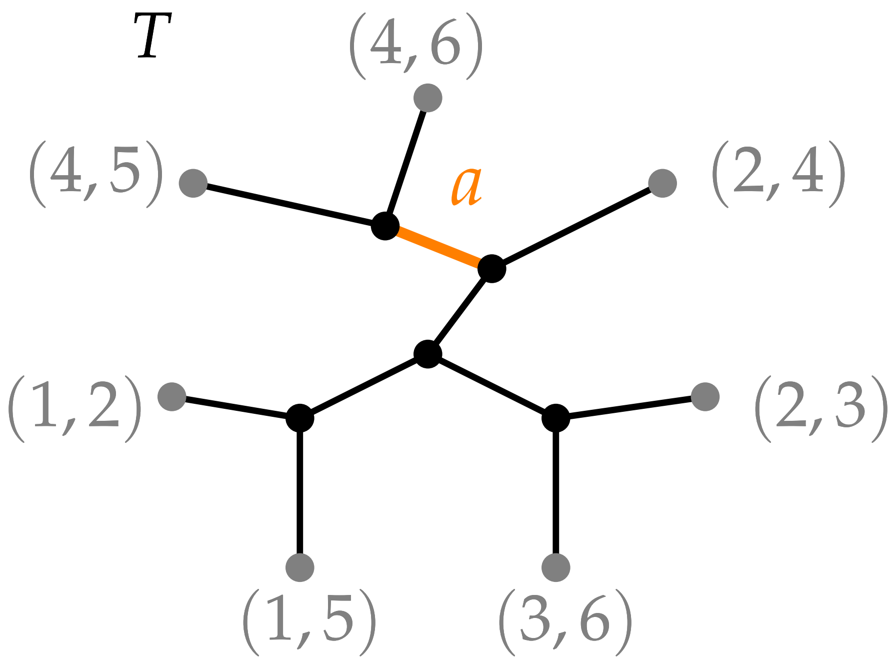
Let G be a connected planar graph embedded in the sphere. A **Sphere-Cut Decomposition** of G is a branch decomposition T together with simple closed curves ϕ_a for every edge $a \in T$, such that ϕ_a:



Sphere-Cut Decomposition

Let G be a connected planar graph embedded in the sphere. A **Sphere-Cut Decomposition** of G is a branch decomposition T together with simple closed curves ϕ_a for every edge $a \in T$, such that ϕ_a:

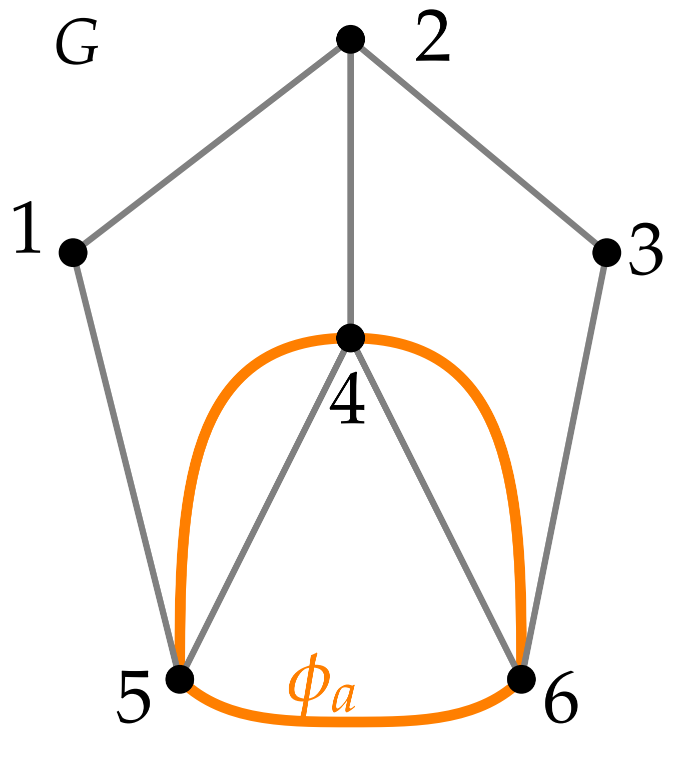


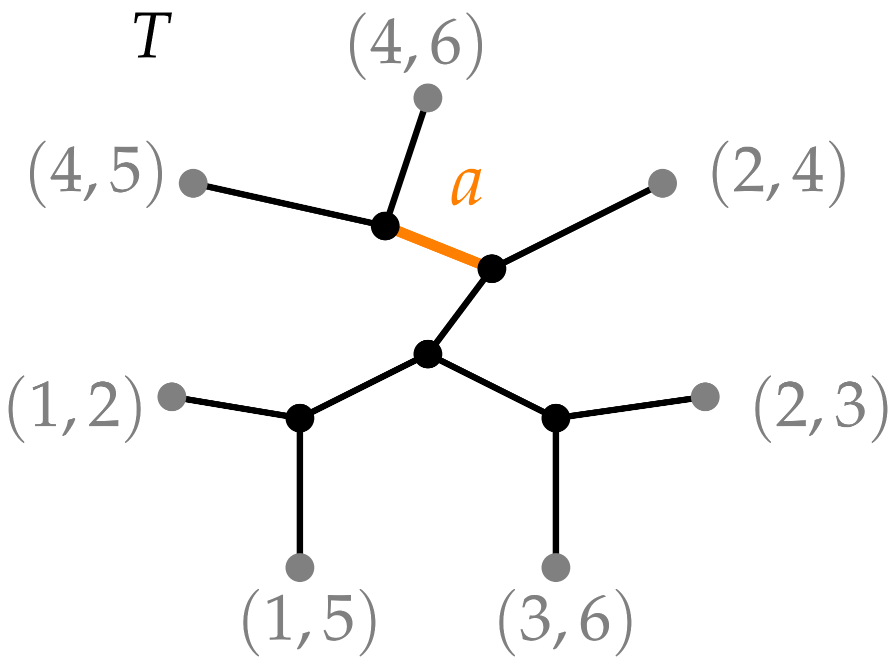


- intersects G only at the middle set of a

Sphere-Cut Decomposition

Let G be a connected planar graph embedded in the sphere. A **Sphere-Cut Decomposition** of G is a branch decomposition T together with simple closed curves ϕ_a for every edge $a \in T$, such that ϕ_a:

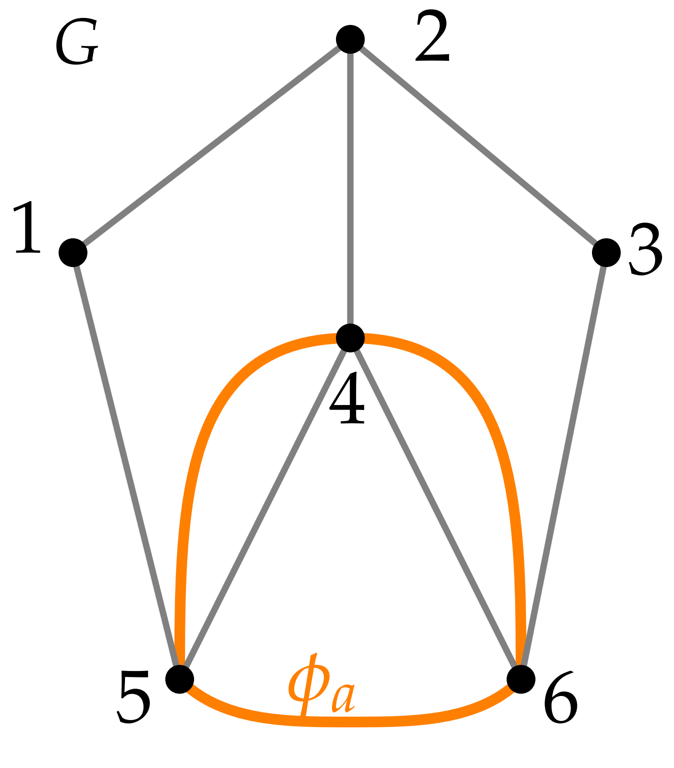


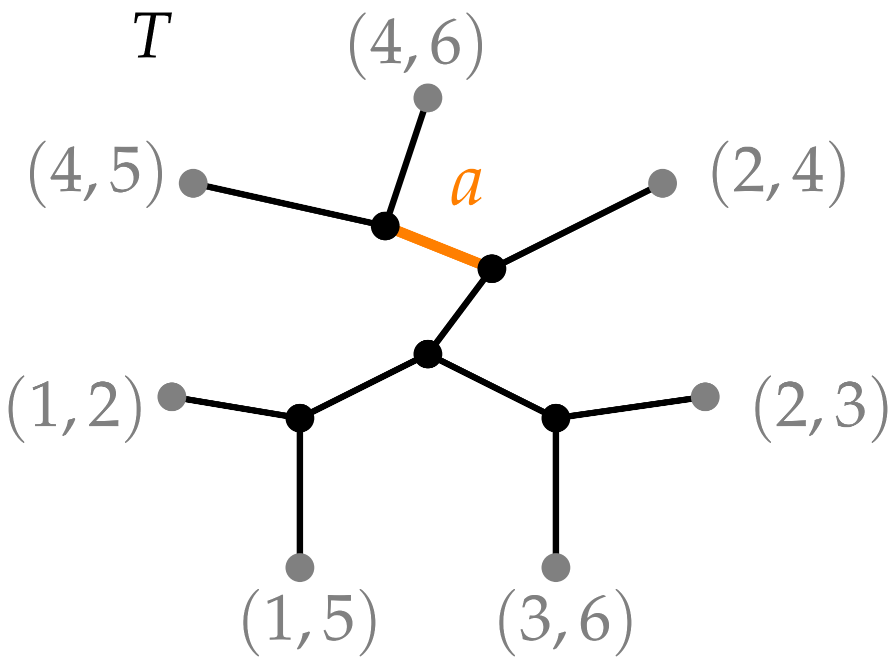


- intersects G only at the middle set of a
- separates the edges in the two components of T

Sphere-Cut Decomposition

Let G be a connected planar graph embedded in the sphere. A **Sphere-Cut Decomposition** of G is a branch decomposition T together with simple closed curves ϕ_a for every edge $a \in T$, such that ϕ_a:

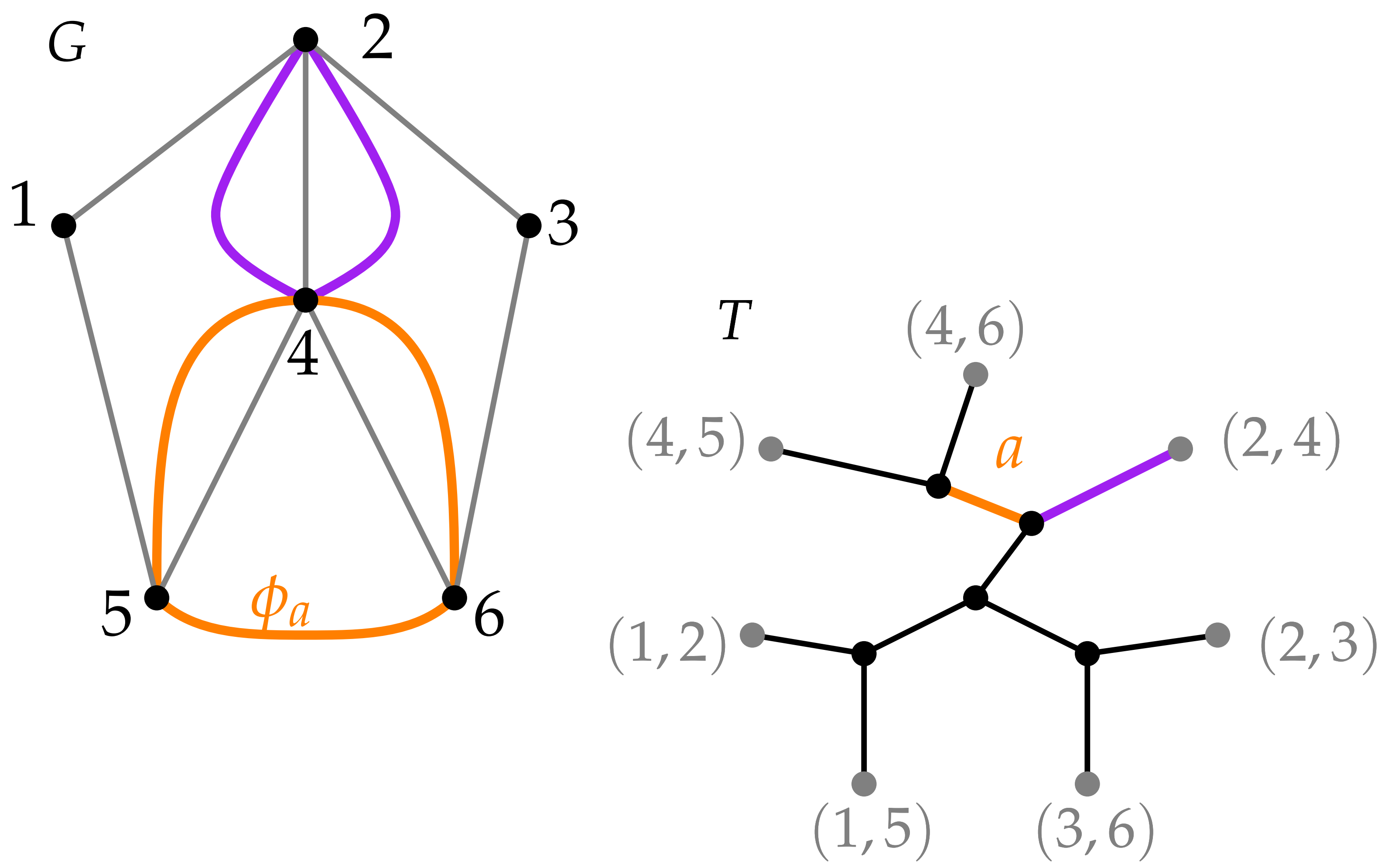




- intersects G only at the middle set of a
- separates the edges in the two components of T
- traverses each face of G at most once

Sphere-Cut Decomposition

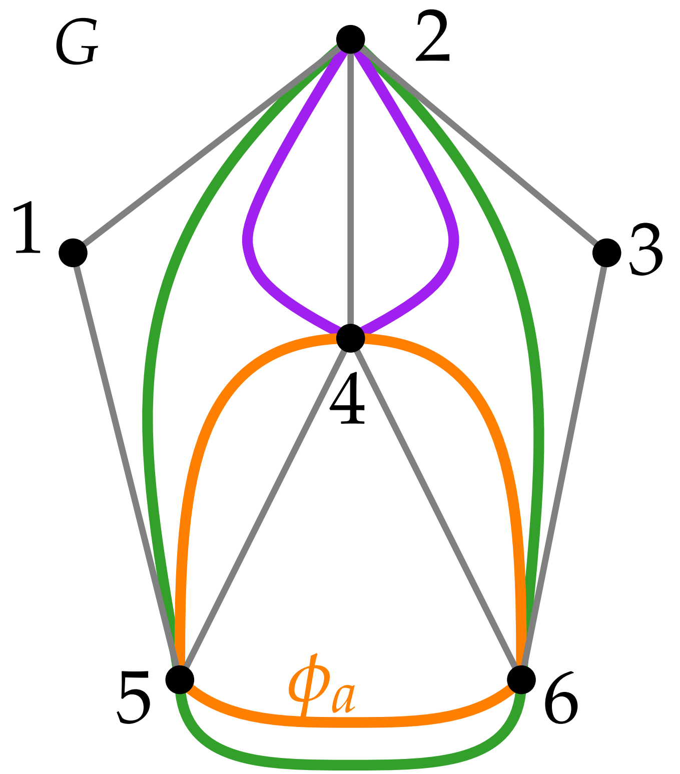
Let G be a connected planar graph embedded in the sphere. A **Sphere-Cut Decomposition** of G is a branch decomposition T together with simple closed curves ϕ_a for every edge $a \in T$, such that ϕ_a:

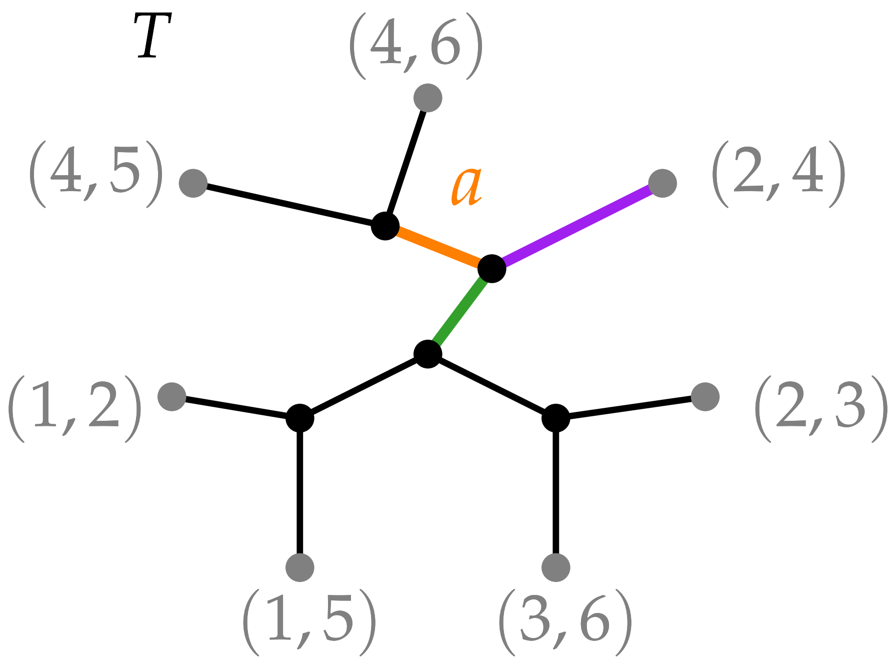


- intersects G only at the middle set of a
- separates the edges in the two components of T
- traverses each face of G at most once

Sphere-Cut Decomposition

Let G be a connected planar graph embedded in the sphere. A **Sphere-Cut Decomposition** of G is a branch decomposition T together with simple closed curves ϕ_a for every edge $a \in T$, such that ϕ_a:

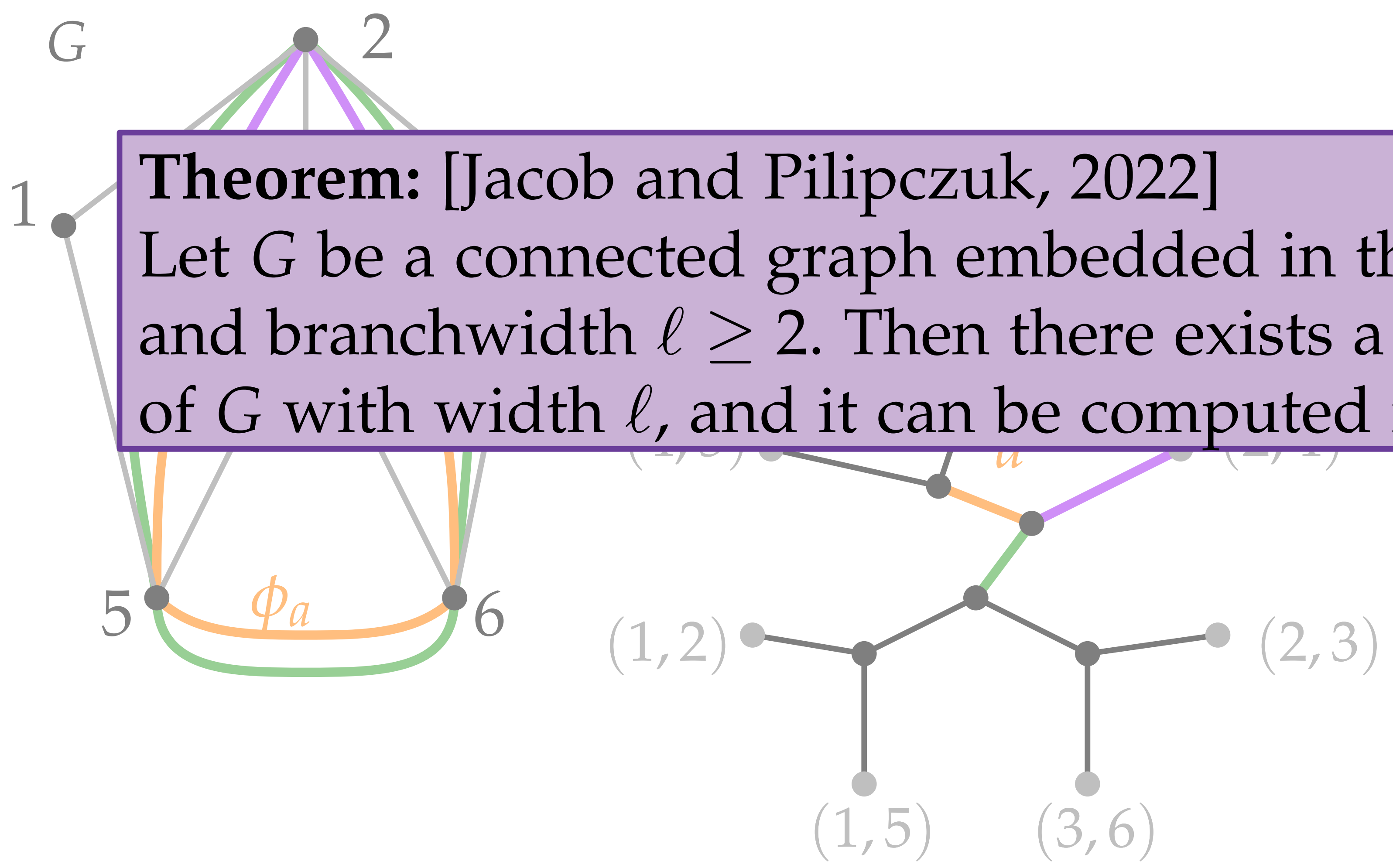




- intersects G only at the middle set of a
- separates the edges in the two components of T
- traverses each face of G at most once

Sphere-Cut Decomposition

Let G be a connected planar graph embedded in the sphere. A **Sphere-Cut Decomposition** of G is a branch decomposition T together with simple closed curves ϕ_a for every edge $a \in T$, such that ϕ_a:



Theorem: [Jacob and Pilipczuk, 2022]
Let G be a connected graph embedded in the sphere with n vertices and branchwidth $\ell \geq 2$. Then there exists a sphere-cut decomposition of G with width ℓ, and it can be computed in $\mathcal{O}(n^3)$ time.

- … the
- … the
- traverses each face of G at most once

Parametrization by Branchwidth

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Parametrization by Branchwidth: Configurations

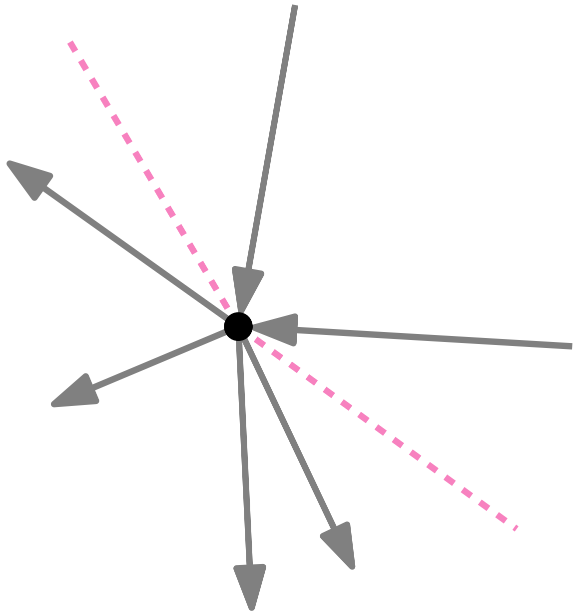
Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

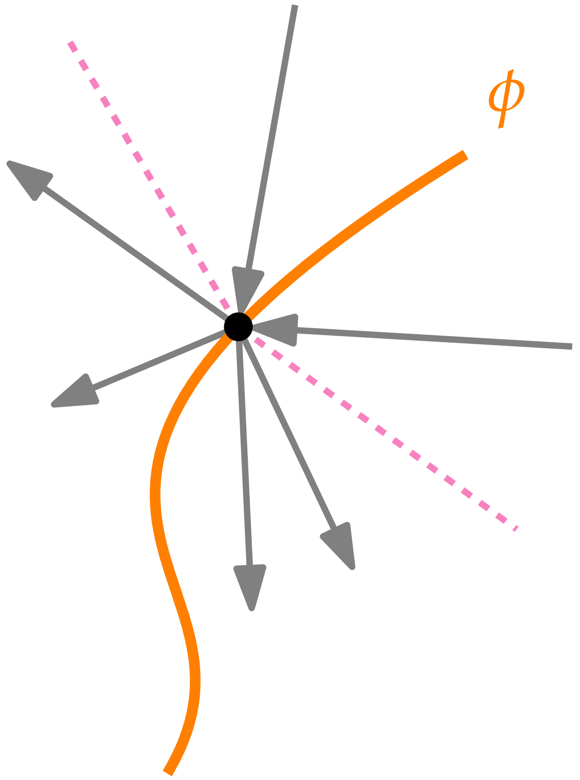


- v is bimodal $\Leftrightarrow$ at most one switch from incoming to outgoing edges and vice versa in the clockwise order of the edges incident to v.

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

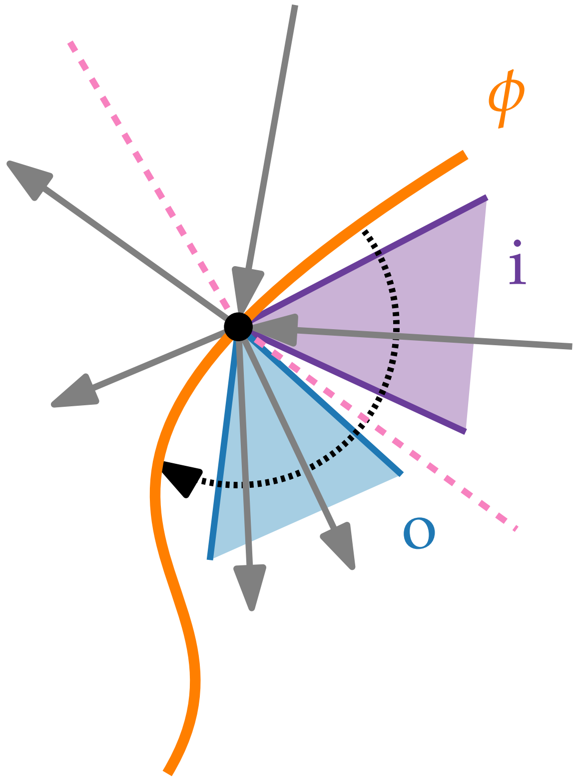


- v is bimodal $\Leftrightarrow$ at most one switch from incoming to outgoing edges and vice versa in the clockwise order of the edges incident to v.
- If v is cut by a curve ϕ: keep track on which sides of ϕ the switches are.

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\text{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

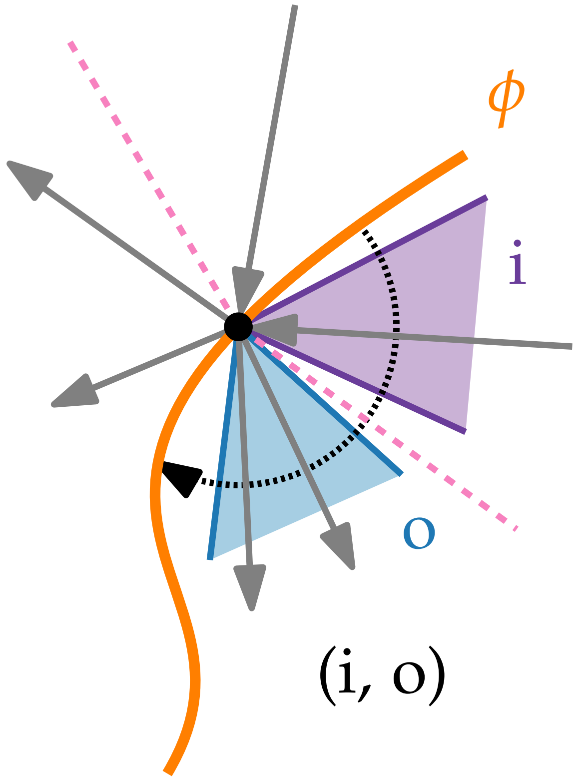


- v is bimodal $\Leftrightarrow$ at most one switch from incoming to outgoing edges and vice versa in the clockwise order of the edges incident to v.
- If v is cut by a curve ϕ: keep track on which sides of ϕ the switches are.
- Encode switches as **configurations** by the cw order of in- and outgoing edges.

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

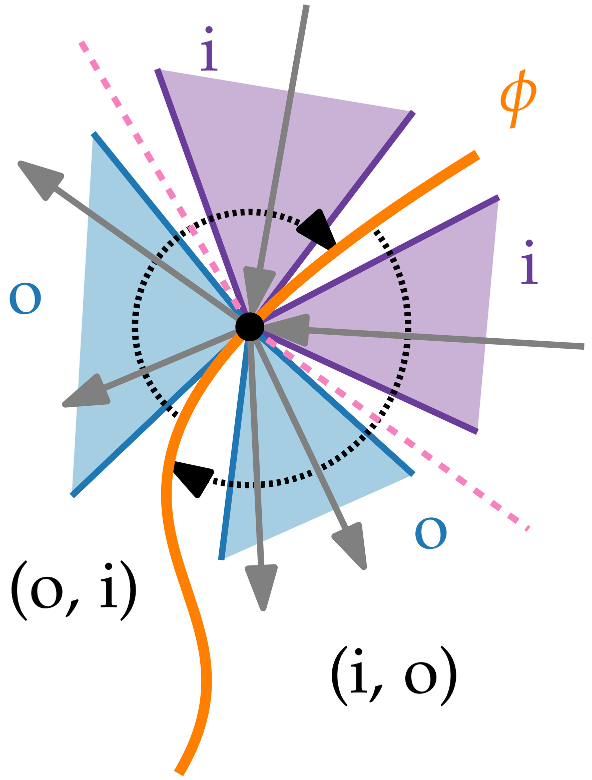


- v is bimodal $\Leftrightarrow$ at most one switch from incoming to outgoing edges and vice versa in the clockwise order of the edges incident to v.
- If v is cut by a curve ϕ: keep track on which sides of ϕ the switches are.
- Encode switches as **configurations** by the cw order of in- and outgoing edges.

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

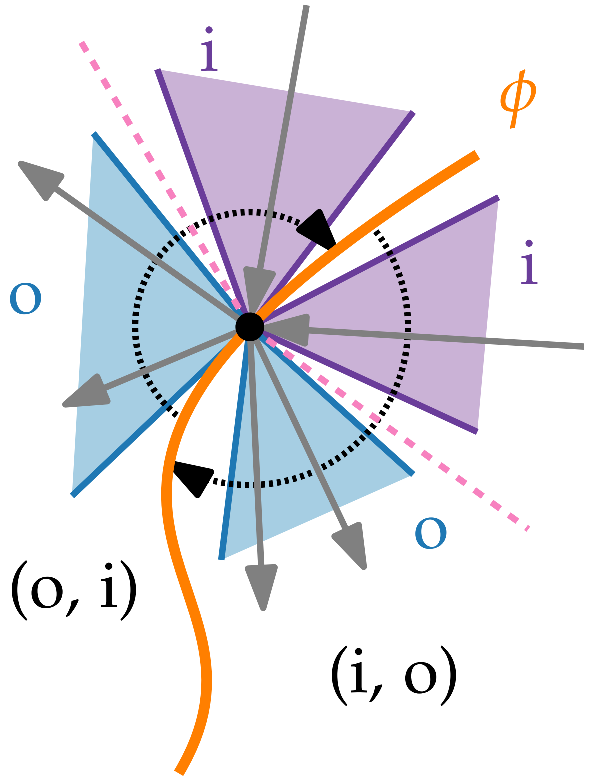


- v is bimodal $\Leftrightarrow$ at most one switch from incoming to outgoing edges and vice versa in the clockwise order of the edges incident to v.
- If v is cut by a curve ϕ: keep track on which sides of ϕ the switches are.
- Encode switches as **configurations** by the cw order of in- and outgoing edges.

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

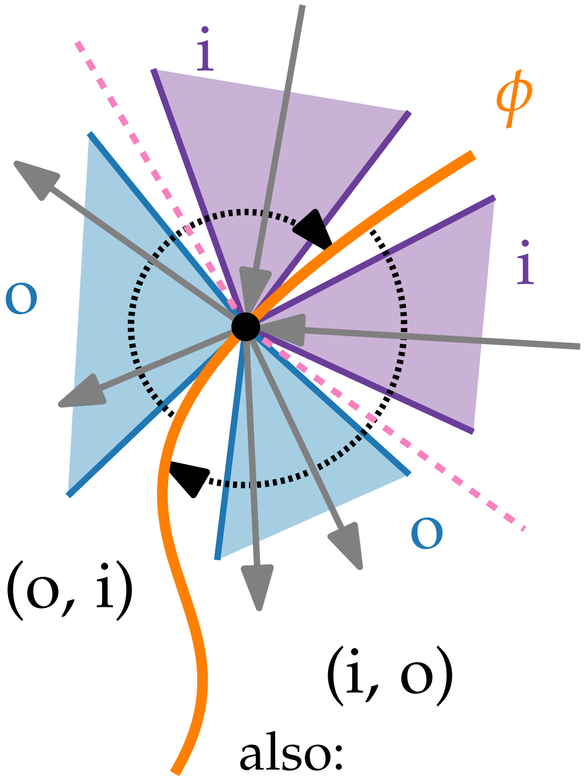


- v is bimodal $\Leftrightarrow$ at most one switch from incoming to outgoing edges and vice versa in the clockwise order of the edges incident to v.
- If v is cut by a curve ϕ: keep track on which sides of ϕ the switches are.
- Encode switches as **configurations** by the cw order of in- and outgoing edges.
- If v is bimodal, there are 6 possible configurations: (o), (i), (o, i), (i, o), (o, i, o), (i, o, i)

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\text{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

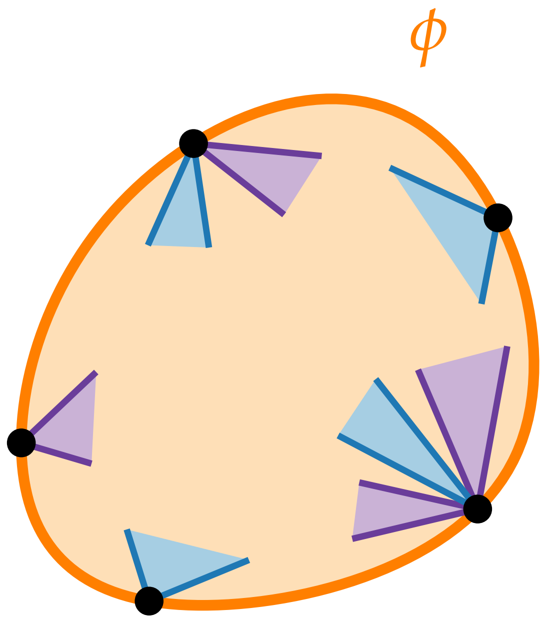


- v is bimodal $\Leftrightarrow$ at most one switch from incoming to outgoing edges and vice versa in the clockwise order of the edges incident to v.
- If v is cut by a curve ϕ: keep track on which sides of ϕ the switches are.
- Encode switches as **configurations** by the cw order of in- and outgoing edges.
- If v is bimodal, there are 6 possible configurations: (o), (i), (o, i), (i, o), (o, i, o), (i, o, i)
- Not unique:. E.g. (o) implies (i, o), (o, i), …

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\text{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

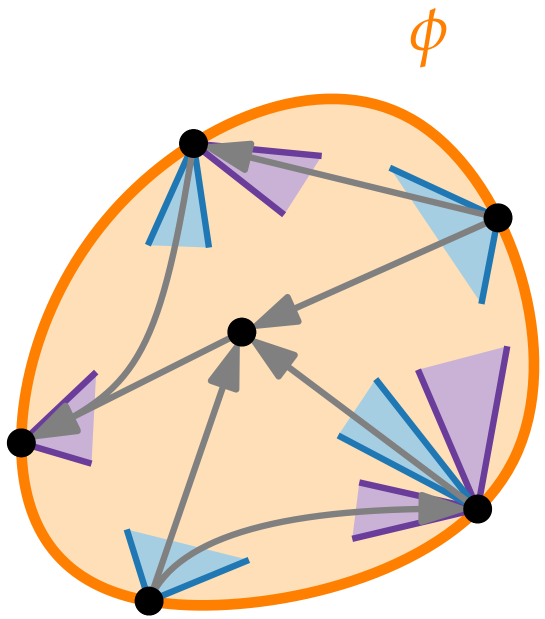


- A **configuration set** $\mathcal{X}$ for ϕ is a set with a configuration X_v for every vertex v cut by ϕ.

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

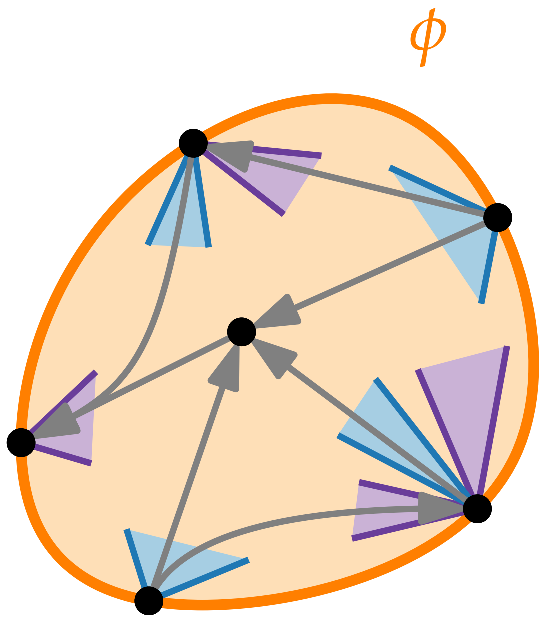


- A **configuration set** $\mathcal{X}$ for ϕ is a set with a configuration X_v for every vertex v cut by ϕ.
- G has configuration set $\mathcal{X}$, if every v cut by ϕ is of configuration X_v in ϕ.

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

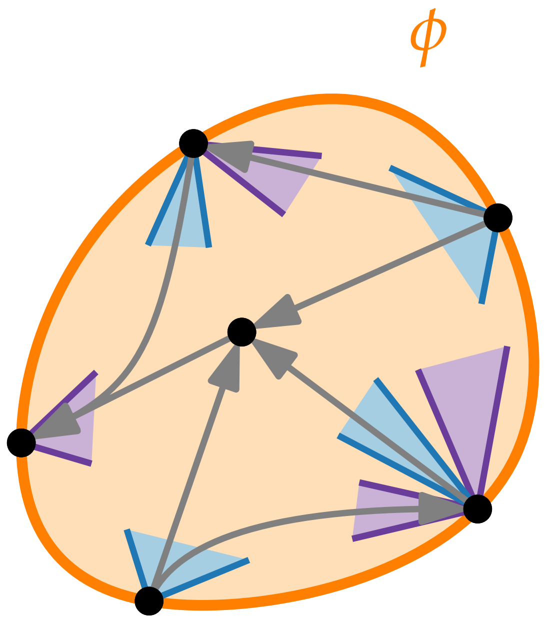


- A **configuration set** $\mathcal{X}$ for ϕ is a set with a configuration X_v for every vertex v cut by ϕ.
- G has configuration set $\mathcal{X}$, if every v cut by ϕ is of configuration X_v in ϕ.
- If ϕ corresponds to an edge of T in an sphere-cut decomposition, it cuts at most $\mathrm{bw}(G)$ vertices.

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:



- A **configuration set** $\mathcal{X}$ for ϕ is a set with a configuration X_v for every vertex v cut by ϕ.
- G has configuration set $\mathcal{X}$, if every v cut by ϕ is of configuration X_v in ϕ.
- If ϕ corresponds to an edge of T in an sphere-cut decomposition, it cuts at most $\mathrm{bw}(G)$ vertices.

  $\rightarrow$ There exist at most $6^{\mathrm{bw}(G)}$ configuration sets for ϕ.

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\text{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

- Two configurations X, X' are compatible, if their concatenation – after deleting consecutive duplicates – is a substring of (o, i, o) or (i, o, i).

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

- Two configurations X, X' are compatible, if their concatenation – after deleting consecutive duplicates – is a substring of (o, i, o) or (i, o, i).

(i, o, i), (i) $\rightarrow$ (i, o, i, i) $\rightarrow$ (i, o, i)

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

- Two configurations X, X' are compatible, if their concatenation – after deleting consecutive duplicates – is a substring of (o, i, o) or (i, o, i).

(i, o, i), (i) $\rightarrow$ (i, o, i, i) $\rightarrow$ (i, o, i) compatible

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\text{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

- Two configurations X, X' are compatible, if their concatenation – after deleting consecutive duplicates – is a substring of (o, i, o) or (i, o, i).

(i, o, i), (i)	→ (i, o, i, i)	→ (i, o, i)	compatible
(i, o), (o)	→ (i, o, o)	→ (i, o)	compatible

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\text{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

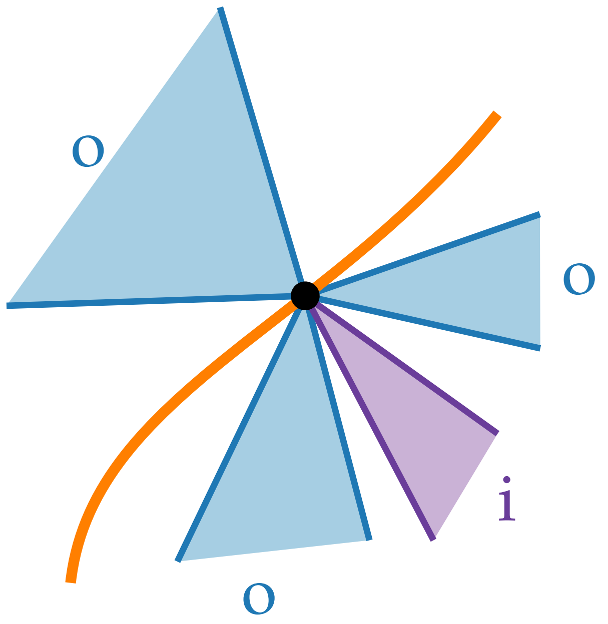
- Two configurations X, X' are compatible, if their concatenation – after deleting consecutive duplicates – is a substring of (o, i, o) or (i, o, i).

(i, o, i), (i)	$\rightarrow$	(i, o, i, i)	$\rightarrow$	(i, o, i)	compatible
(i, o), (o)	$\rightarrow$	(i, o, o)	$\rightarrow$	(i, o)	compatible
(o, i), (i, o, i)	$\rightarrow$	(o, i, i, o, i)	$\rightarrow$	(o, i, o, i)	not compatible

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\text{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:



- Two configurations X, X' are compatible, if their concatenation – after deleting consecutive duplicates – is a substring of (o, i, o) or (i, o, i).

(i, o, i), (i)	$\rightarrow$	(i, o, i, i)	$\rightarrow$	(i, o, i)	compatible
(i, o), (o)	$\rightarrow$	(i, o, o)	$\rightarrow$	(i, o)	compatible
(o, i), (i, o, i)	$\rightarrow$	(o, i, i, o, i)	$\rightarrow$	(o, i, o, i)	not compatible

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

- Two configurations X, X′ are compatible with respect to a configuration X^*, if their concatenation – after deleting consecutive duplicates – is a substring of X^*.

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

- Two configurations X, X′ are compatible with respect to a configuration X^*, if their concatenation – after deleting consecutive duplicates – is a substring of X^*.

(i, o), (o) $\rightarrow$ (i, o, o) $\rightarrow$ (i, o) compatible with (o, i, o)

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

- Two configurations X, X′ are compatible with respect to a configuration X^*, if their concatenation – after deleting consecutive duplicates – is a substring of X^*.

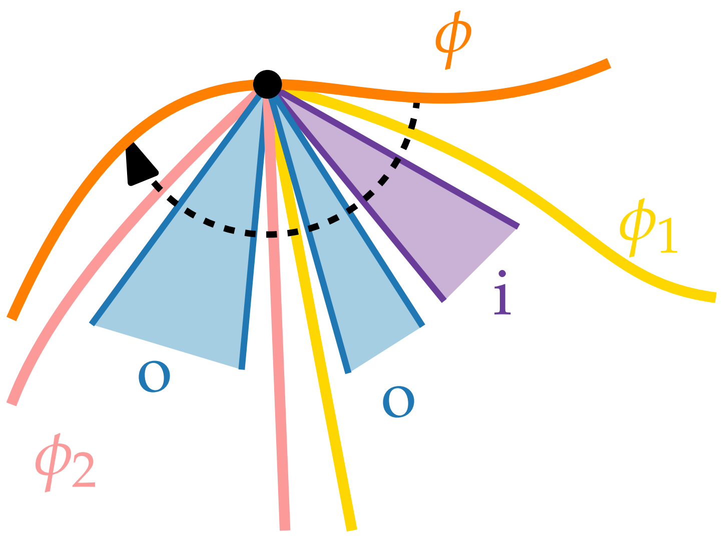
(i, o), (o) $\rightarrow$ (i, o, o) $\rightarrow$ (i, o) compatible with (o, i, o) but not compatible with (o, i)

Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

- Two configurations X, X′ are compatible with respect to a configuration X^*, if their concatenation – after deleting consecutive duplicates – is a substring of X^*.



(i, o), (o) $\rightarrow$ (i, o, o) $\rightarrow$ (i, o)

compatible with (o, i, o)
but not compatible with (o, i)

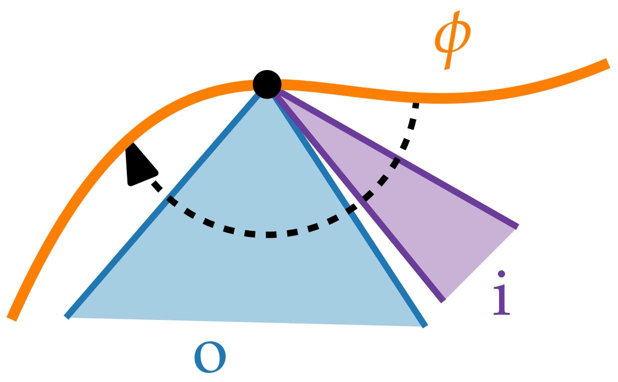
Parametrization by Branchwidth: Configurations

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Idea:

- Two configurations X, X′ are compatible with respect to a configuration X^*, if their concatenation – after deleting consecutive duplicates – is a substring of X^*.

(i, o), (o) → (i, o, o) → (i, o) compatible with (o, i, o) but not compatible with (o, i)



(i, o), but not (o, i)

Parametrization by Branchwidth: Proof Sketch

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

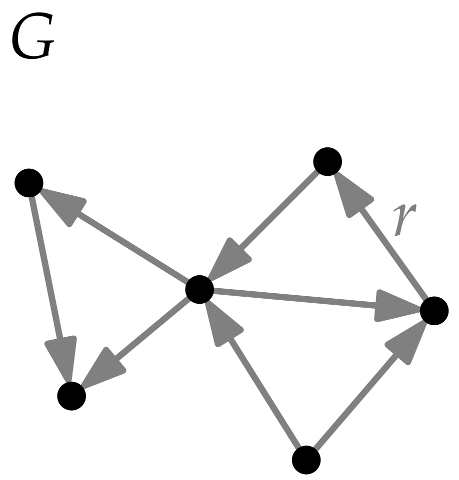
Proof sketch:

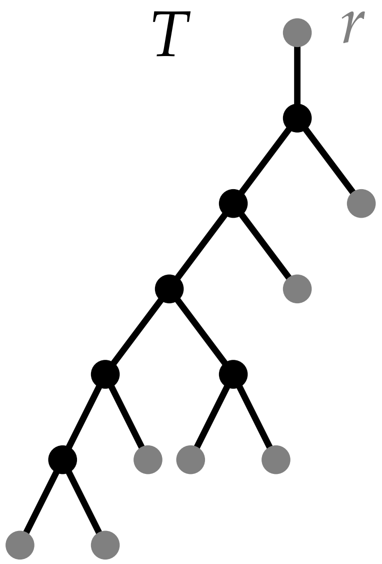
Parametrization by Branchwidth: Proof Sketch

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\text{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Proof sketch:

- Compute an optimal Sphere-Cut Decomposition T, root T arbitrarily at a leaf r.



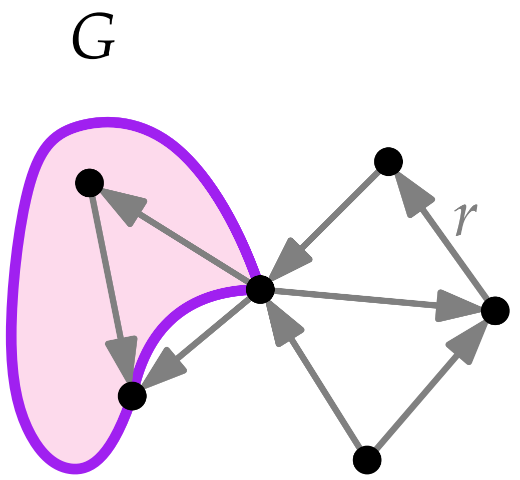


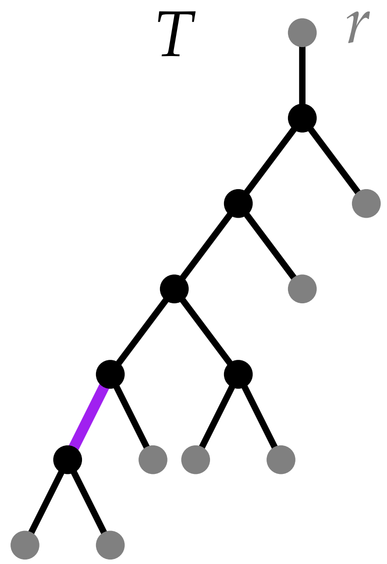
Parametrization by Branchwidth: Proof Sketch

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Proof sketch:

- Compute an optimal Sphere-Cut Decomposition T, root T arbitrarily at a leaf r.
- Let the **inside** of a curve be the side not containing r



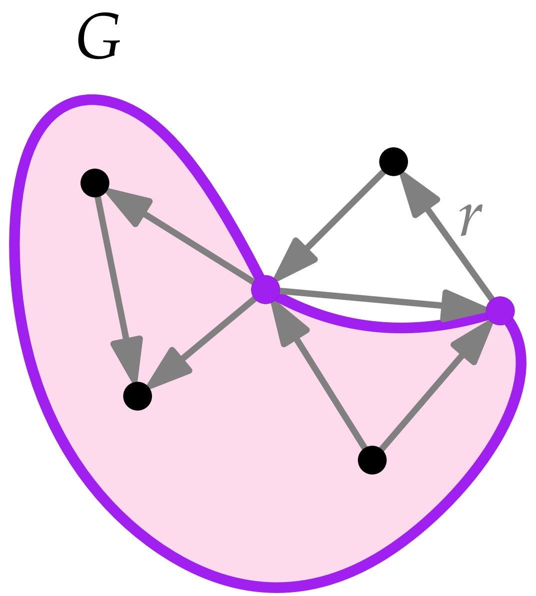


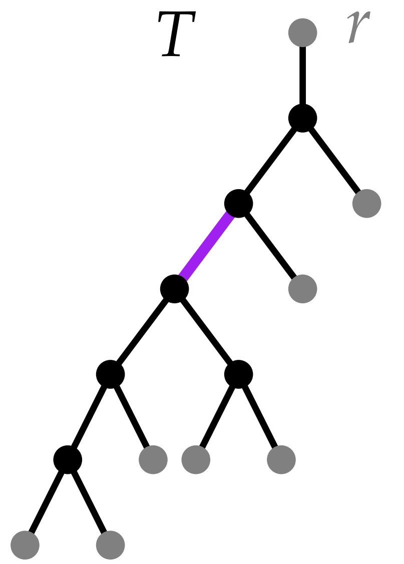
Parametrization by Branchwidth: Proof Sketch

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Proof sketch:

- Compute an optimal Sphere-Cut Decomposition T, root T arbitrarily at a leaf r.
- Let the **inside** of a curve be the side not containing r
- Compute **bottom up** for every curve ϕ_a and every configuration set $\mathcal{X}$ for ϕ_a the maximum subgraph of G that is bimodal in ϕ_a and has $\mathcal{X}$ in ϕ_a.



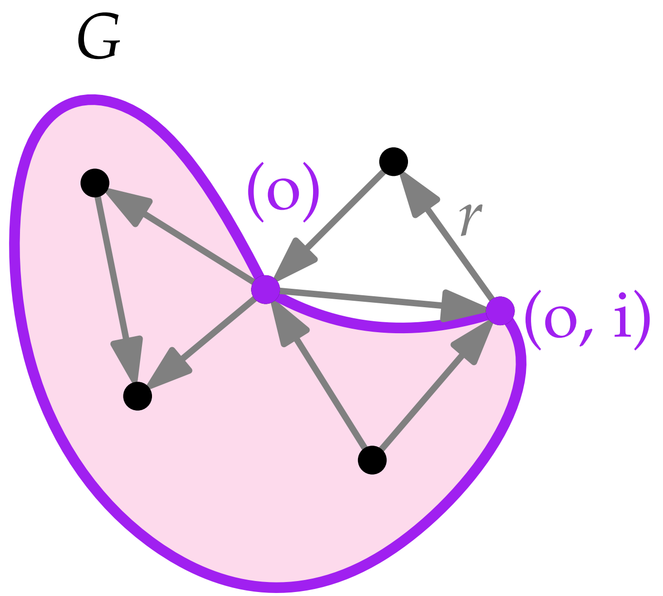


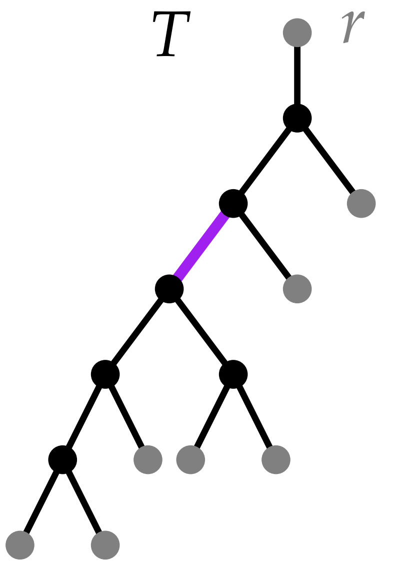
Parametrization by Branchwidth: Proof Sketch

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Proof sketch:

- Compute an optimal Sphere-Cut Decomposition T, root T arbitrarily at a leaf r.
- Let the **inside** of a curve be the side not containing r
- Compute **bottom up** for every curve ϕ_a and every configuration set $\mathcal{X}$ for ϕ_a the maximum subgraph of G that is bimodal in ϕ_a and has $\mathcal{X}$ in ϕ_a.



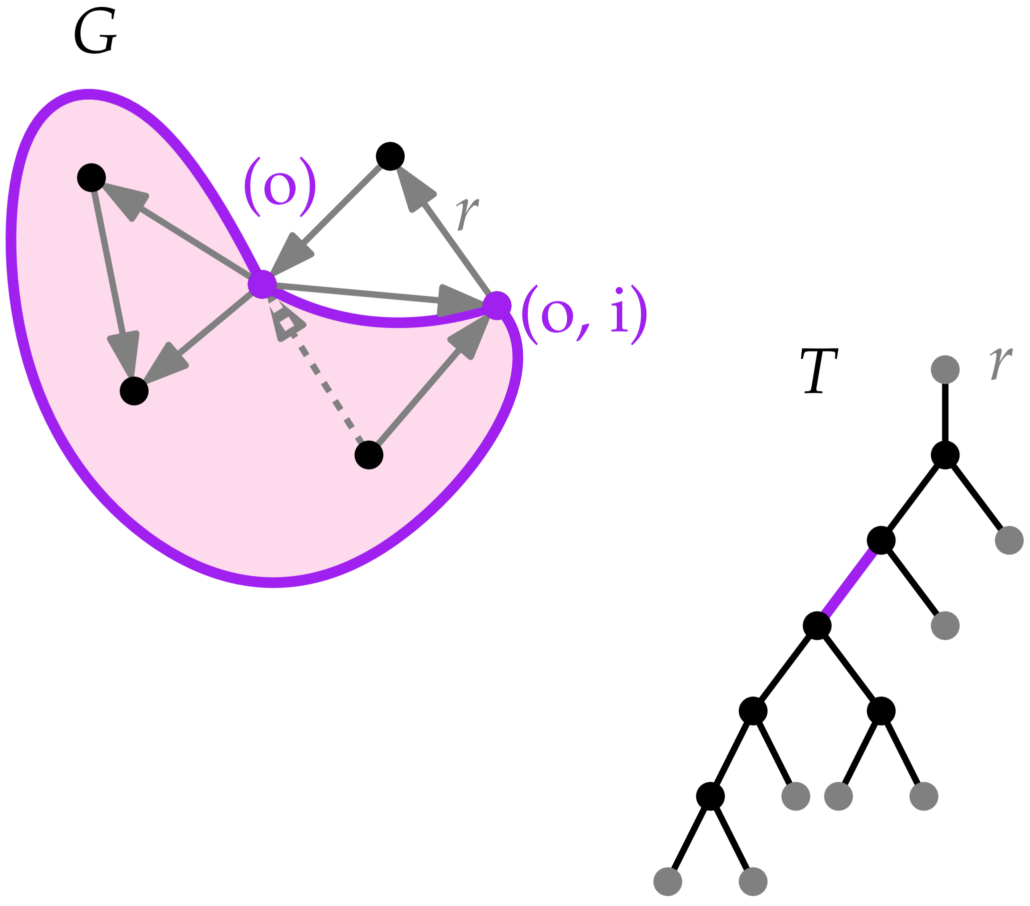


Parametrization by Branchwidth: Proof Sketch

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Proof sketch:

- Compute an optimal Sphere-Cut Decomposition T, root T arbitrarily at a leaf r.
- Let the **inside** of a curve be the side not containing r
- Compute **bottom up** for every curve ϕ_a and every configuration set $\mathcal{X}$ for ϕ_a the maximum subgraph of G that is bimodal in ϕ_a and has $\mathcal{X}$ in ϕ_a.

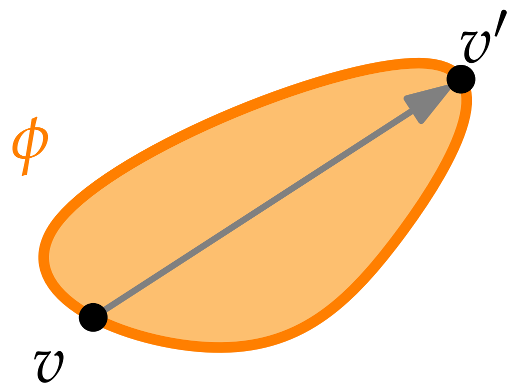


Parametrization by Branchwidth: Proof Sketch

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Proof sketch:

- Base Case: The curve ϕ contains a single edge $e = (v, v')$.

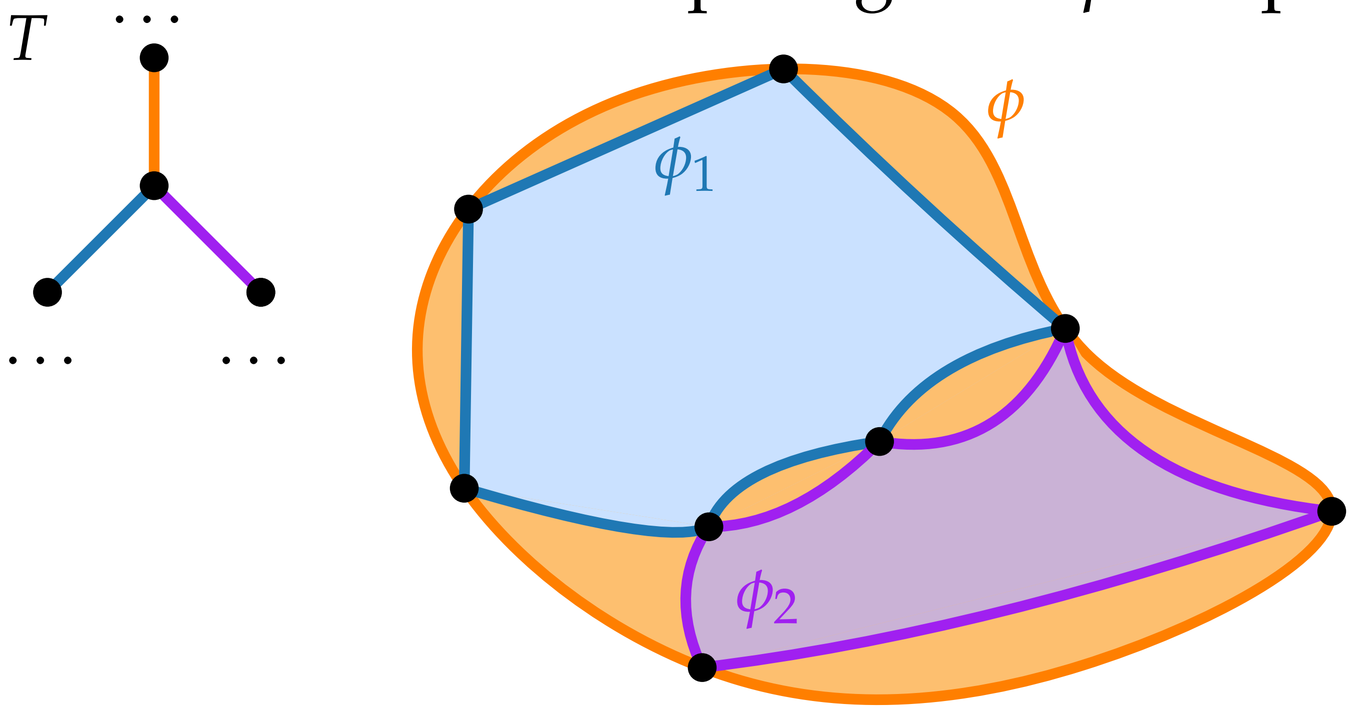


Parametrization by Branchwidth: Proof Sketch

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Proof sketch:

- Inductive Step: edges in ϕ are partitioned by ϕ_1, ϕ_2

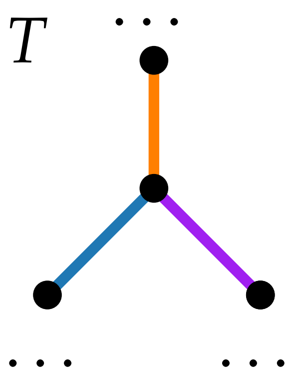


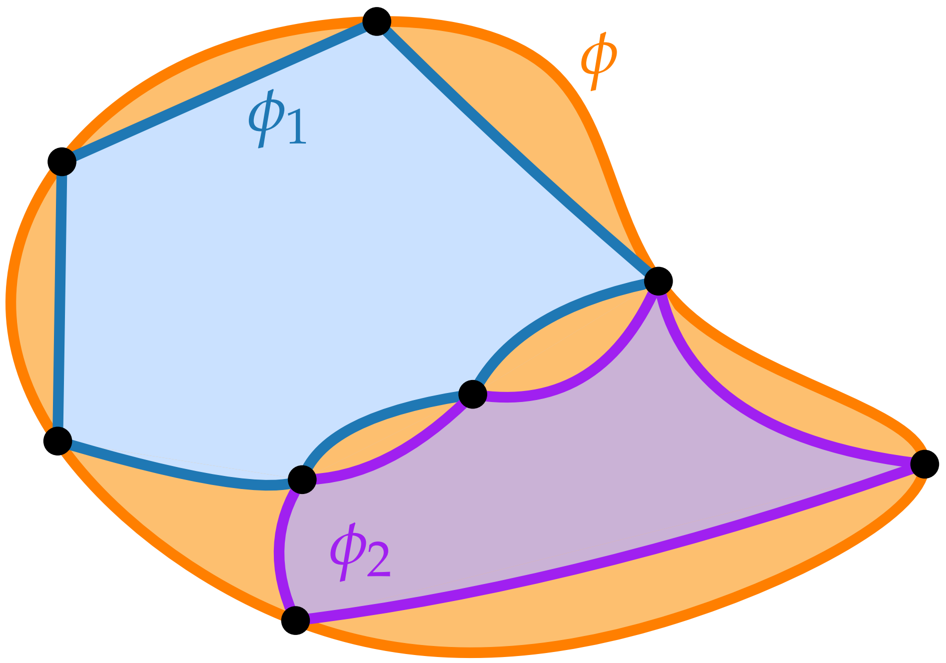
Parametrization by Branchwidth: Proof Sketch

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Proof sketch:

- Inductive Step: edges in ϕ are partitioned by ϕ_1, ϕ_2





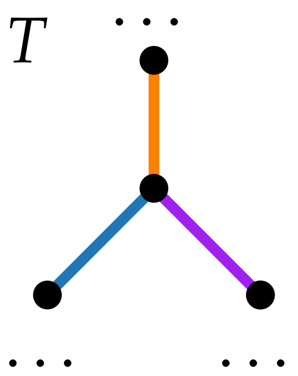
Iterate through every combination of configuration sets $\mathcal{X}, \mathcal{X}_1, \mathcal{X}_2$ for the curve ϕ, ϕ_1, ϕ_2.

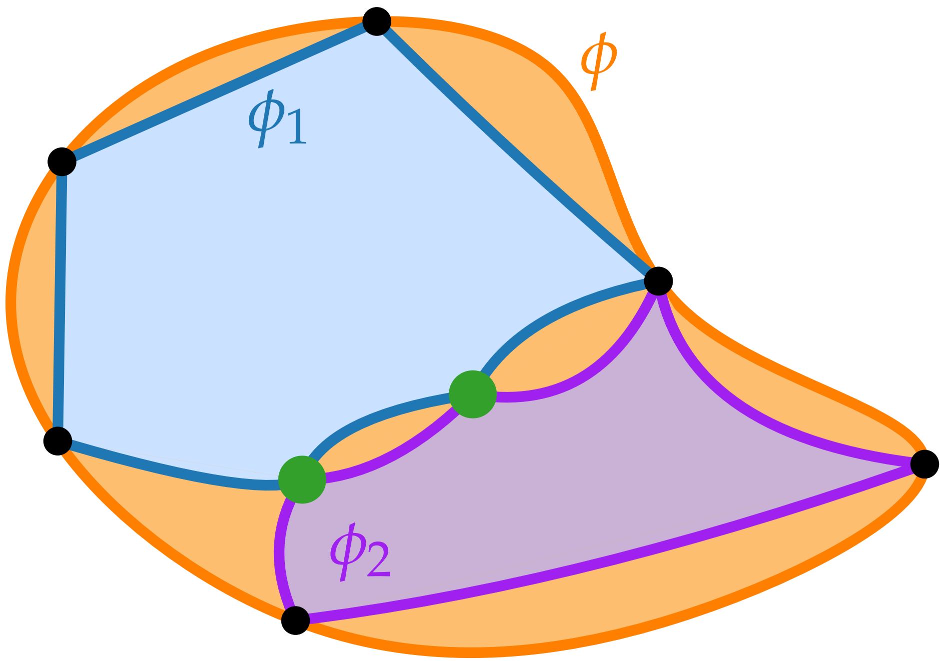
Parametrization by Branchwidth: Proof Sketch

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Proof sketch:

- Inductive Step: edges in ϕ are partitioned by ϕ_1, ϕ_2





Test for every vertex v that is cut by at least one of ϕ, ϕ_1, ϕ_2:

- If v is cut by ϕ_1 and ϕ_2, but not ϕ:

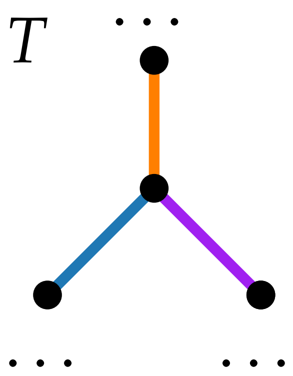
  Are $X_{v,1}, X_{v,2}$ compatible?

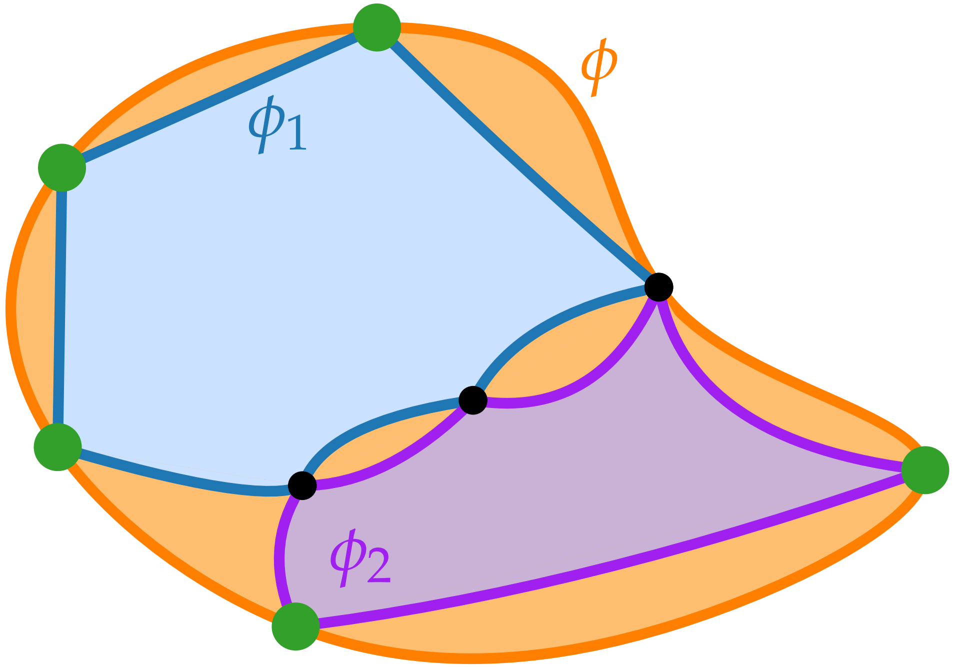
Parametrization by Branchwidth: Proof Sketch

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\text{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Proof sketch:

- Inductive Step: edges in ϕ are partitioned by ϕ_1, ϕ_2





Test for every vertex v that is cut by at least one of ϕ, ϕ_1, ϕ_2:

- If v is cut by ϕ and only one of ϕ_1, ϕ_2:

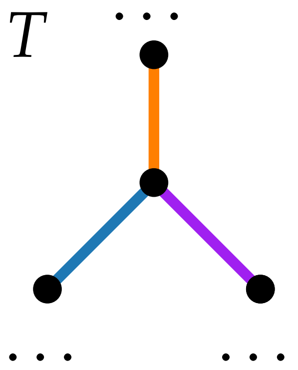
  Is $X_{v,1}$ (or $X_{v,2}$) a substring of X_v?

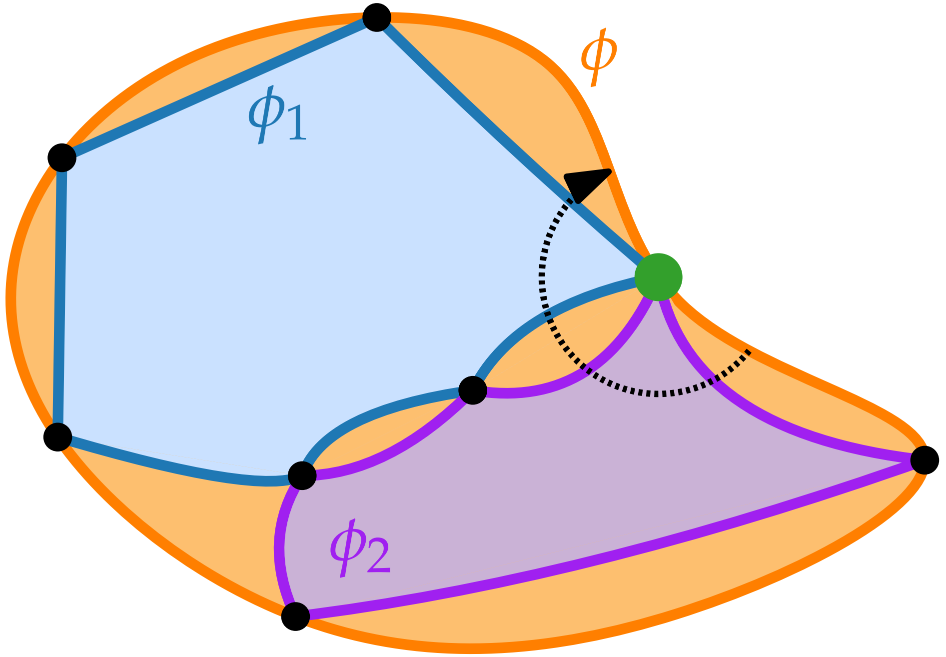
Parametrization by Branchwidth: Proof Sketch

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\text{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Proof sketch:

- Inductive Step: edges in ϕ are partitioned by ϕ_1, ϕ_2





Test for every vertex v that is cut by at least one of ϕ, ϕ_1, ϕ_2:

- If v is cut by all three of ϕ, ϕ_1, ϕ_2:

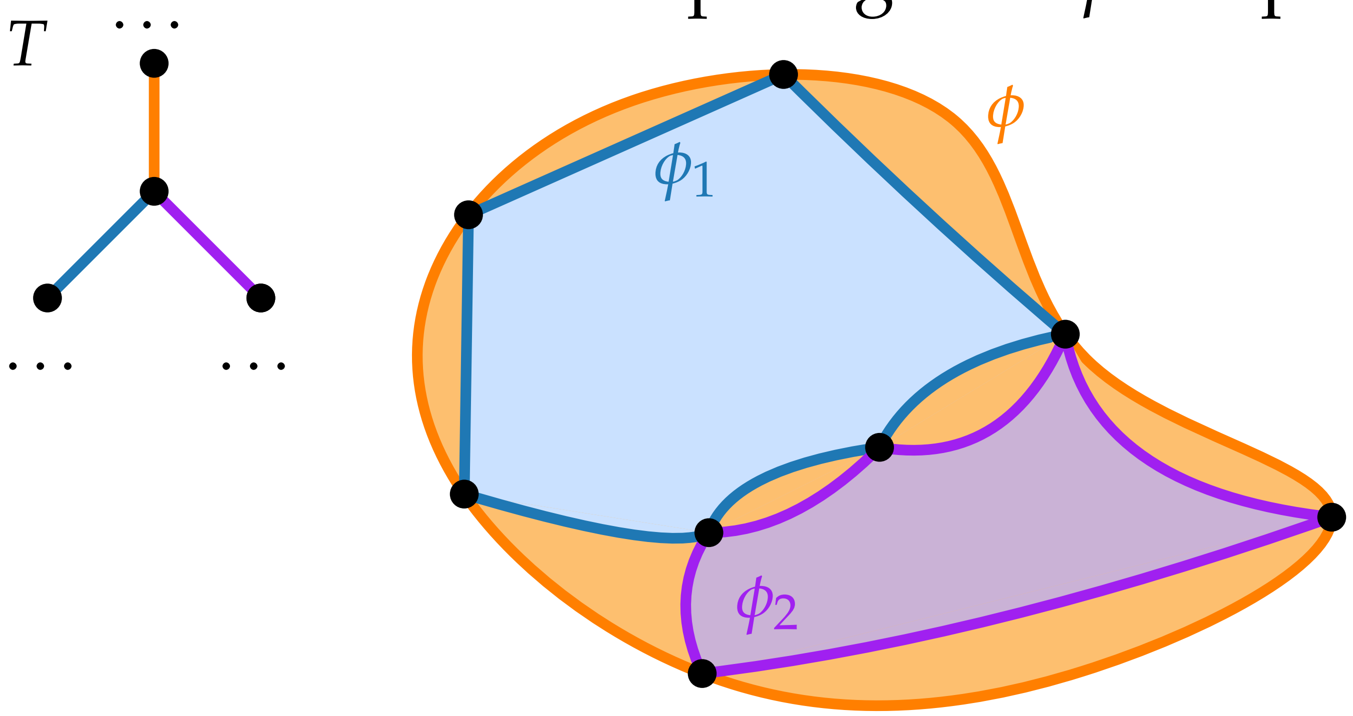
  Are $\mathcal{X}_{v,2}$ and $\mathcal{X}_{v,1}$ compatible with respect to $\mathcal{X}_v$?

Parametrization by Branchwidth: Proof Sketch

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\text{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Proof sketch:

- Inductive Step: edges in ϕ are partitioned by ϕ_1, ϕ_2



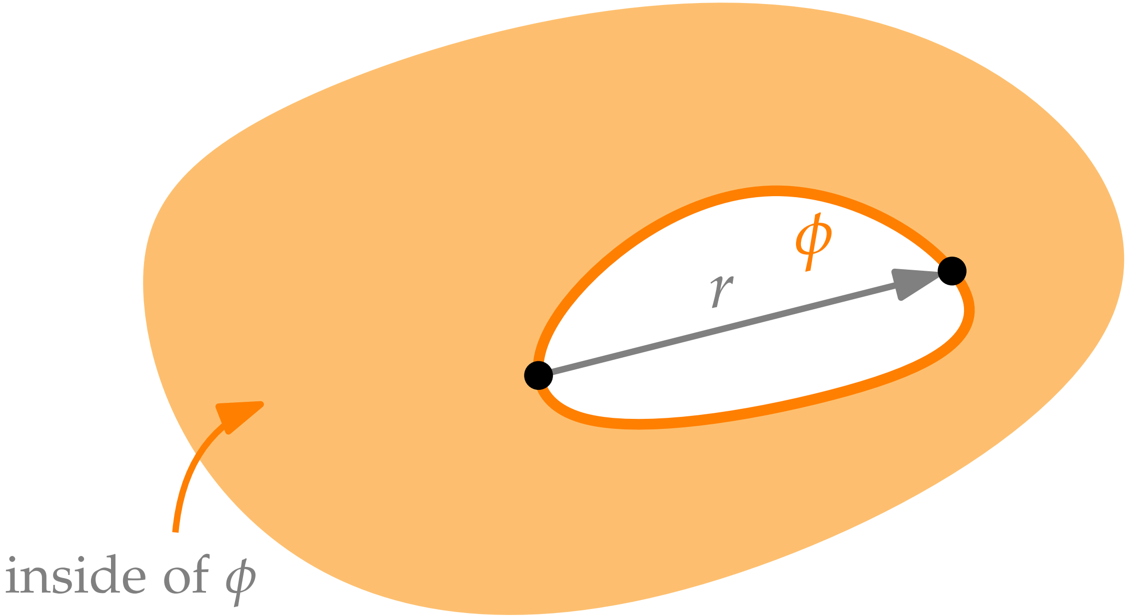
Runtime for one step:
$\mathcal{O}(6^{3\cdot \text{bw}(G)}) \cdot n^{\mathcal{O}(1)} = 2^{\mathcal{O}(\text{bw}(G))} \cdot n^{\mathcal{O}(1)}$.

Parametrization by Branchwidth: Proof Sketch

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Proof sketch:

- Final Step: only root-edge left

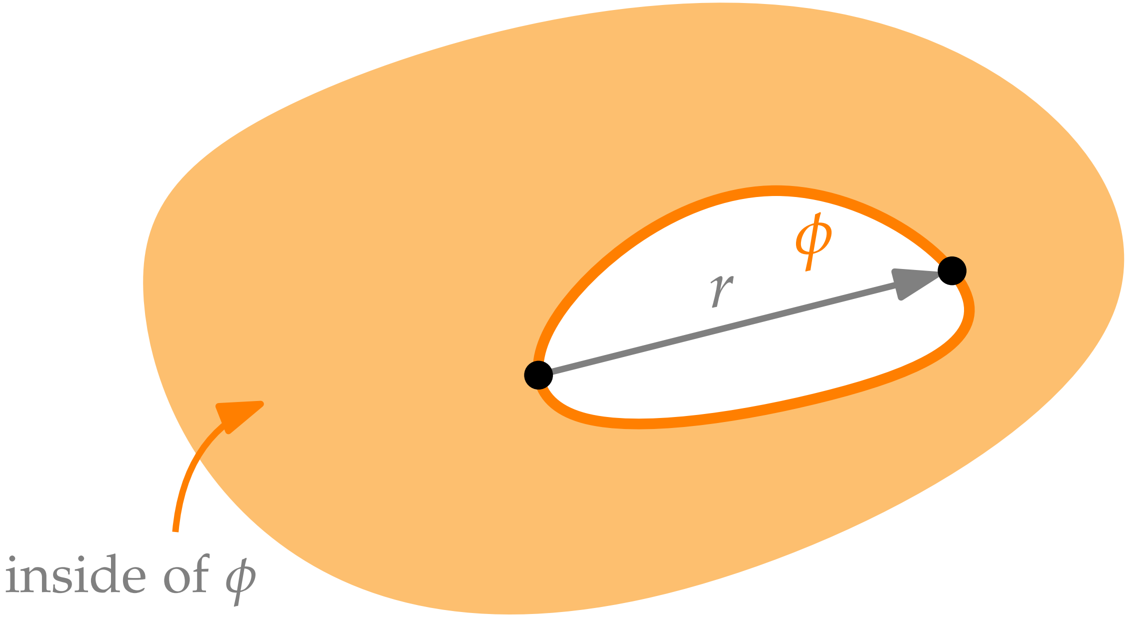


Parametrization by Branchwidth: Proof Sketch

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Proof sketch:

- Final Step: only root-edge left



□

Parametrization by Branchwidth

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Parametrization by Branchwidth

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Since $\mathrm{bw}(G) - 1 \leq \mathrm{tw}(G) \leq \lfloor \frac{3}{2}\mathrm{bw}(G) \rfloor - 1$: [Robertson and Seymour, 1991]

Parametrization by Branchwidth

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Since $\mathrm{bw}(G) - 1 \leq \mathrm{tw}(G) \leq \lfloor \frac{3}{2}\mathrm{bw}(G) \rfloor - 1$: [Robertson and Seymour, 1991]

Corollary 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{tw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by treewidth.

Parametrization by Branchwidth

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Since $\mathrm{bw}(G) - 1 \leq \mathrm{tw}(G) \leq \lfloor \frac{3}{2}\mathrm{bw}(G) \rfloor - 1$: [Robertson and Seymour, 1991]

Corollary 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{tw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by treewidth.

Since the treewidth of a planar graph with n vertices is bounded in $\mathcal{O}(\sqrt{n})$:

Parametrization by Branchwidth

Theorem 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{bw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by branchwidth.

Since $\mathrm{bw}(G) - 1 \leq \mathrm{tw}(G) \leq \lfloor \frac{3}{2}\mathrm{bw}(G) \rfloor - 1$: [Robertson and Seymour, 1991]

Corollary 1:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\mathrm{tw}(G))} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by treewidth.

Since the treewidth of a planar graph with n vertices is bounded in $\mathcal{O}(\sqrt{n})$:

Corollary 2:
There is an algorithm that solves MWBS in $2^{\mathcal{O}(\sqrt{n})}$ time.

Parametrization by the Number of Non-Bimodal Vertices

Theorem 2:
There exists an algorithm that solves MWBS with b non-bimodal vertices in $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by b.

Parametrization by the Number of Non-Bimodal Vertices

Theorem 2:
There exists an algorithm that solves MWBS with b non-bimodal vertices in $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by b.

Proof sketch:

Parametrization by the Number of Non-Bimodal Vertices

Theorem 2:
There exists an algorithm that solves MWBS with b non-bimodal vertices in $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by b.

Proof sketch: Consider the decision version of MWBS with target value W.

Parametrization by the Number of Non-Bimodal Vertices

Theorem 2:
There exists an algorithm that solves MWBS with b non-bimodal vertices in $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by b.

Proof sketch: Consider the decision version of MWBS with target value W.

Reduction Rule 1: Delete isolated vertices.

Parametrization by the Number of Non-Bimodal Vertices

Theorem 2:
There exists an algorithm that solves MWBS with b non-bimodal vertices in $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by b.

Proof sketch: Consider the decision version of MWBS with target value W.

Reduction Rule 1: Delete isolated vertices.

Reduction Rule 2: Delete an edge e that is incident to two bimodal vertices. Reduce W to $W - w(e)$.

Parametrization by the Number of Non-Bimodal Vertices

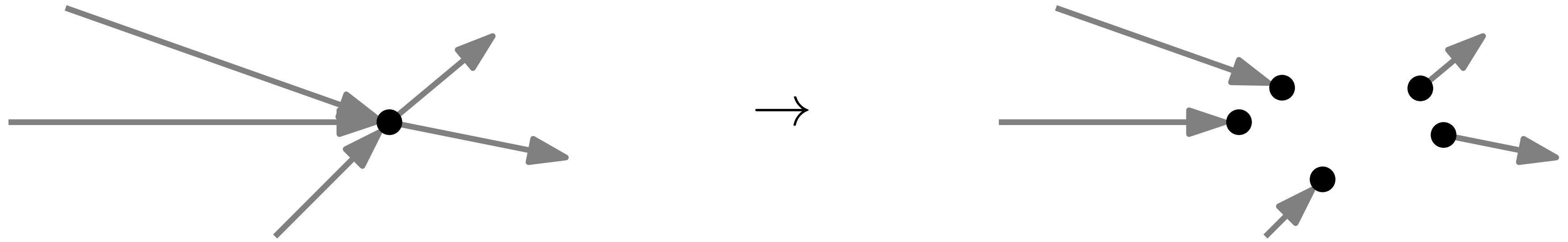
Theorem 2:
There exists an algorithm that solves MWBS with b non-bimodal vertices in $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by b.

Proof sketch: Consider the decision version of MWBS with target value W.

Reduction Rule 1: Delete isolated vertices.

Reduction Rule 2: Delete an edge e that is incident to two bimodal vertices. Reduce W to $W - w(e)$.

Reduction Rule 3: Replace bimodal vertices of degree > 1 with one vertex of degree 1 per incident edge.



Parametrization by the Number of Non-Bimodal Vertices

Theorem 2:
There exists an algorithm that solves MWBS with b non-bimodal vertices in $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by b.

Proof sketch: Consider the decision version of MWBS with target value W.

Reduction Rule 1: Delete isolated vertices.

Reduction Rule 2: Delete an edge e that is incident to two bimodal vertices. Reduce W to $W - w(e)$.

Reduction Rule 3: Replace bimodal vertices of degree > 1 with one vertex of degree 1 per incident edge.

$\rightarrow$ At most b vertices with degree ≥ 2.

Parametrization by the Number of Non-Bimodal Vertices

Theorem 2:
There exists an algorithm that solves MWBS with b non-bimodal vertices in $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by b.

Proof sketch: Consider the decision version of MWBS with target value W.

G' has treewidth bounded in $\mathcal{O}(\sqrt{b})$.

Parametrization by the Number of Non-Bimodal Vertices

Theorem 2:
There exists an algorithm that solves MWBS with b non-bimodal vertices in $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by b.

Proof sketch: Consider the decision version of MWBS with target value W.

G' has treewidth bounded in $\mathcal{O}(\sqrt{b})$.

$\rightarrow$ Use the algorithm from Theorem 1.

Parametrization by the Number of Non-Bimodal Vertices

Theorem 2:
There exists an algorithm that solves MWBS with b non-bimodal vertices in $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by b.

Proof sketch: Consider the decision version of MWBS with target value W.

G' has treewidth bounded in $\mathcal{O}(\sqrt{b})$.

$\rightarrow$ Use the algorithm from Theorem 1.

Running time:

Parametrization by the Number of Non-Bimodal Vertices

Theorem 2:
There exists an algorithm that solves MWBS with b non-bimodal vertices in $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by b.

Proof sketch: Consider the decision version of MWBS with target value W.

G' has treewidth bounded in $\mathcal{O}(\sqrt{b})$.

$\rightarrow$ Use the algorithm from Theorem 1.

Running time: $2^{\mathcal{O}(\mathrm{bw(G')})} \cdot n^{\mathcal{O}(1)}$

Parametrization by the Number of Non-Bimodal Vertices

Theorem 2:
There exists an algorithm that solves MWBS with b non-bimodal vertices in $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by b.

Proof sketch: Consider the decision version of MWBS with target value W.

G' has treewidth bounded in $\mathcal{O}(\sqrt{b})$.

$\rightarrow$ Use the algorithm from Theorem 1.

Running time: $2^{\mathcal{O}(\mathrm{bw(G')})} \cdot n^{\mathcal{O}(1)} = 2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$

Parametrization by the Number of Non-Bimodal Vertices

Theorem 2:
There exists an algorithm that solves MWBS with b non-bimodal vertices in $2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$ time. In particular, MWBS is FPT if parameterized by b.

Proof sketch: Consider the decision version of MWBS with target value W.

G' has treewidth bounded in $\mathcal{O}(\sqrt{b})$.

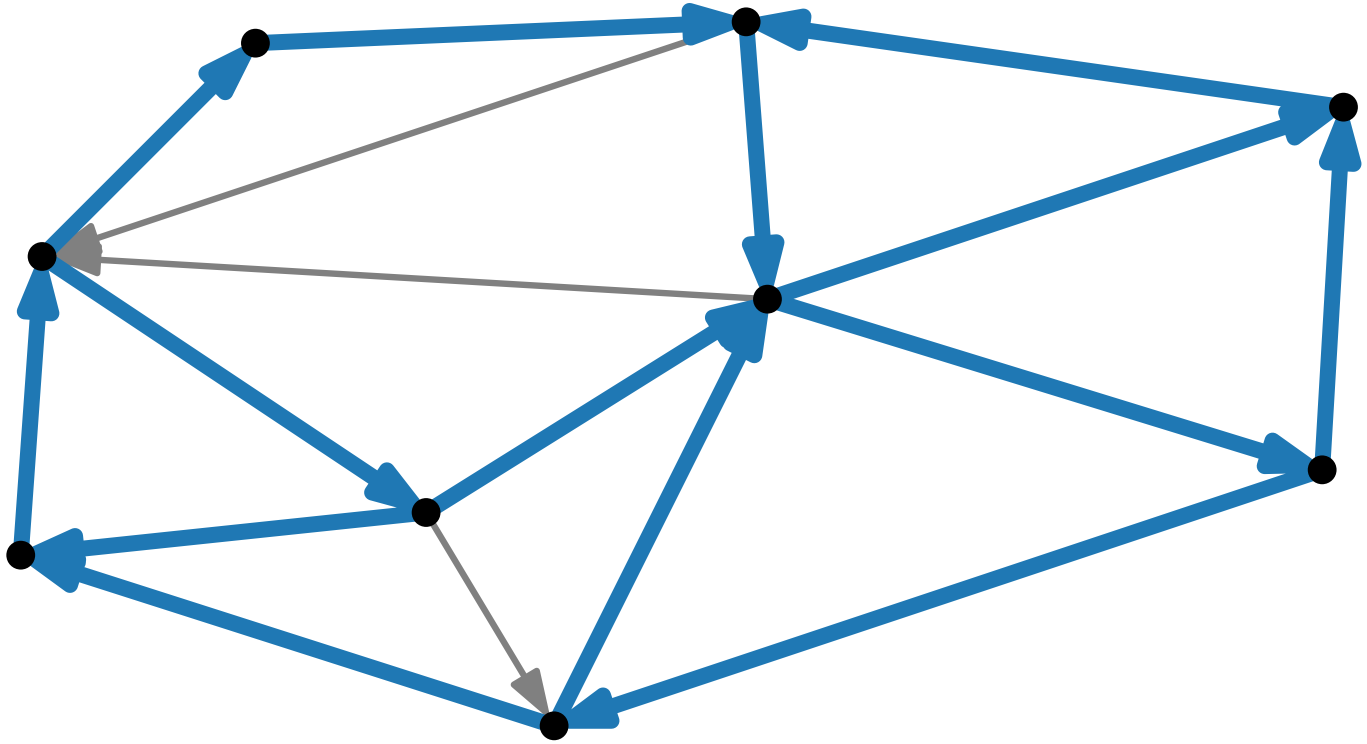
$\rightarrow$ Use the algorithm from Theorem 1.

Running time: $2^{\mathcal{O}(\mathrm{bw}(G'))} \cdot n^{\mathcal{O}(1)} = 2^{\mathcal{O}(\sqrt{b})} \cdot n^{\mathcal{O}(1)}$

□

Open Problems

- Extend to the maximum k-modal subgraph problem for any given even integer $k \geq 2$.

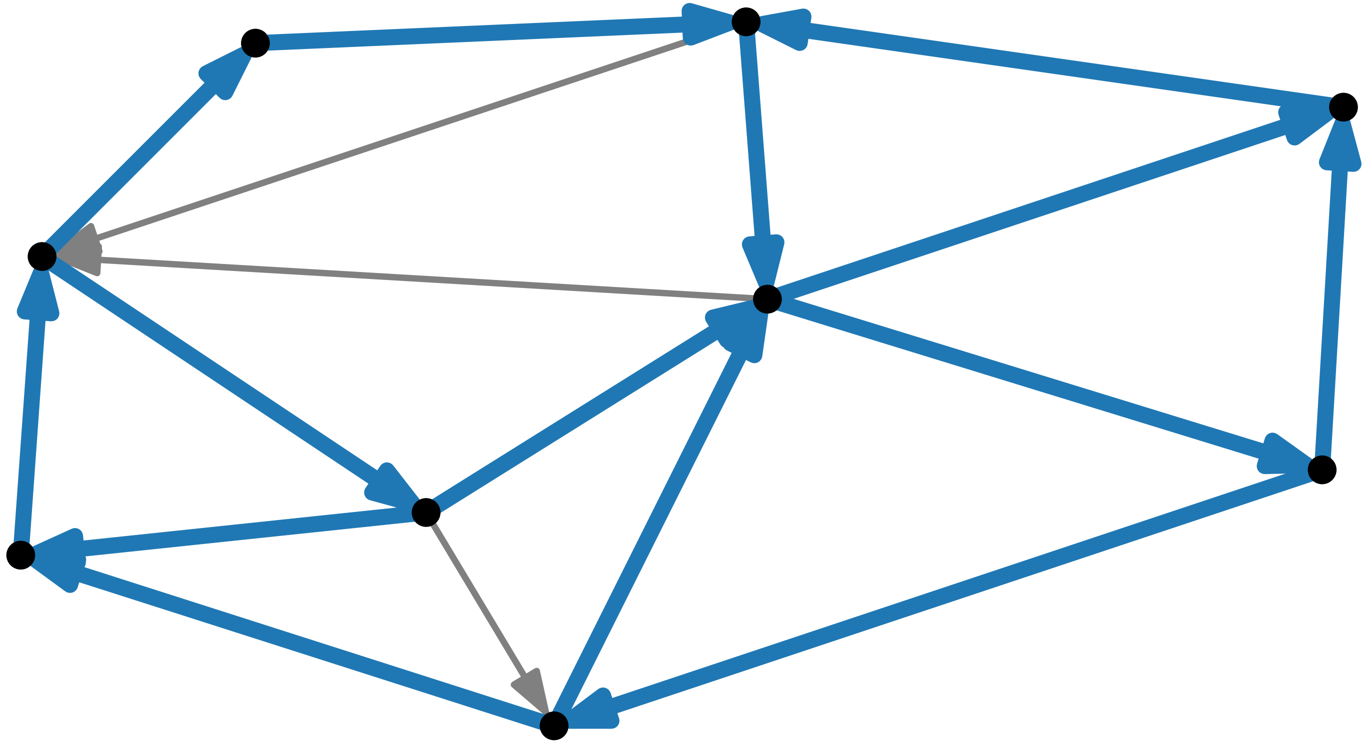


Open Problems

- Extend to the maximum k-modal subgraph problem for any given even integer $k \geq 2$.

- Limit the number of edges that can be deleted by an integer h.

  Possible parameters:
  branchwidth/treewidth; h



Open Problems

- Extend to the maximum k-modal subgraph problem for any given even integer $k \geq 2$.
- Limit the number of edges that can be deleted by an integer h.

  Possible parameters: branchwidth/treewidth; h
- Study MBS in the variable embedding setting.