Multi-Agent Systems

Kai Cai

cai@omu.ac.jp

Graph theory: basic concepts

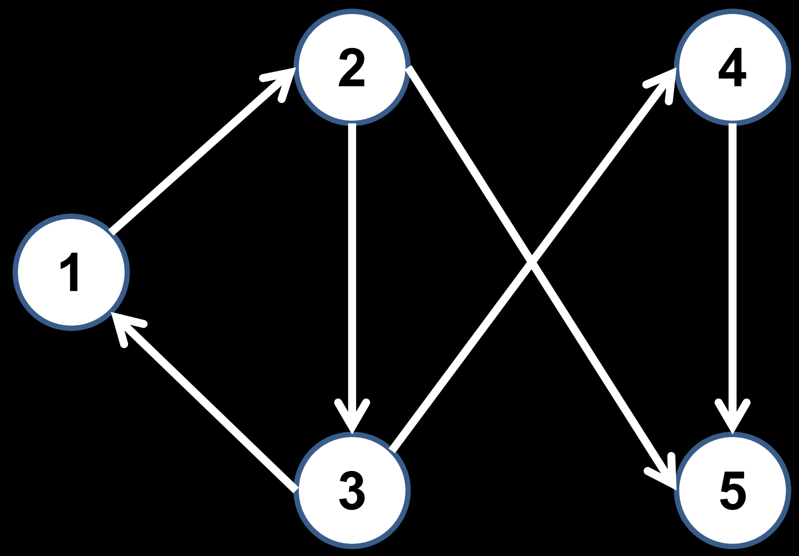
Graph

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

node set $\mathcal{V} = \{v_1, \ldots, v_n\}$

edge set $\mathcal{E} = \{(v_i, v_j), \ldots\}$

example:

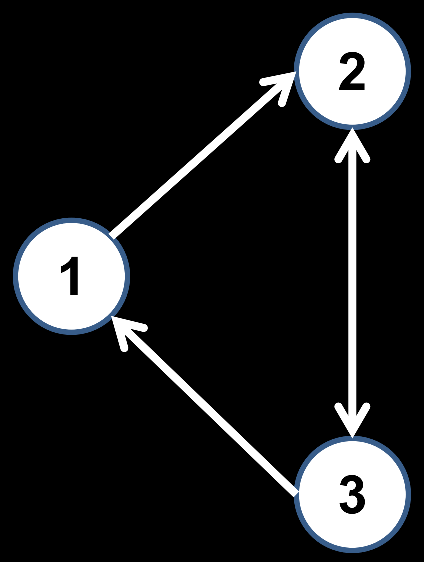


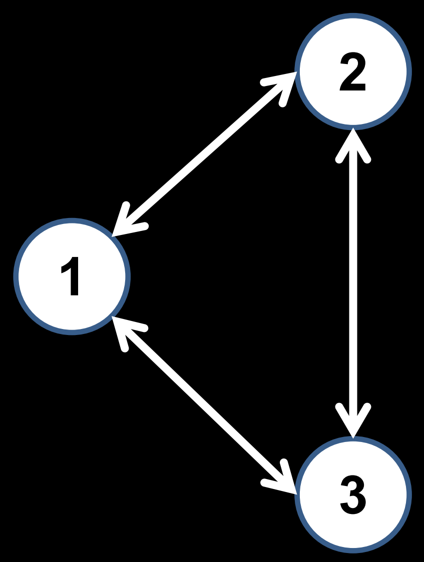
Directed, undirected

generally $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is directed (directed graph, or digraph)

$\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is undirected if $(\forall v_i, v_j \in \mathcal{V})(v_i, v_j) \in \mathcal{E} \Rightarrow (v_j, v_i) \in \mathcal{E}$

example:



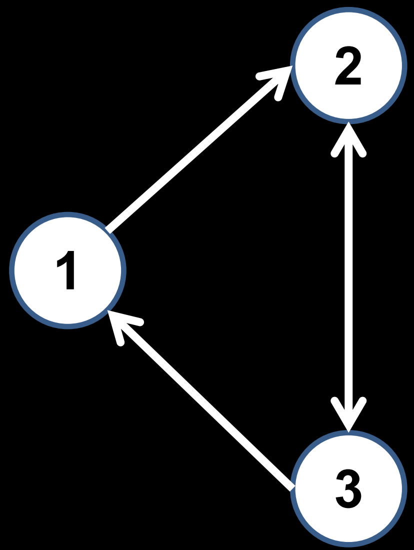


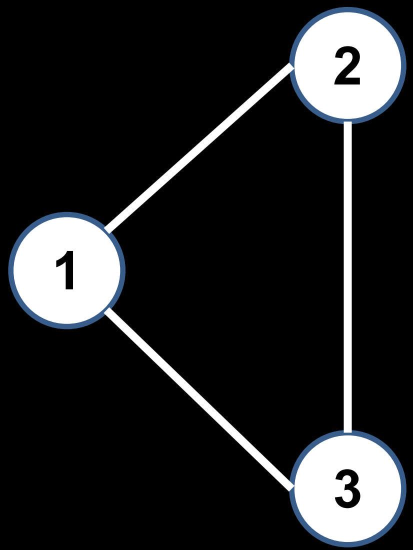
Directed, undirected

generally $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is directed (directed graph, or digraph)

$\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is undirected if
$(\forall v_i, v_j \in \mathcal{V})(v_i, v_j) \in \mathcal{E} \Rightarrow (v_j, v_i) \in \mathcal{E}$

example:

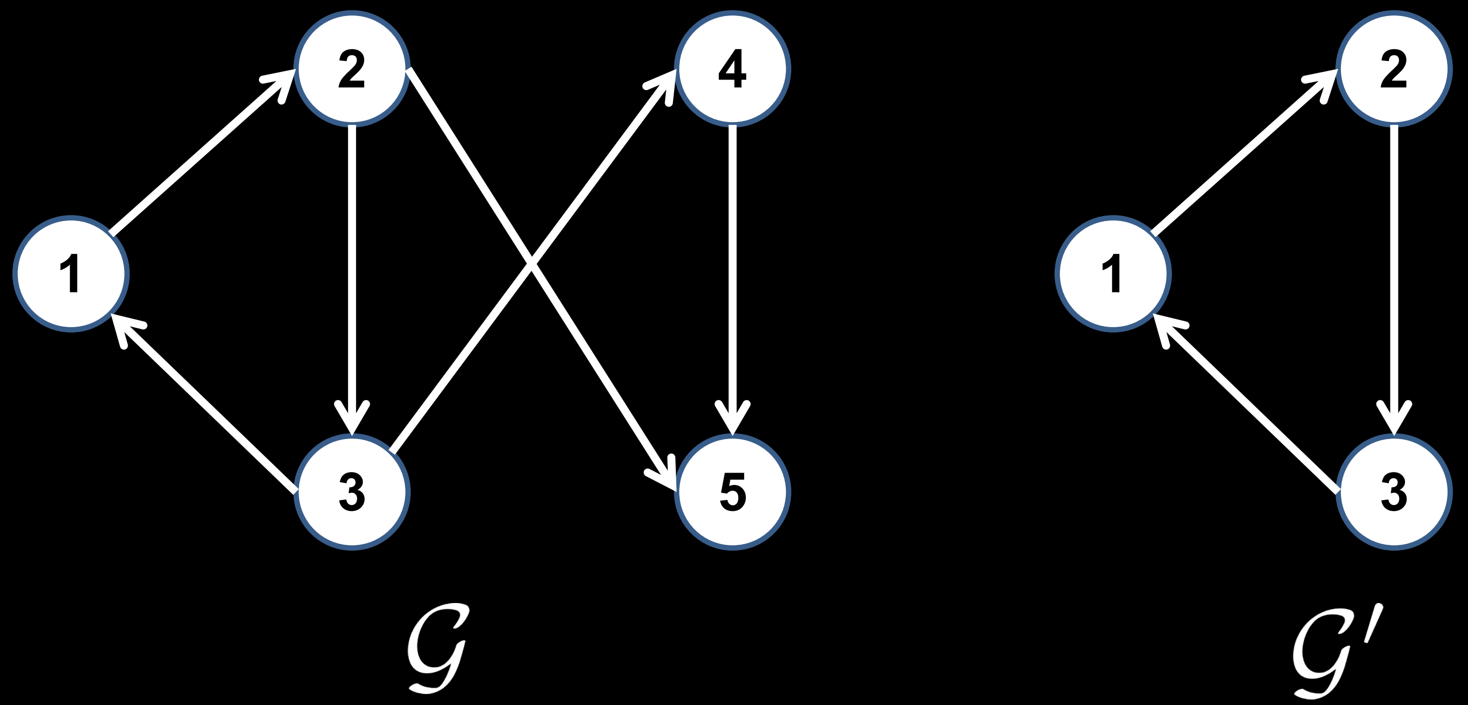




Subgraph

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
graph $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$ is a subgraph of $\mathcal{G}$
if $\mathcal{V}' \subseteq \mathcal{V}$ and $\mathcal{E}' \subseteq \mathcal{E}$

example:

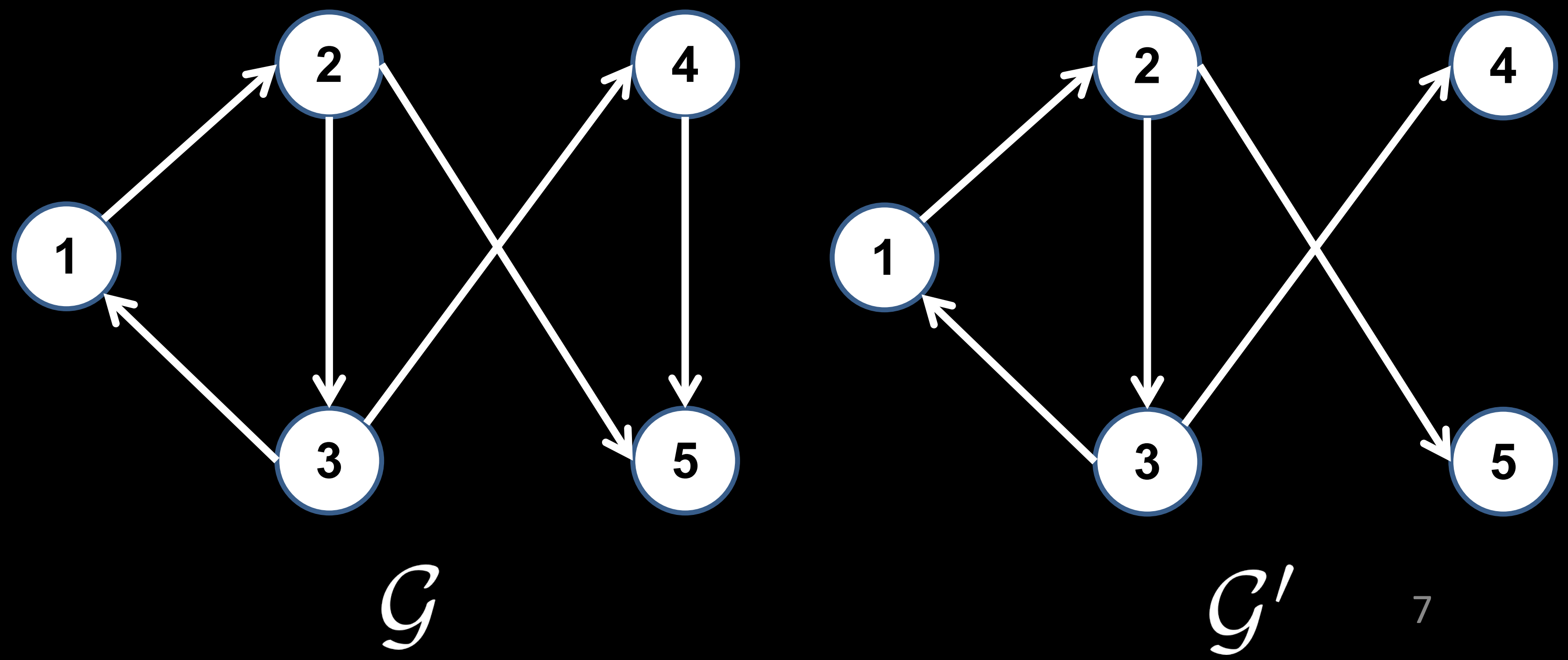


Spanning subgraph

subgraph $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$ of $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

if $\mathcal{V}' = \mathcal{V}$ and $\mathcal{E}' \subseteq \mathcal{E}$

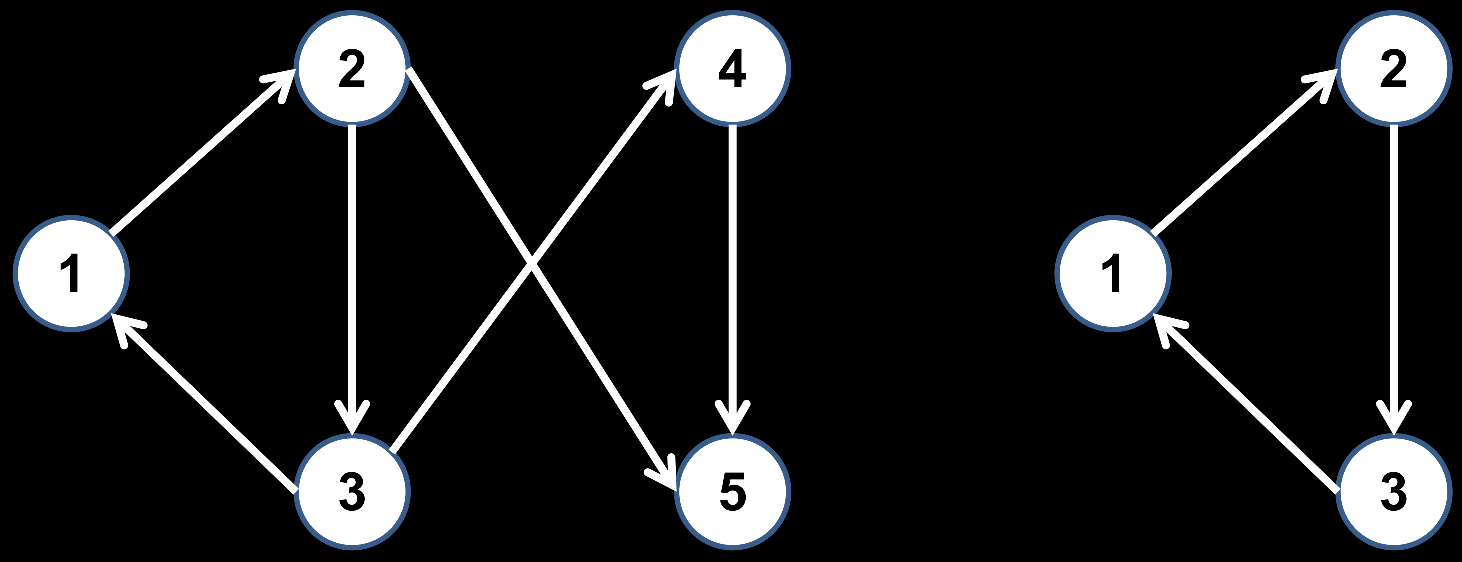
example:



Spanning subgraph

subgraph $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$ of $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
if $\mathcal{V}' = \mathcal{V}$ and $\mathcal{E}' \subseteq \mathcal{E}$

example:

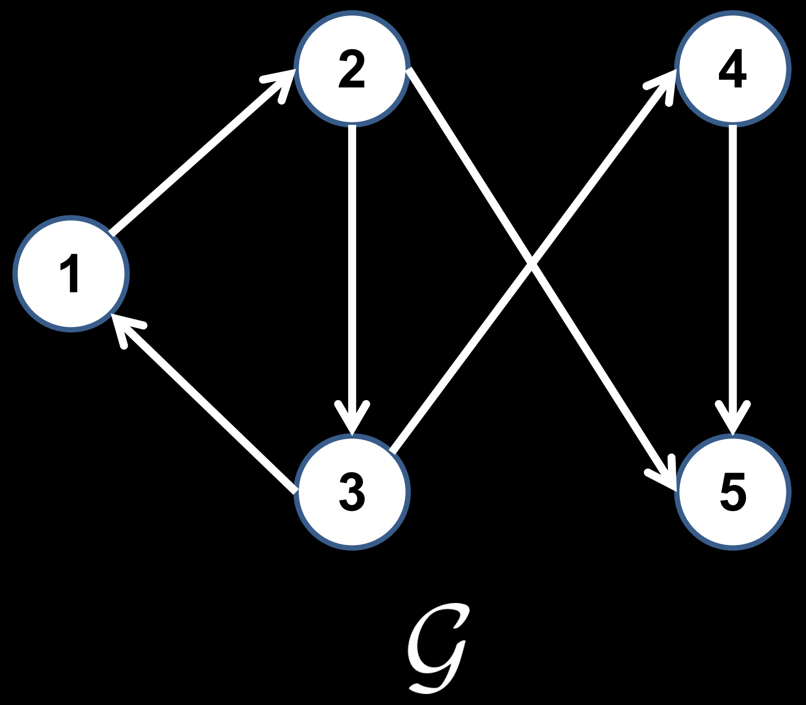


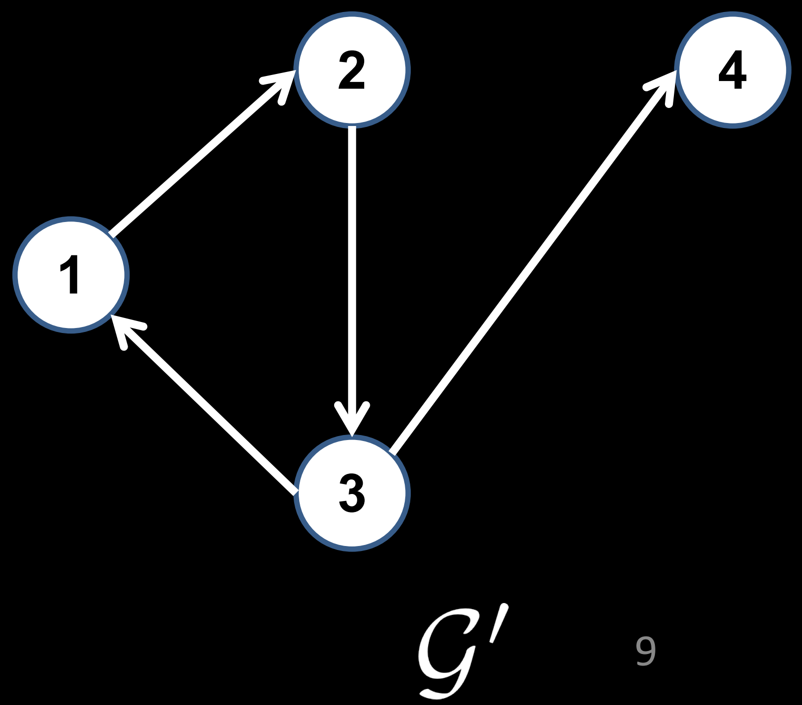
$\mathcal{G}$ $\mathcal{G}'$

Induced subgraph

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and $\emptyset \neq \mathcal{V}' \subseteq \mathcal{V}$
induced subgraph by $\mathcal{V}'$ is
$\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$, $\mathcal{E}' = \mathcal{E} \cap (\mathcal{V}' \times \mathcal{V}')$

example: $\mathcal{V}' = \{1, 2, 3, 4\}$

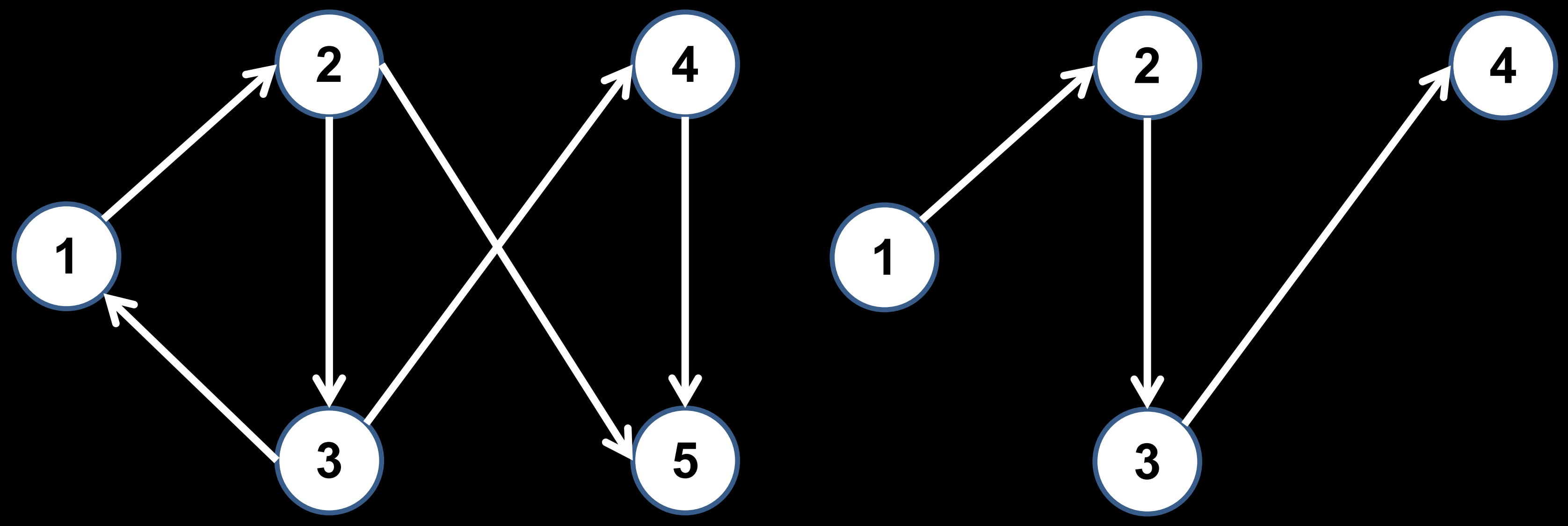




Induced subgraph

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and $\emptyset \neq \mathcal{V}' \subseteq \mathcal{V}$
induced subgraph by $\mathcal{V}'$ is
$\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$, $\mathcal{E}' = \mathcal{E} \cap (\mathcal{V}' \times \mathcal{V}')$

example: $\mathcal{V}' = \{1, 2, 3, 4\}$



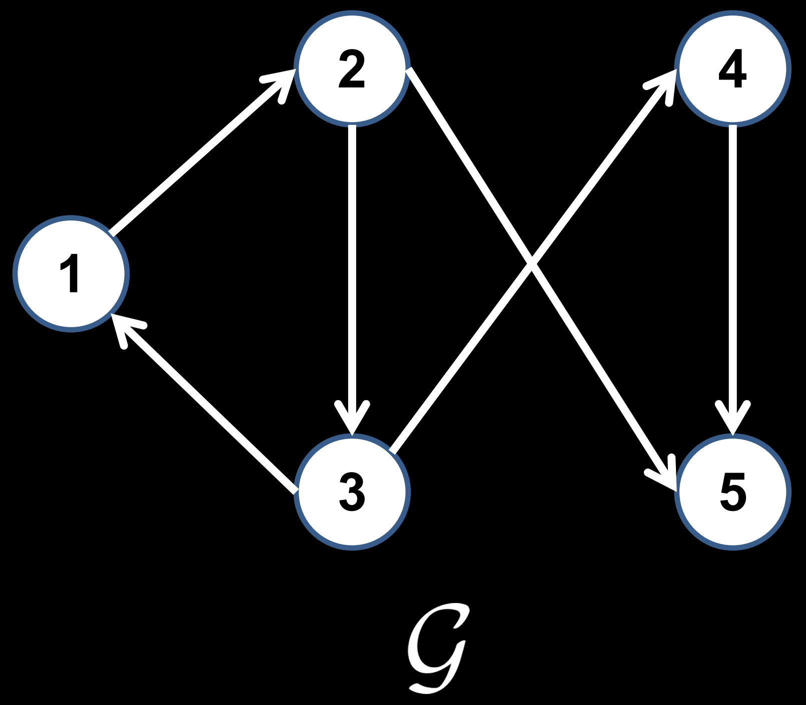
$\mathcal{G}$ $\mathcal{G}'$

Neighbor

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and node $v \in \mathcal{V}$
neighbor set of v is

$$\mathcal{N}_v = \{u \in \mathcal{V} \mid (u, v) \in \mathcal{E}\}$$

example:



$\mathcal{N}_1$
$\mathcal{N}_2$
$\mathcal{N}_3$
$\mathcal{N}_4$
$\mathcal{N}_5$

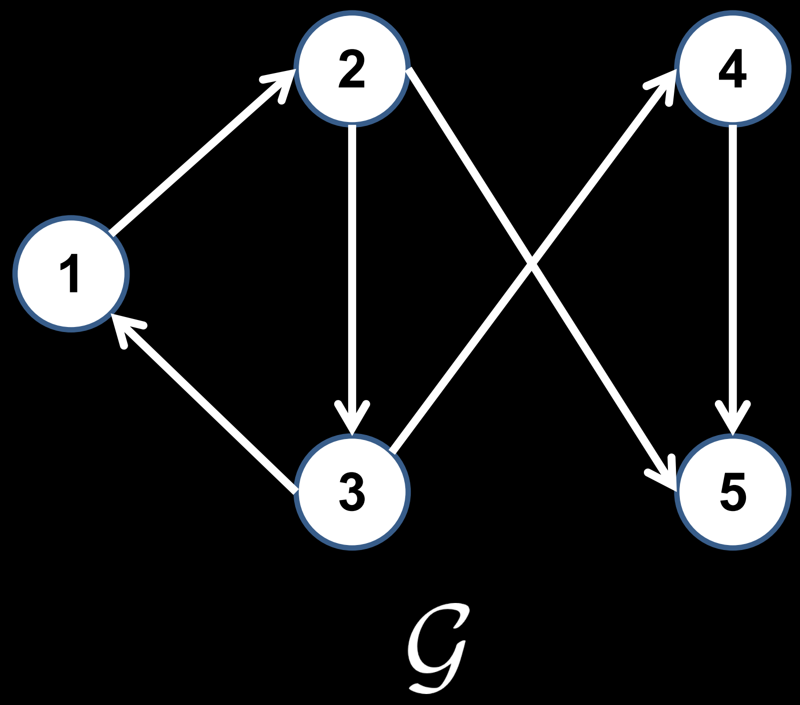
Neighbor

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and node $v \in \mathcal{V}$

out-neighbor set of v is

$$\mathcal{N}_v^o = \{u \in \mathcal{V} \mid (v,u) \in \mathcal{E}\}$$

example:



$\mathcal{N}_1^o$

$\mathcal{N}_2^o$

$\mathcal{N}_3^o$

$\mathcal{N}_4^o$

$\mathcal{N}_5^o$

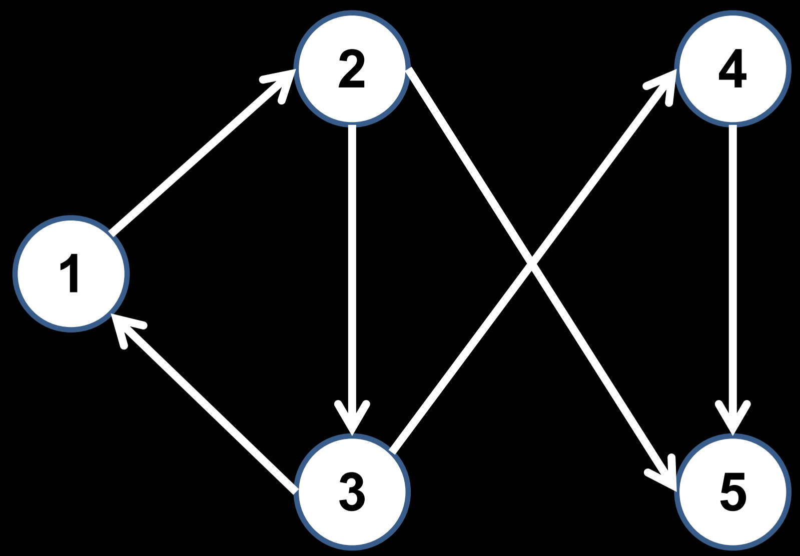
Degree

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and node $v \in \mathcal{V}$

degree of v is $d_v = |\mathcal{N}_v|$

($|\cdot|$: number of elements in the set)

example:



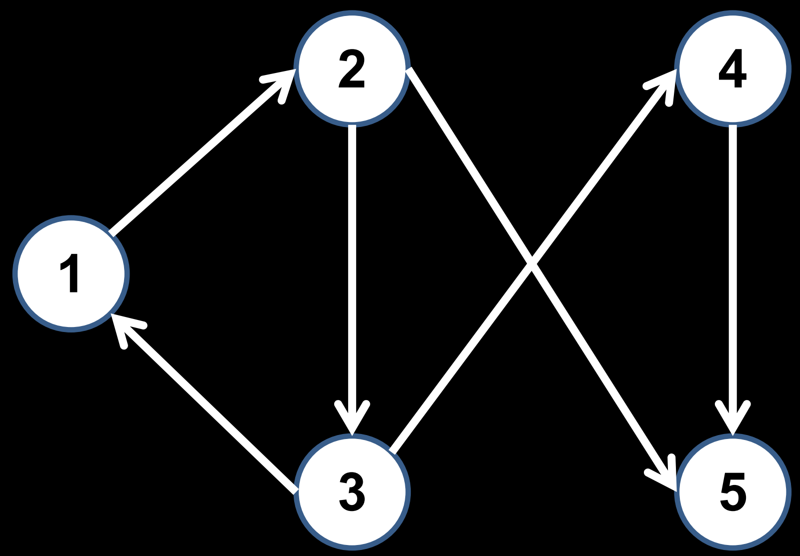
$\mathcal{G}$

Degree

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and node $v \in \mathcal{V}$

out-degree of v is $d_v^o = |\mathcal{N}_v^o|$

example:

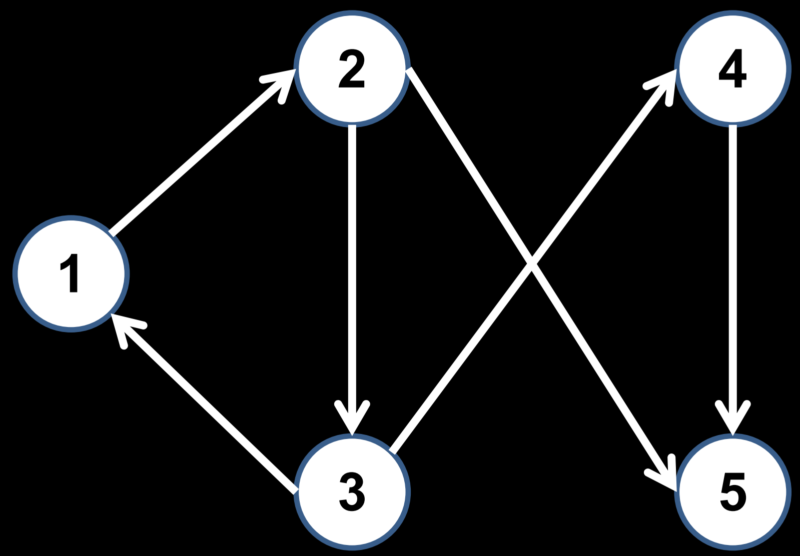


$\mathcal{G}$

Balanced graphs

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
node $v \in \mathcal{V}$ is balanced if $d_v = d_v^o$

example:

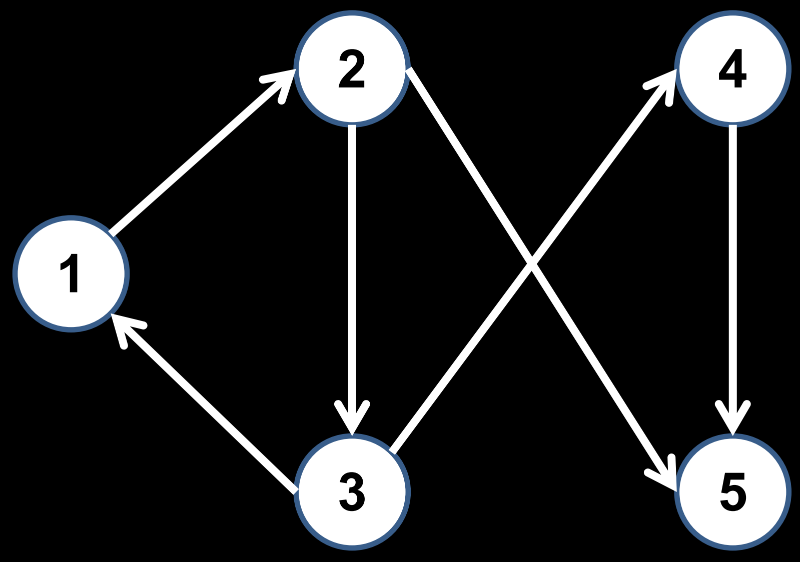


$\mathcal{G}$

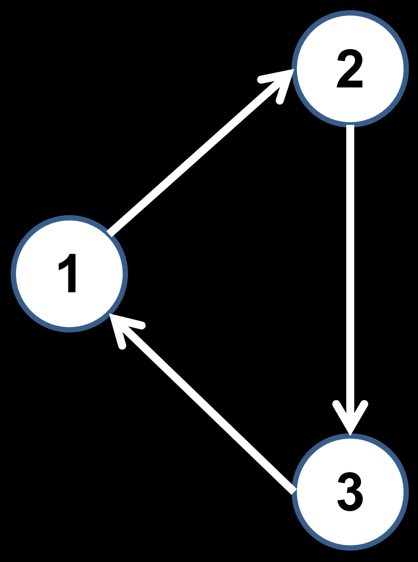
Balanced graphs

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
node $v \in \mathcal{V}$ is balanced if $d_v = d_v^o$
$\mathcal{G}$ is balanced if every v is balanced
(all undirected graphs are balanced)

example:



$\mathcal{G}$



$\mathcal{G}'$

Graph theory: connectivity

Path

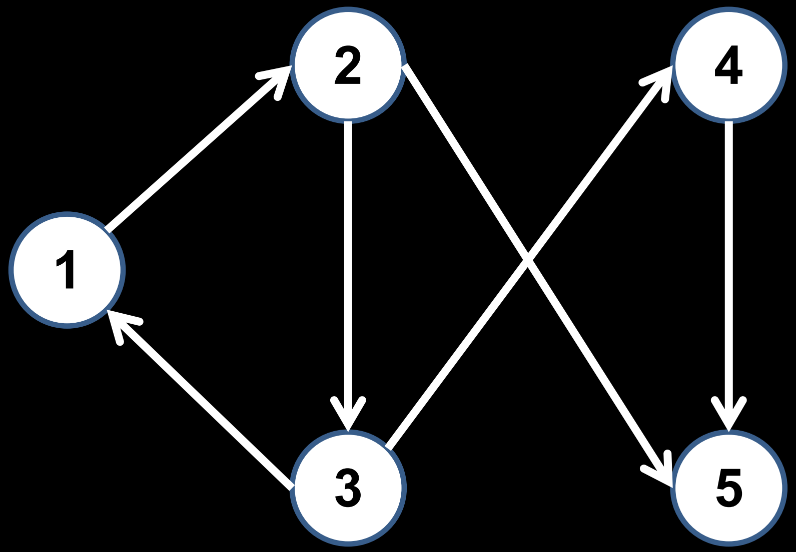
graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

a path in $\mathcal{G}$ is a sequence of nodes

$$v_1 v_2 \cdots v_k \quad (k \geq 1)$$

s.t. $(v_i, v_{i+1}) \in \mathcal{E}$ for $i \in [1, k-1]$

example:



$\mathcal{G}$

1 2 3 4 5

1 2 3 1

4

Path

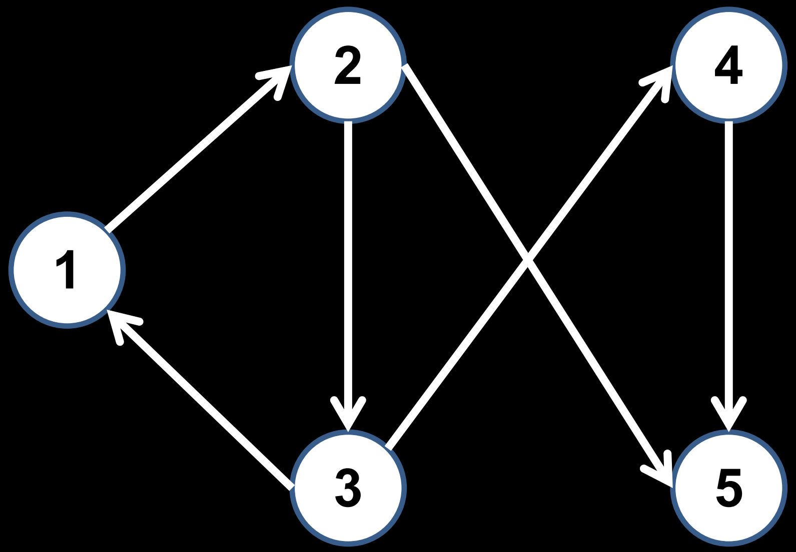
graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

a path in $\mathcal{G}$

$$v_1 v_2 \cdots v_k \quad (k \geq 1)$$

has length $k - 1$

example:



$\mathcal{G}$

1 2 3 4 5

1 2 3 1

4

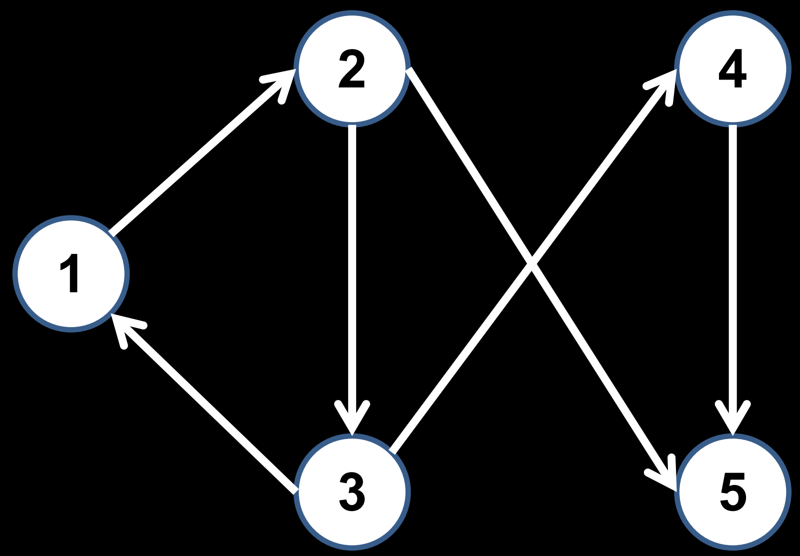
Path

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

$v_1 v_2 \cdots v_k$ is a path from v_1 to v_k

if $v_1 = v_k$, $v_1 v_2 \cdots v_k$ is a cycle

example:



$\mathcal{G}$

1 2 3 4 5

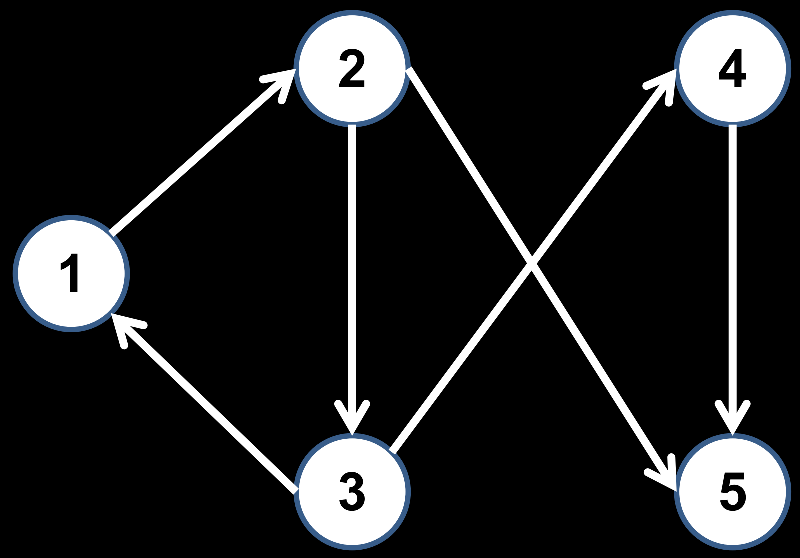
1 2 3 1

4

Reachability

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, two nodes $v_i, v_j \in \mathcal{V}$
v_i is reachable from v_j if
there is a path from v_j to v_i

example:



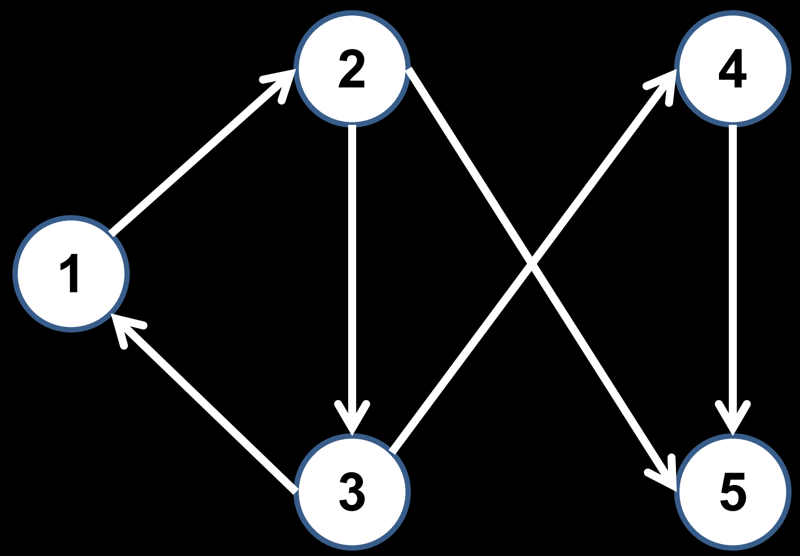
$\mathcal{G}$

Reachability

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, two nodes $v_i, v_j \in \mathcal{V}$

every node is reachable from itself

example:



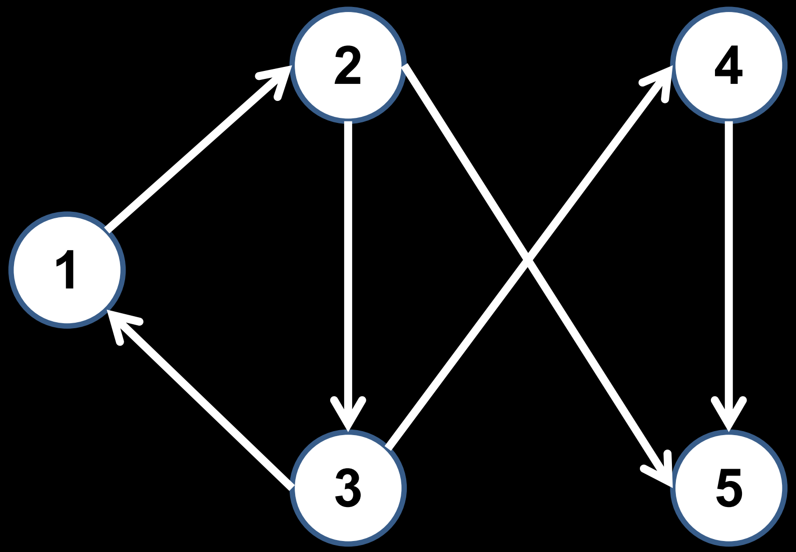
$\mathcal{G}$

Strongly connected

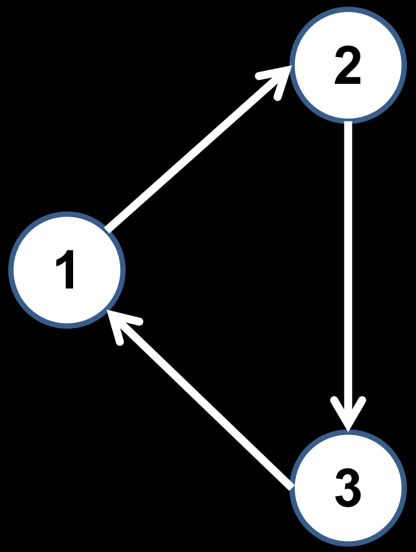
graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is strongly connected

if $(\forall v_i, v_j \in \mathcal{V})$ v_i is reachable from v_j

example:



$\mathcal{G}_1$



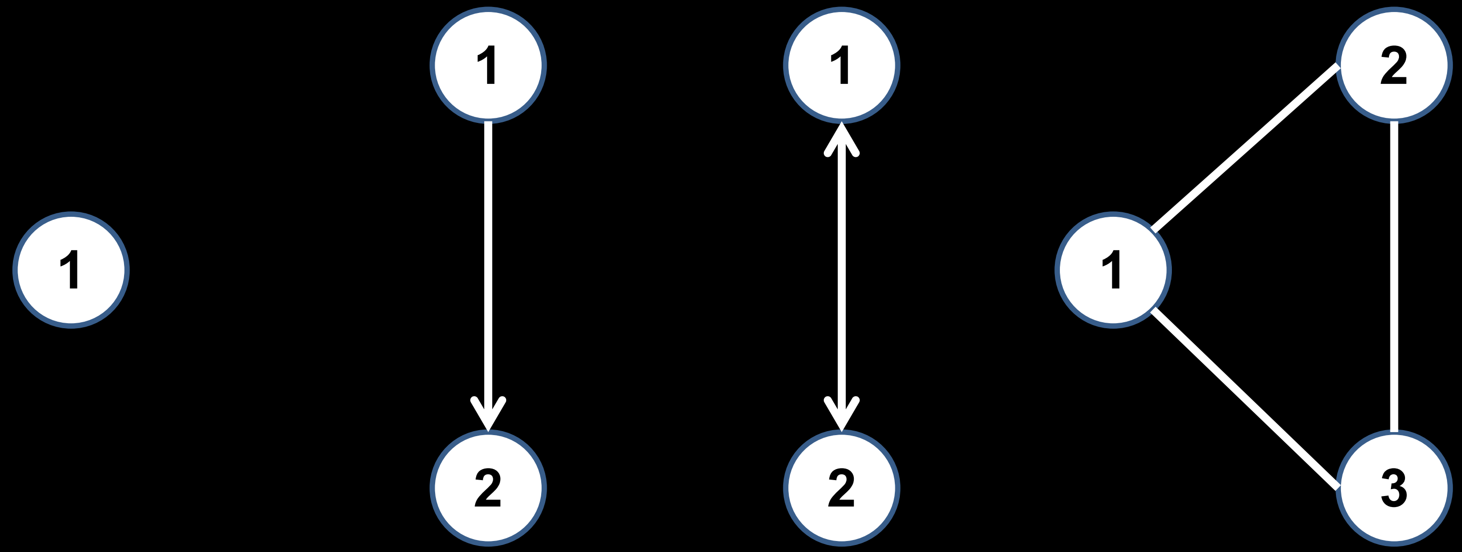
$\mathcal{G}_2$

Strongly connected

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is strongly connected

if $(\forall v_i, v_j \in \mathcal{V})$ v_i is reachable from v_j

example:



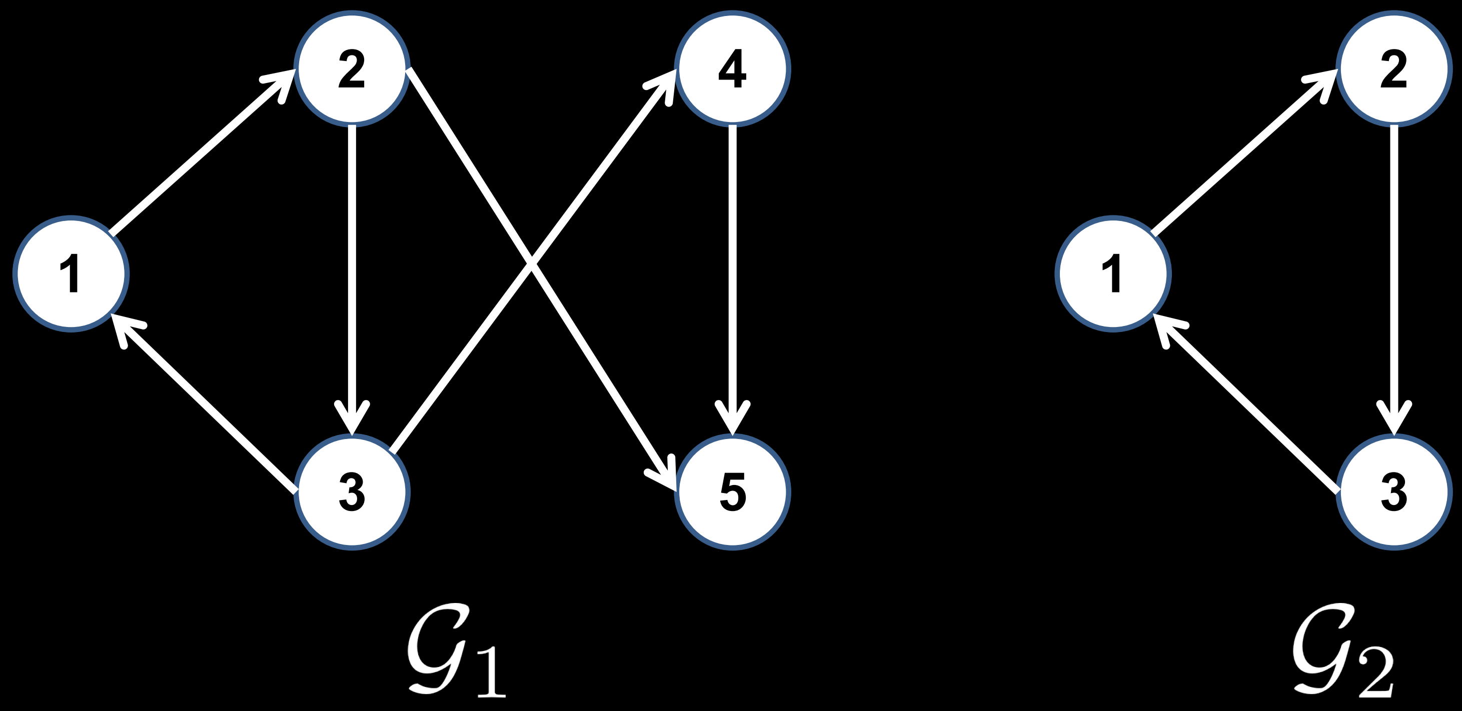
$\mathcal{G}_3$ $\mathcal{G}_4$ $\mathcal{G}_5$ $\mathcal{G}_6$

Root

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

node $v \in \mathcal{V}$ is a root if
every node is reachable from v

example:

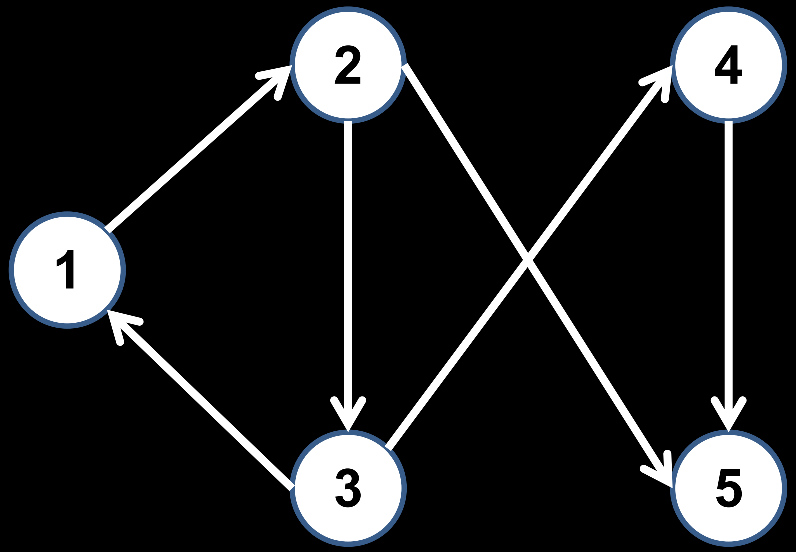


Root

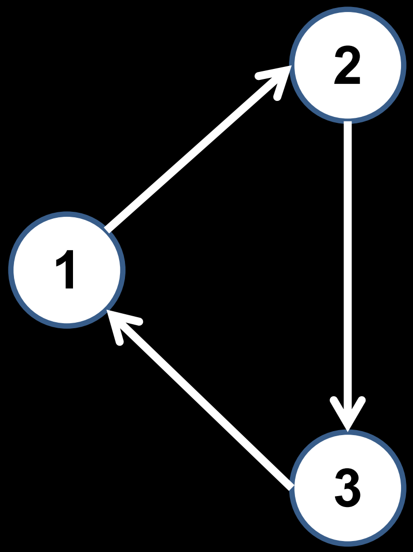
graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

$\mathcal{G}$ is strongly connected iff all nodes are roots

example:



$\mathcal{G}_1$

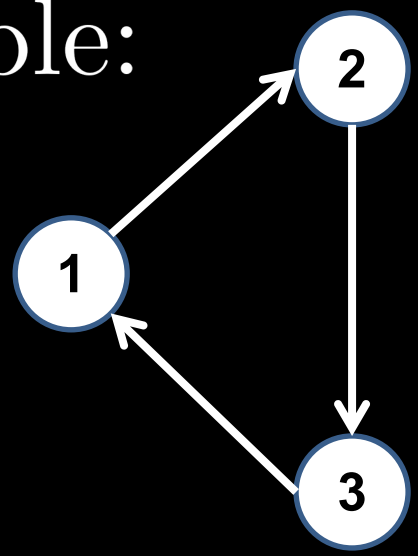


$\mathcal{G}_2$

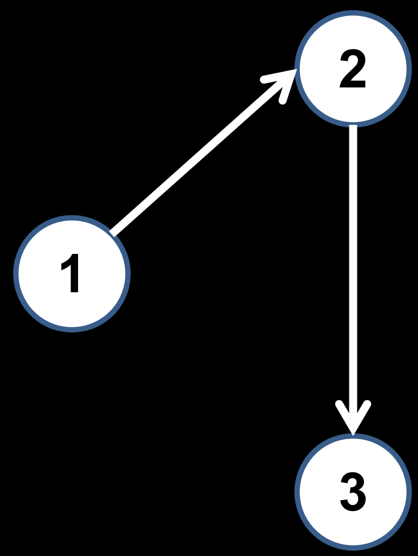
Spanning tree

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and root $r \in \mathcal{V}$
a spanning subgraph $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$ is
a spanning tree with root r if
1) r has no neighbor, i.e. $\mathcal{N}_r = \emptyset$
2) every $v \in \mathcal{V} \setminus \{r\}$ has one neighbor

example:



$\mathcal{G}$

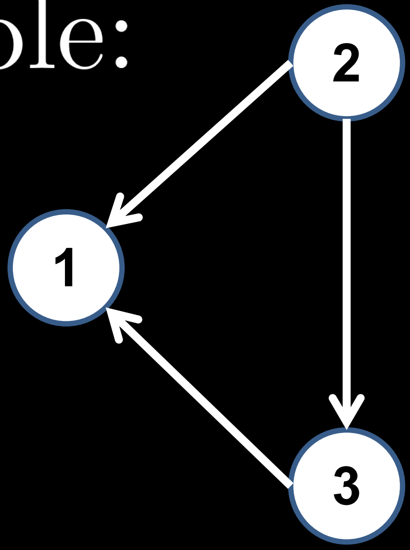


$\mathcal{G}'$

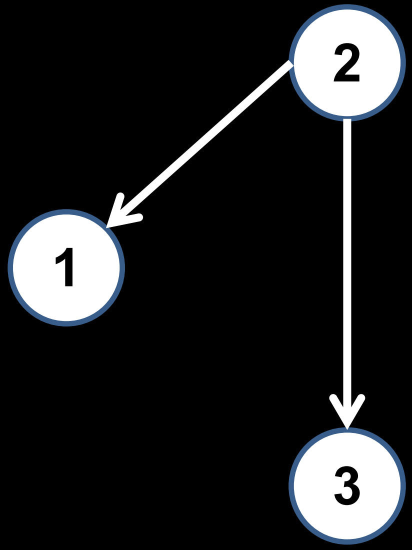
Spanning tree

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and root $r \in \mathcal{V}$
a spanning subgraph $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$ is
a spanning tree with root r if
1) r has no neighbor, i.e. $\mathcal{N}_r = \emptyset$
2) every $v \in \mathcal{V} \setminus \{r\}$ has one neighbor

example:



$\mathcal{G}$

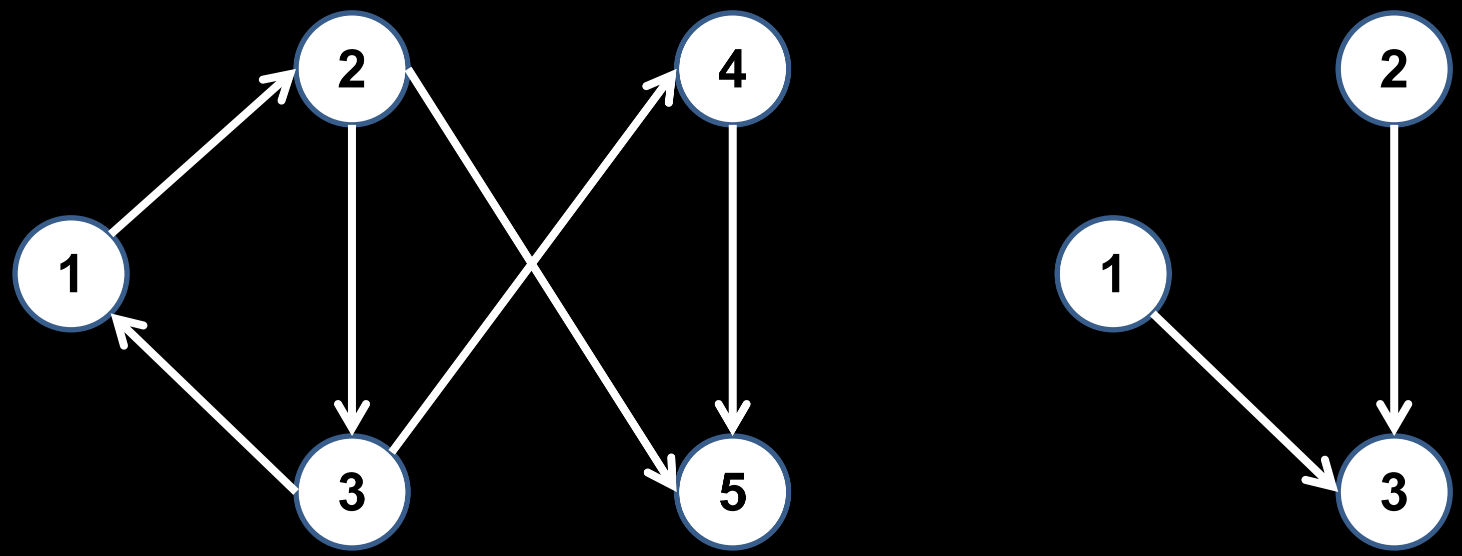


$\mathcal{G}'$

Spanning tree

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ contains a spanning tree if there exists a subgraph of $\mathcal{G}$ that is a spanning tree

example:



$\mathcal{G}_1$ $\mathcal{G}_2$

Spanning tree

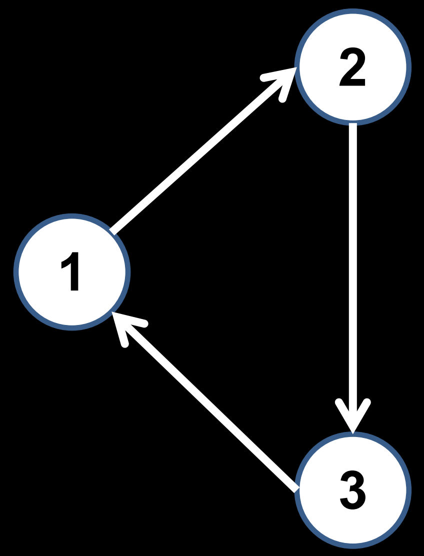
graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

$\mathcal{G}$ is strongly connected

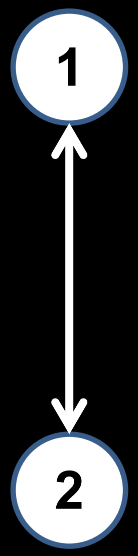
$$\Downarrow$$

$\mathcal{G}$ contains a spanning tree

example:



$\mathcal{G}_1$



$\mathcal{G}_2$

Spanning tree

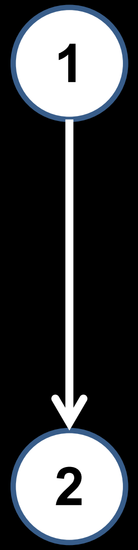
graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

$\mathcal{G}$ is strongly connected

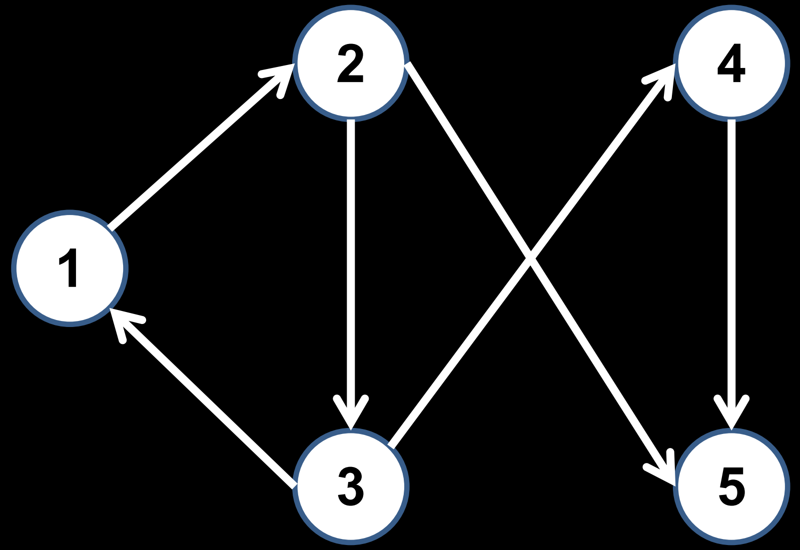
$$\Downarrow$$

$\mathcal{G}$ contains a spanning tree

example:



$\mathcal{G}_3$

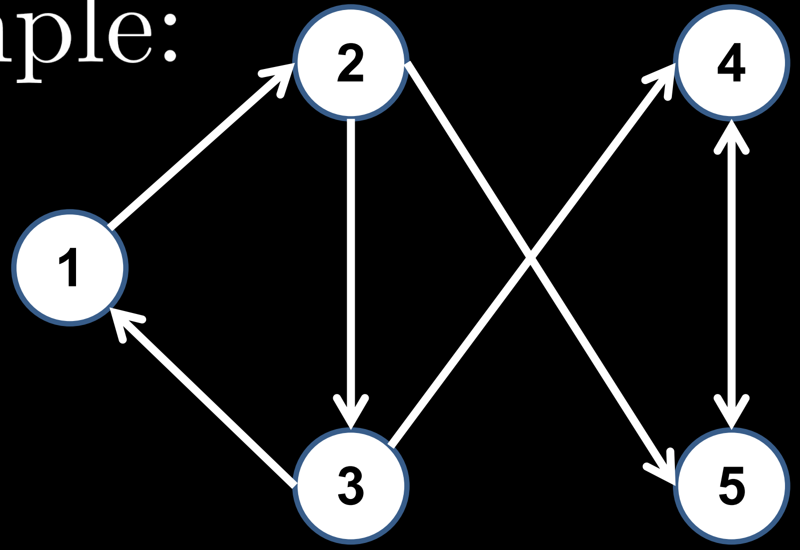


$\mathcal{G}_4$

Strong component

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and $\emptyset \neq \mathcal{V}' \subseteq \mathcal{V}$
induced subgraph $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$ is a strong component of $\mathcal{G}$ if
$\mathcal{G}'$ is a maximally strongly connected induced subgraph of $\mathcal{G}$

example:



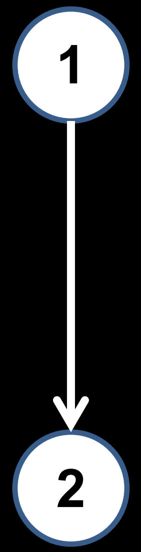
$\mathcal{G}$

$\mathcal{G}'$

Strong component

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and $\emptyset \neq \mathcal{V}' \subseteq \mathcal{V}$
induced subgraph $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$ is a strong component of $\mathcal{G}$ if
$\mathcal{G}'$ is a maximally strongly connected induced subgraph of $\mathcal{G}$

example:



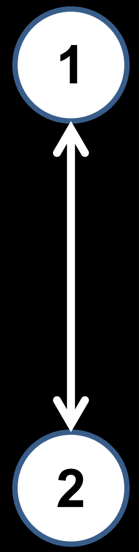
$\mathcal{G}$

$\mathcal{G}'$

Strong component

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and $\emptyset \neq \mathcal{V}' \subseteq \mathcal{V}$
induced subgraph $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$ is a strong component of $\mathcal{G}$ if
$\mathcal{G}'$ is a maximally strongly connected induced subgraph of $\mathcal{G}$

example:



$\mathcal{G}$

$\mathcal{G}'$

Strong component

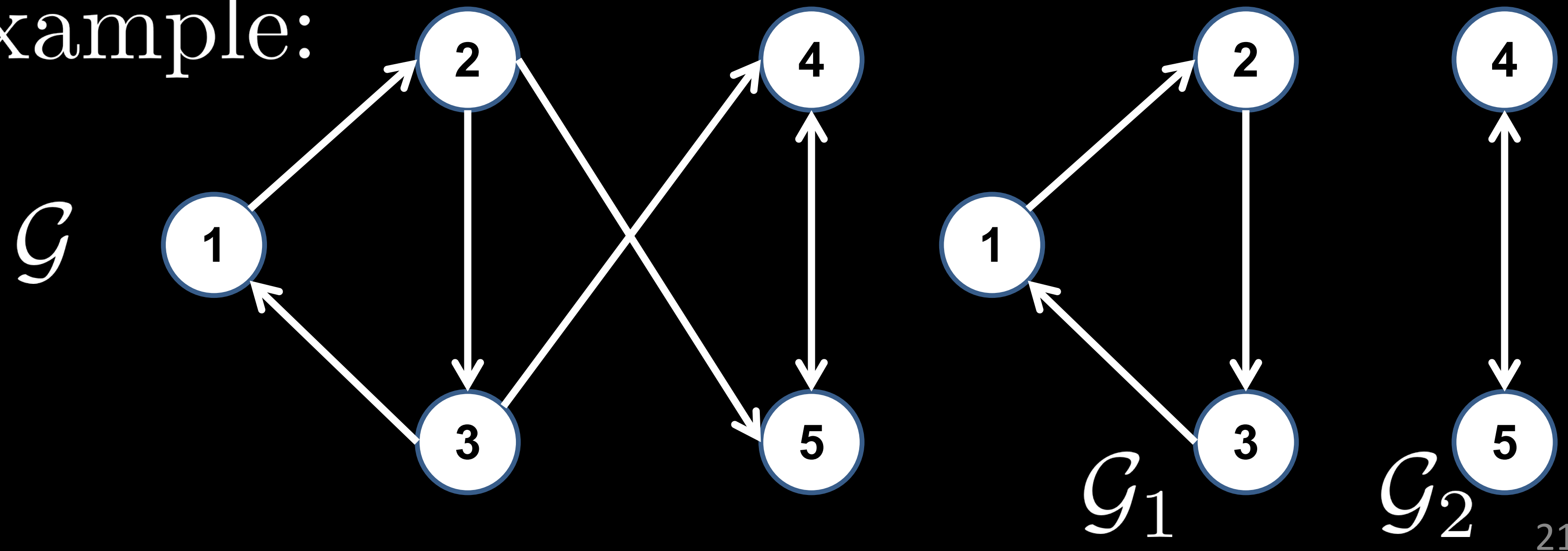
graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ may have multiple strong components

Let $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$, $\mathcal{G}_2 = (\mathcal{V}_2, \mathcal{E}_2)$ be two strong components of $\mathcal{G}$

either $\mathcal{V}_1 = \mathcal{V}_2, \mathcal{E}_1 = \mathcal{E}_2$

or $\mathcal{V}_1 \cap \mathcal{V}_2 = \emptyset, \mathcal{E}_1 \cap \mathcal{E}_2 = \emptyset$

example:

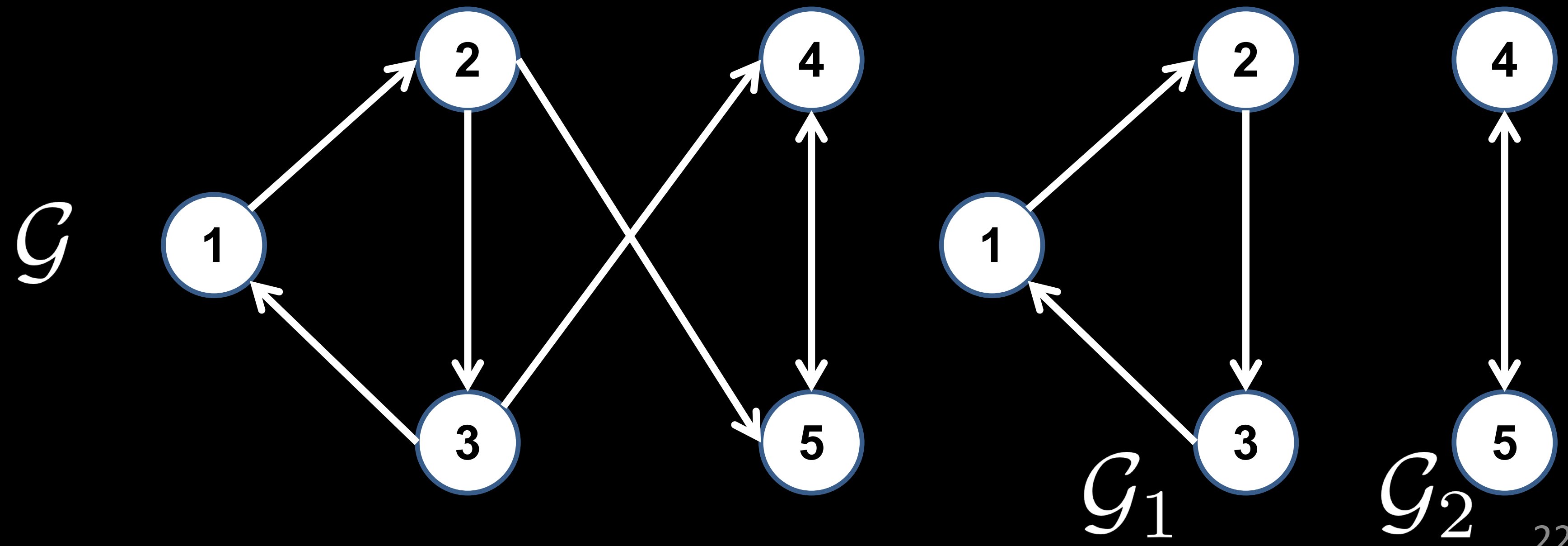


Closed strong component

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
strong component $\mathcal{G}' = (\mathcal{V}', \mathcal{E}')$

$\mathcal{G}'$ is closed if every node in $\mathcal{V}'$ is *not* reachable from any node in $\mathcal{V} \setminus \mathcal{V}'$

example:

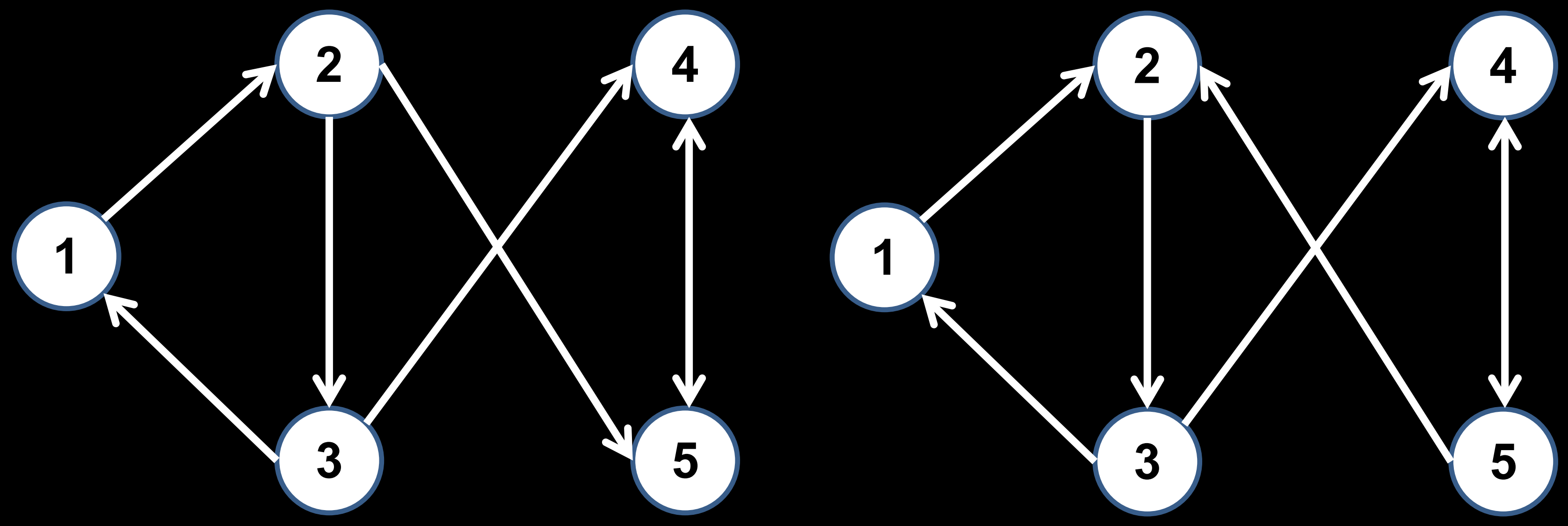


Fact

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

$\mathcal{G}$ contains a spanning tree iff $\mathcal{G}$ contains a unique closed strong component

example:



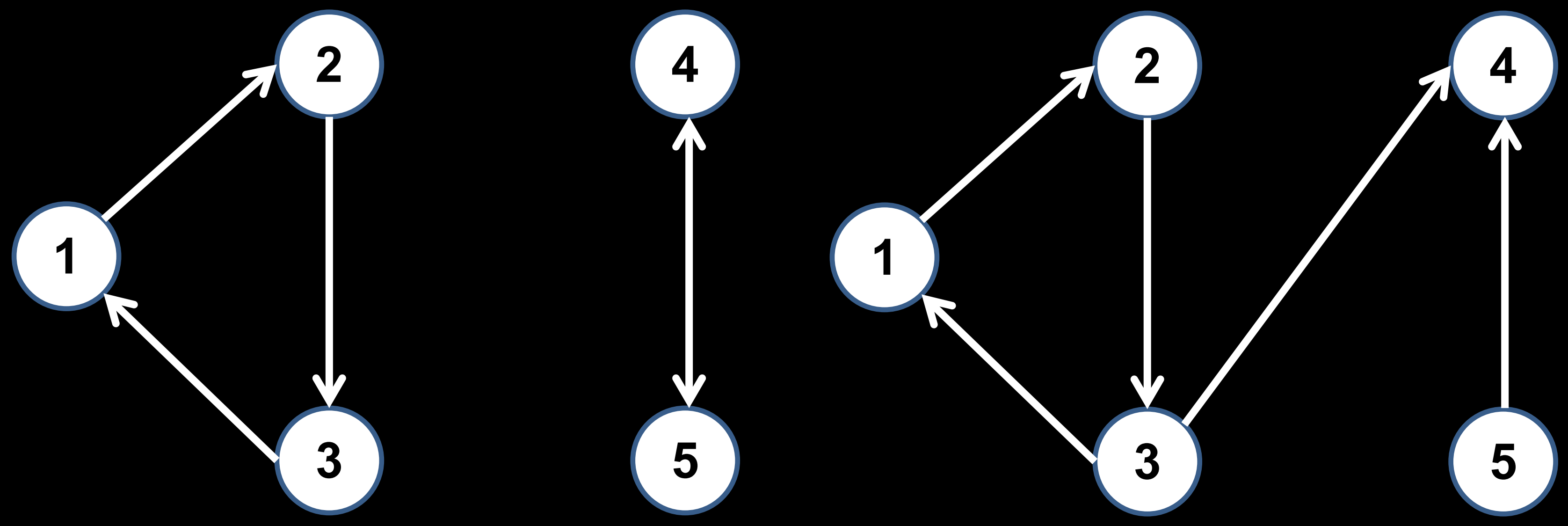
$\mathcal{G}_1$ $\mathcal{G}_2$

Fact

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

$\mathcal{G}$ contains a spanning tree iff $\mathcal{G}$ contains a unique closed strong component

example:



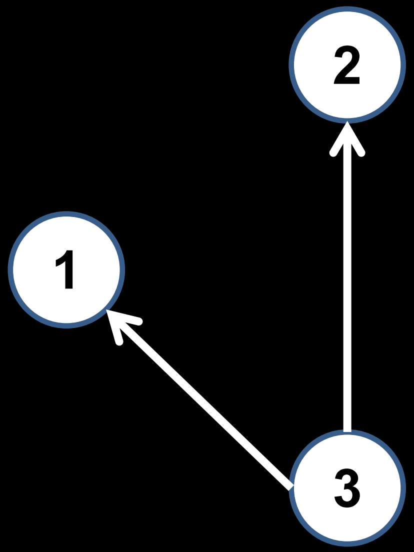
$\mathcal{G}_3$ $\mathcal{G}_4$

Fact

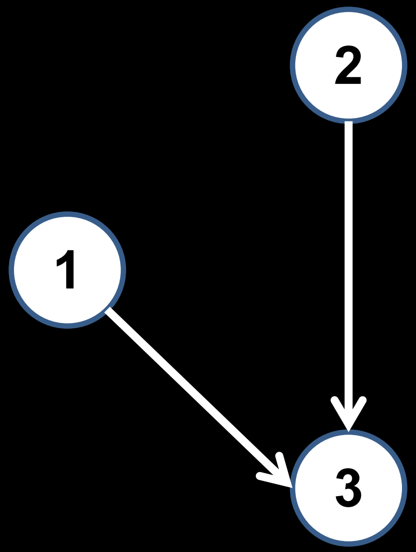
graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

$\mathcal{G}$ contains a spanning tree iff $\mathcal{G}$ contains a unique closed strong component

example:



$\mathcal{G}_5$



$\mathcal{G}_6$

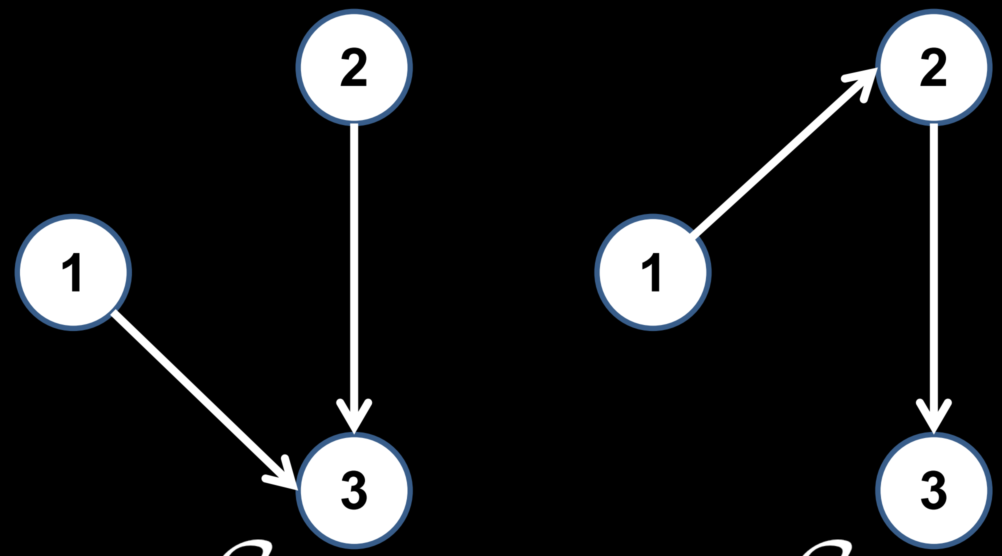
2-reachability

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

$\mathcal{R} = \{r_1, r_2\} \subseteq \mathcal{V}$ is a subset of 2 nodes

node $v \in \mathcal{V} \setminus \mathcal{R}$ is 2-reachable from $\mathcal{R}$ if v is still reachable from a node in $\mathcal{R}$ after removing an arbitrary node (not v)

example:



$\mathcal{G}_1$ $\mathcal{G}_2$

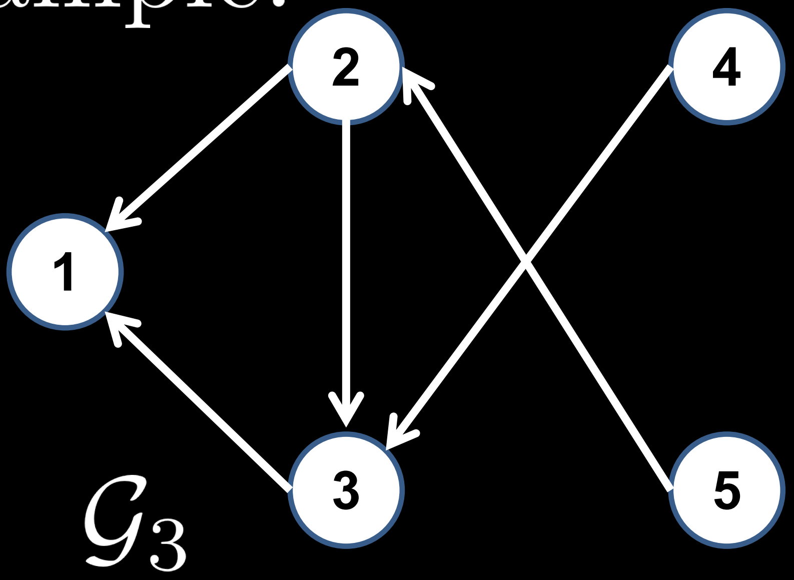
2-reachability

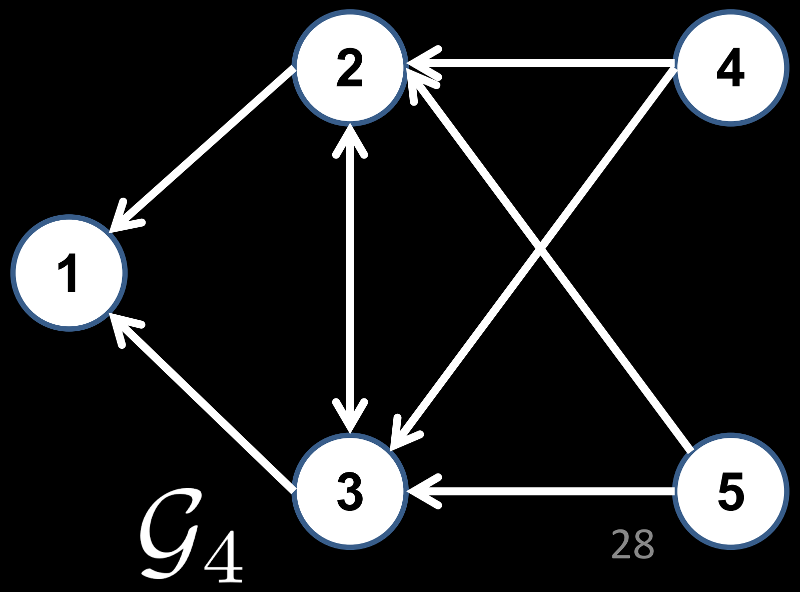
graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

$\mathcal{R} = \{r_1, r_2\} \subseteq \mathcal{V}$ is a subset of 2 nodes

node $v \in \mathcal{V} \setminus \mathcal{R}$ is 2-reachable from $\mathcal{R}$ if v is still reachable from a node in $\mathcal{R}$ after removing an arbitrary node (not v)

example:





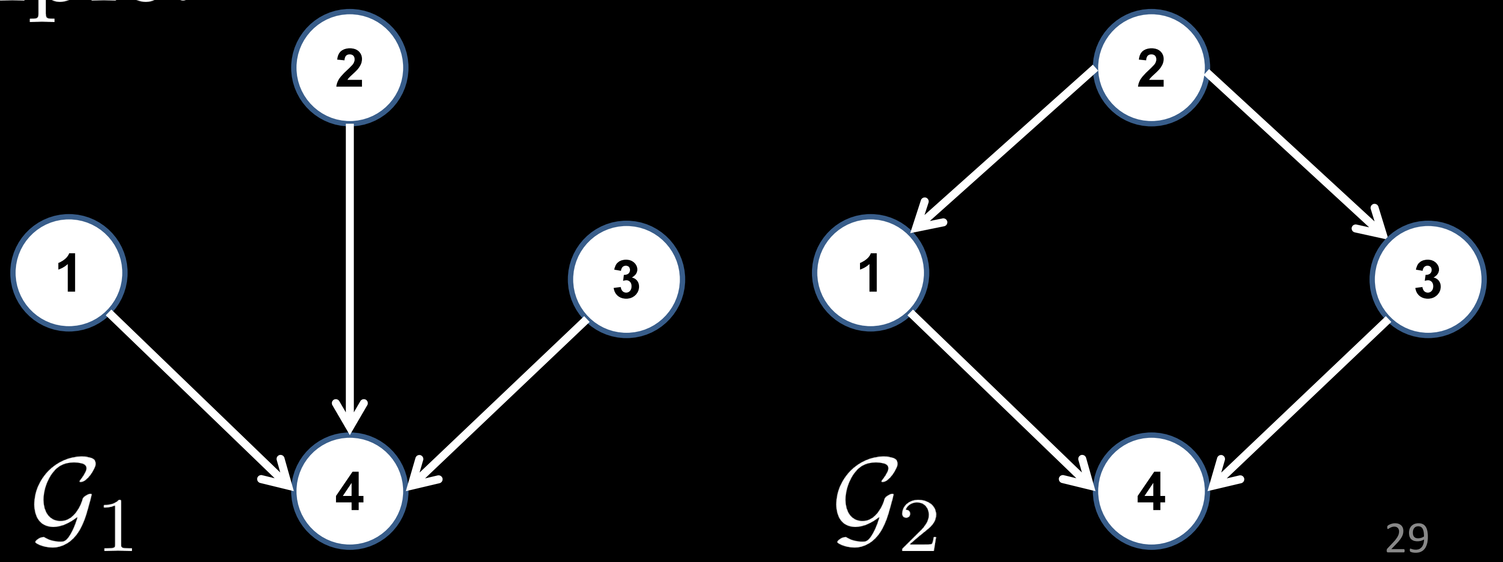
k-reachability

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

$\mathcal{R} \subseteq \mathcal{V}$ is a subset of k nodes $(k \geq 2)$

node $v \in \mathcal{V} \setminus \mathcal{R}$ is k-reachable from $\mathcal{R}$ if v is still reachable from a node in $\mathcal{R}$ after removing arbitrary $k-1$ nodes

example:



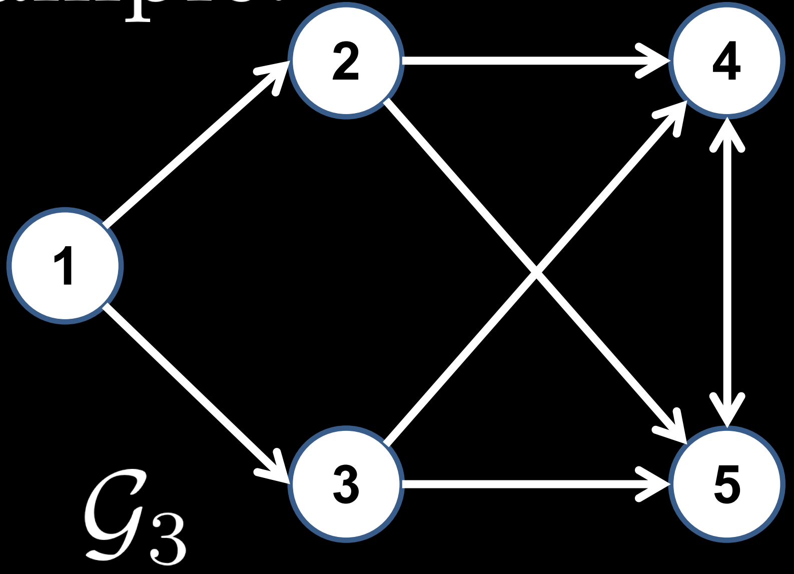
k-reachability

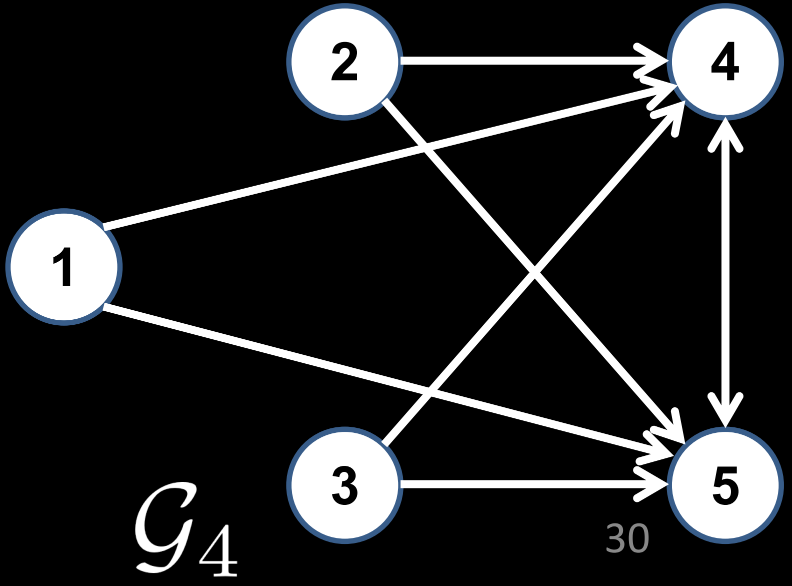
graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

$\mathcal{R} \subseteq \mathcal{V}$ is a subset of k nodes $(k \geq 2)$

node $v \in \mathcal{V} \setminus \mathcal{R}$ is k-reachable from $\mathcal{R}$ if v is still reachable from a node in $\mathcal{R}$ after removing arbitrary $k-1$ nodes

example:



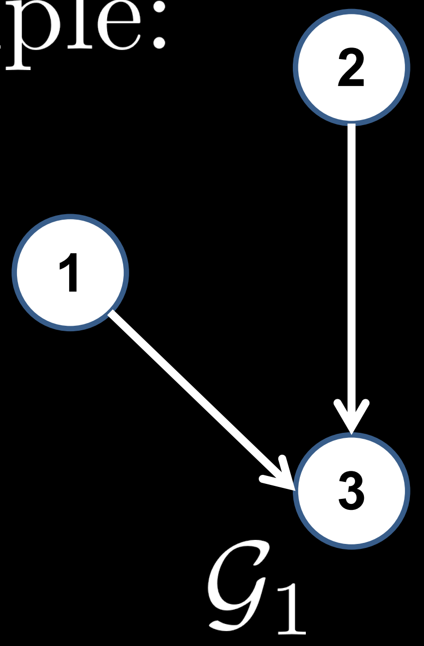


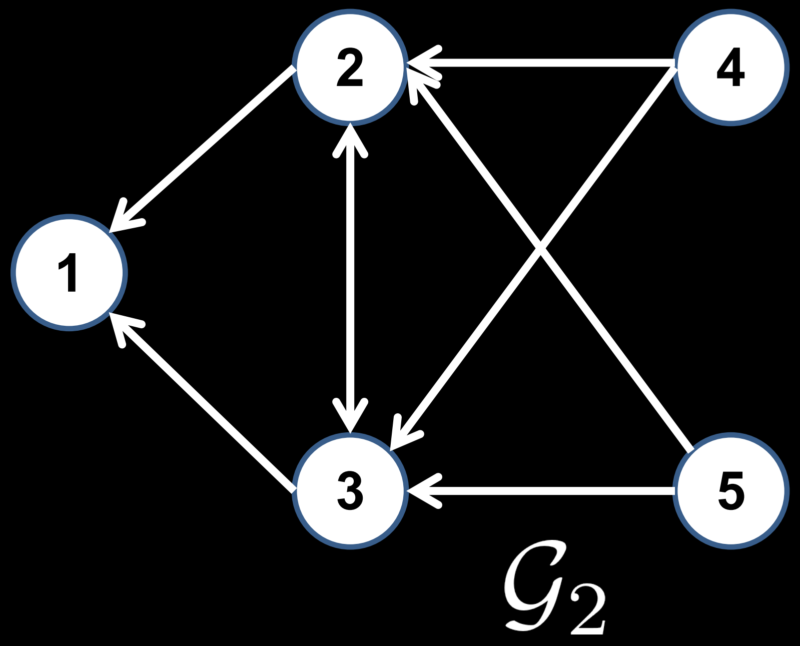
2-root set

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

$\mathcal{R} = \{r_1, r_2\} \subseteq \mathcal{V}$ is a 2-root set if every node $v \in \mathcal{V} \setminus \mathcal{R}$ is 2-reachable from $\mathcal{R}$

example:



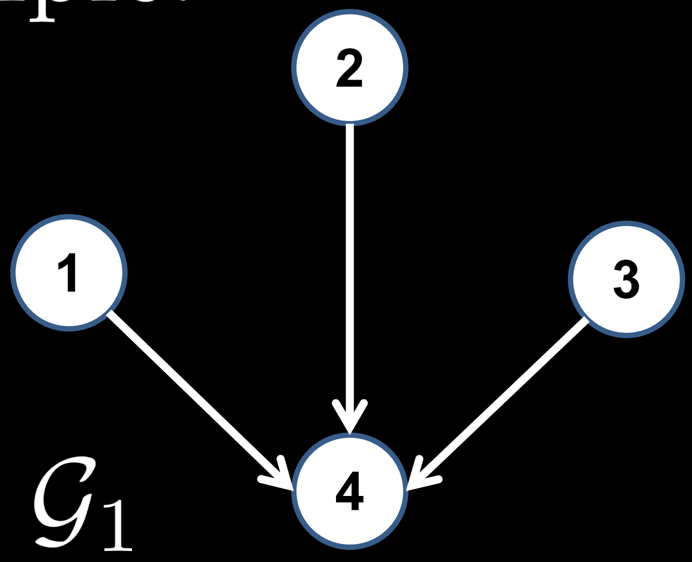


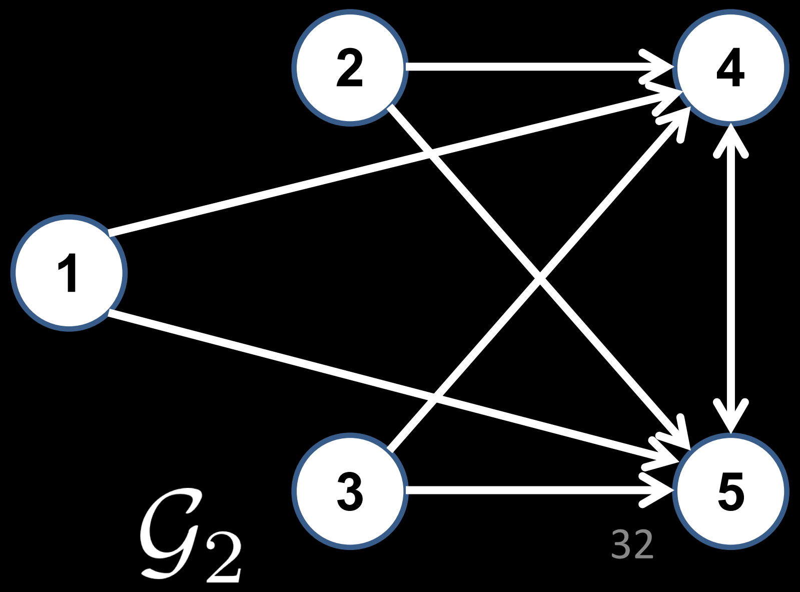
k-root set

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

$\mathcal{R} = \{r_1, \ldots, r_k\} \subseteq \mathcal{V}$ is a k-root set if every node $v \in \mathcal{V} \setminus \mathcal{R}$ is k-reachable from $\mathcal{R}$

example:

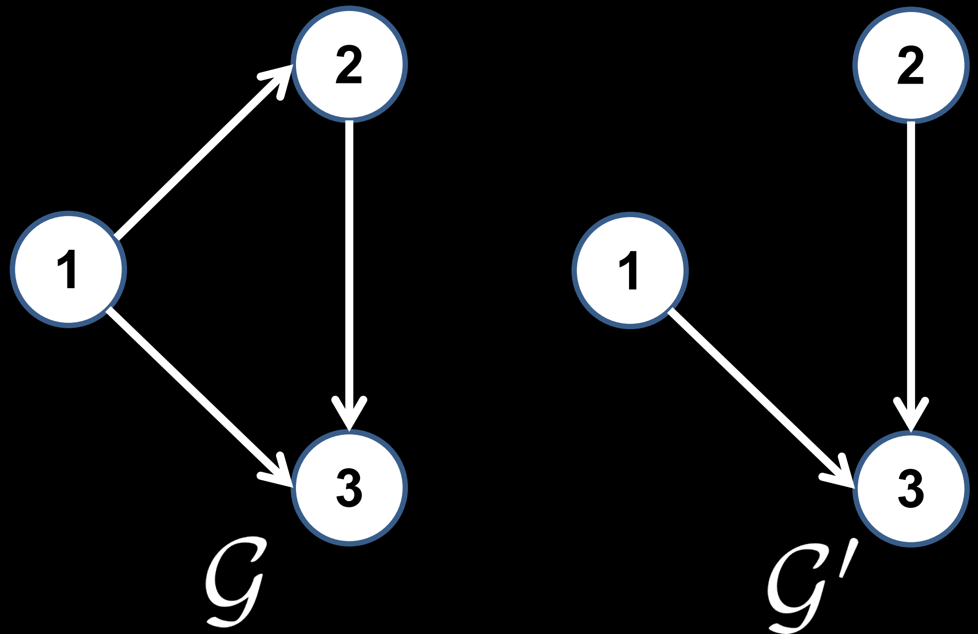




Spanning 2-tree

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and 2-root set $\mathcal{R} \subseteq \mathcal{V}$
a spanning subgraph $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$ is
a spanning 2-tree with $\mathcal{R}$ if
1) every $r_i \in \mathcal{R}$ has no neighbor
2) every $v \in \mathcal{V} \setminus \mathcal{R}$ has 2 neighbors

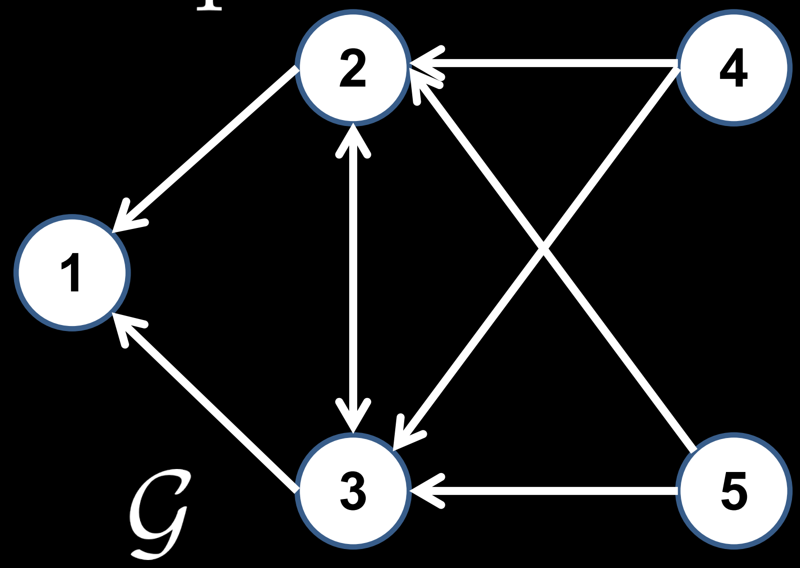
example:

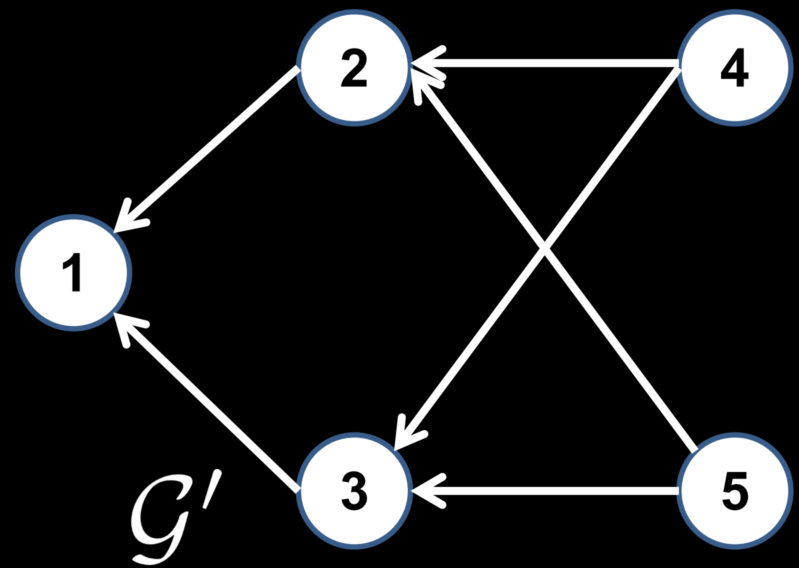


Spanning 2-tree

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and 2-root set $\mathcal{R} \subseteq \mathcal{V}$
a spanning subgraph $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$ is
a spanning 2-tree with $\mathcal{R}$ if
1) every $r_i \in \mathcal{R}$ has no neighbor
2) every $v \in \mathcal{V} \setminus \mathcal{R}$ has 2 neighbors

example:

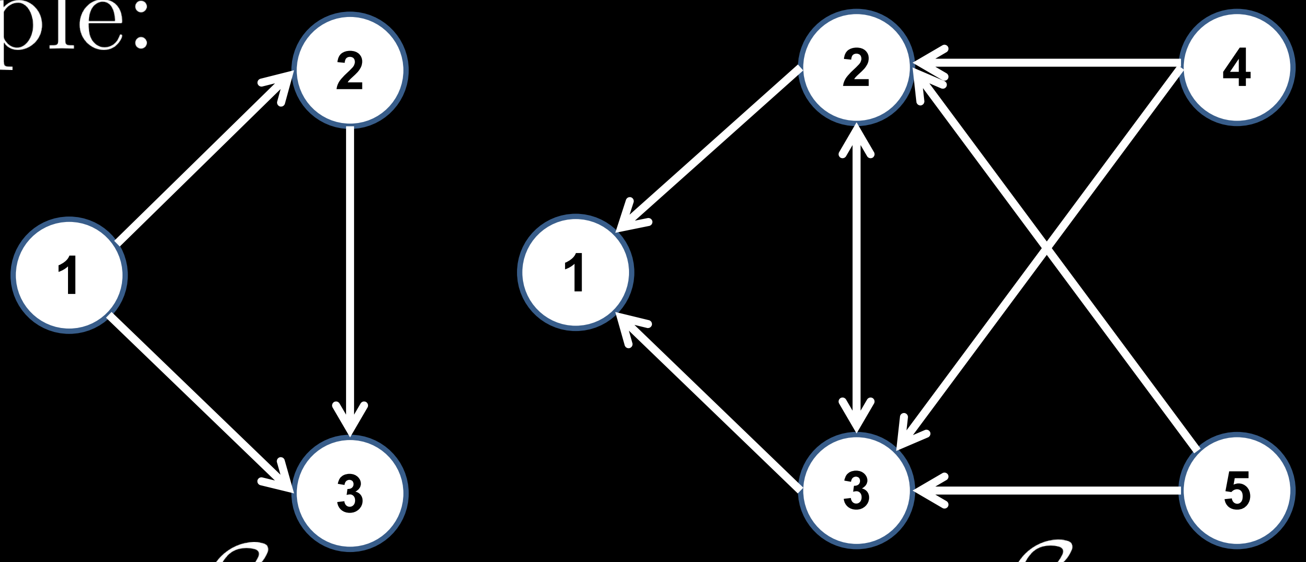




Spanning 2-tree

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ contains a spanning 2-tree if there exists a subdigraph of $\mathcal{G}$ that is a spanning 2-tree

example:



$\mathcal{G}_1$ $\mathcal{G}_2$

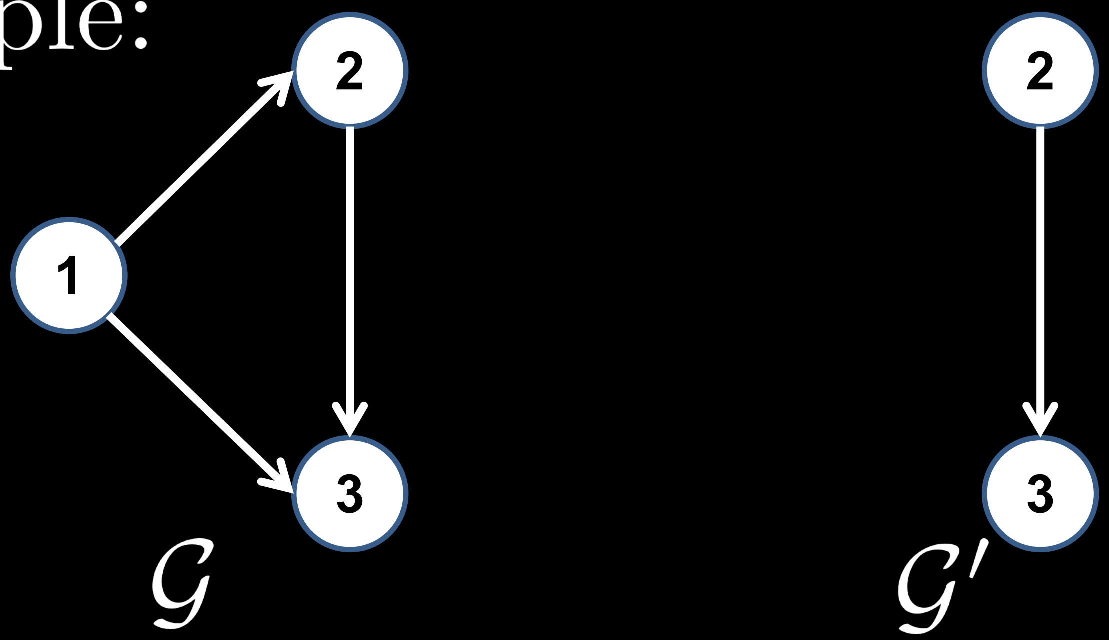
Spanning 2-tree

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ contains a spanning with 2-root set $\mathcal{R} = \{r_1, r_2\}$

Remove one node $r_i \in \mathcal{R}$ from $\mathcal{G}$

induced subgraph of $\mathcal{G}$ by $\mathcal{V} \setminus \{r_i\}$ contains a spanning tree

example:



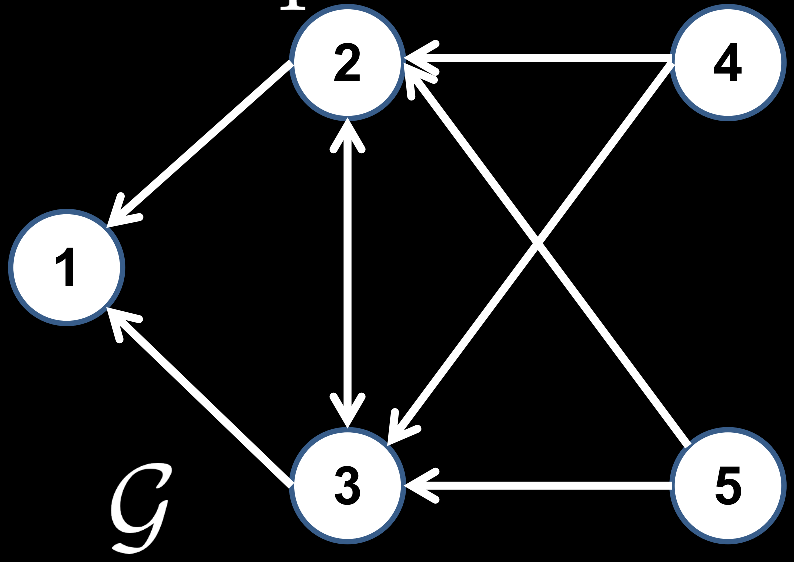
Spanning 2-tree

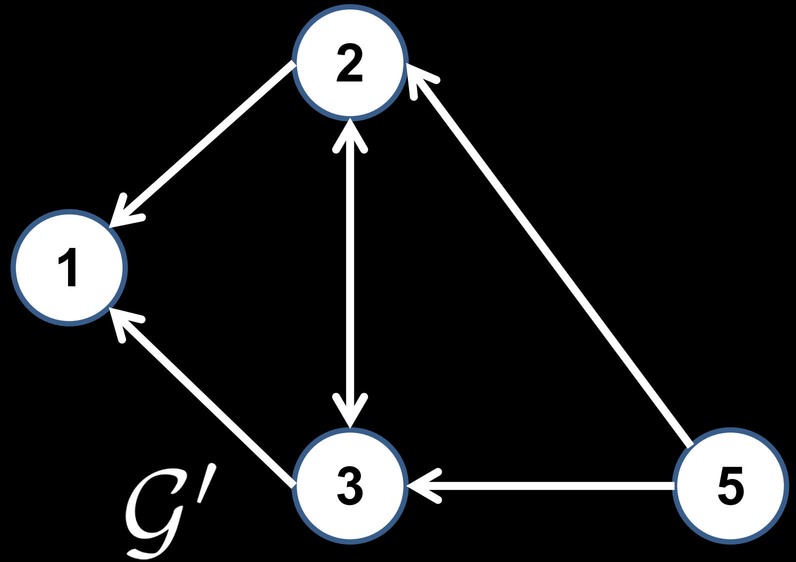
graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ contains a spanning with 2-root set $\mathcal{R} = \{r_1, r_2\}$

Remove one node $r_i \in \mathcal{R}$ from $\mathcal{G}$

induced subgraph of $\mathcal{G}$ by $\mathcal{V} \setminus \{r_i\}$ contains a spanning tree

example:

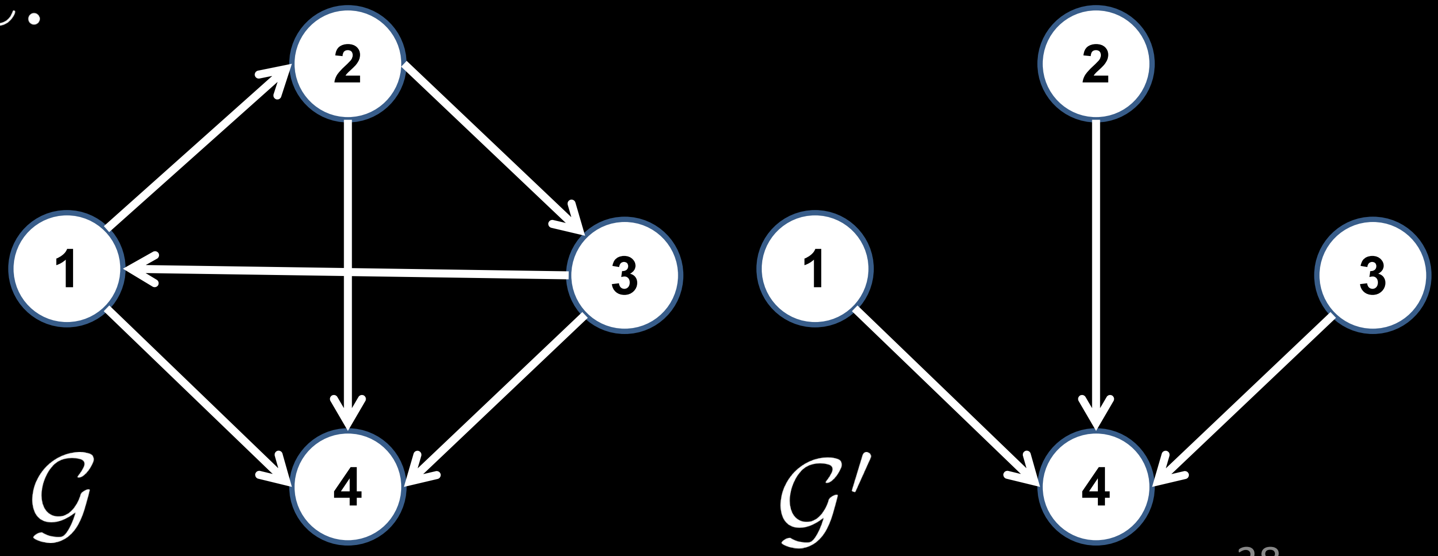




Spanning *k*-tree

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and k-root set $\mathcal{R} \subseteq \mathcal{V}$
a spanning subgraph $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$ is
a spanning k-tree with $\mathcal{R}$ if
1) every $r_i \in \mathcal{R}$ has no neighbor
2) every $v \in \mathcal{V} \setminus \mathcal{R}$ has k neighbors

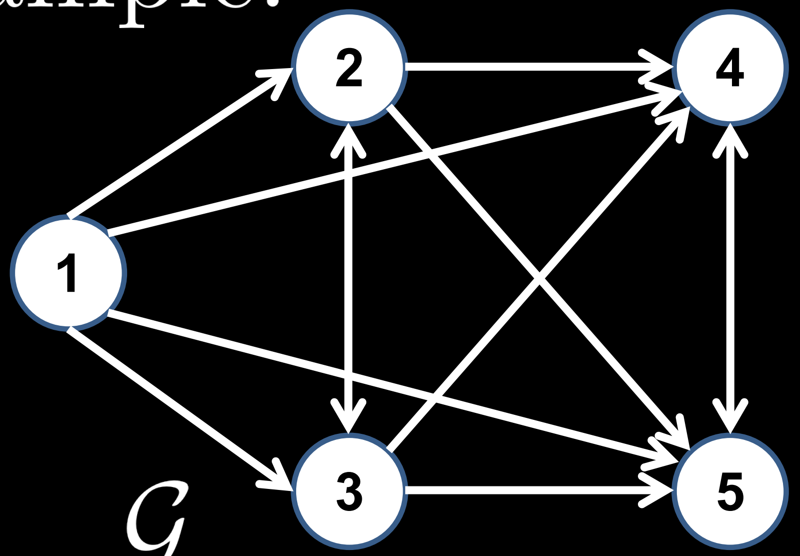
example:



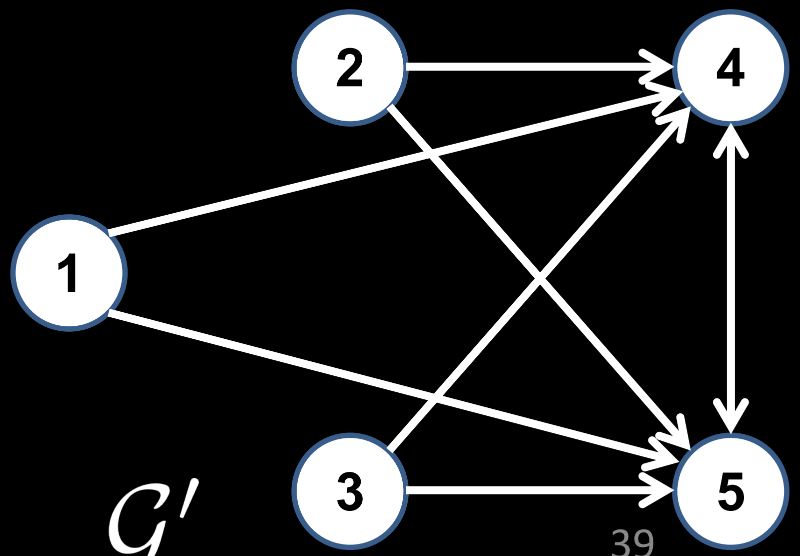
Spanning k-tree

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and k-root set $\mathcal{R} \subseteq \mathcal{V}$
a spanning subgraph $\mathcal{G}' = (\mathcal{V}, \mathcal{E}')$ is
a spanning k-tree with $\mathcal{R}$ if
1) every $r_i \in \mathcal{R}$ has no neighbor
2) every $v \in \mathcal{V} \setminus \mathcal{R}$ has k neighbors

example:



$\mathcal{G}$

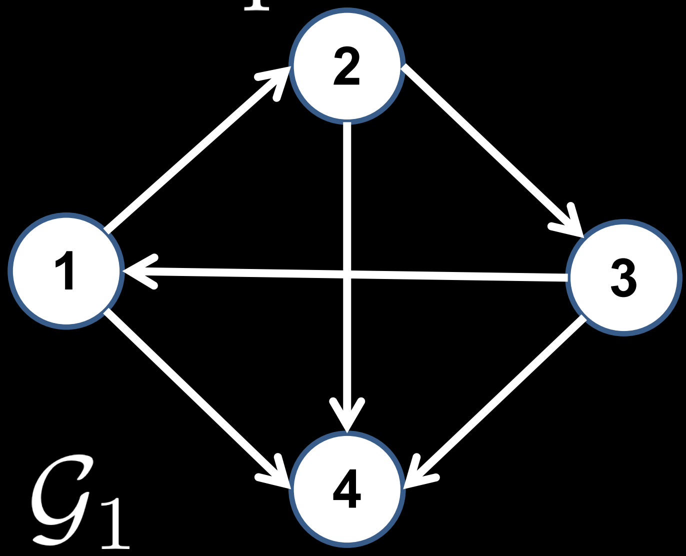


$\mathcal{G}'$

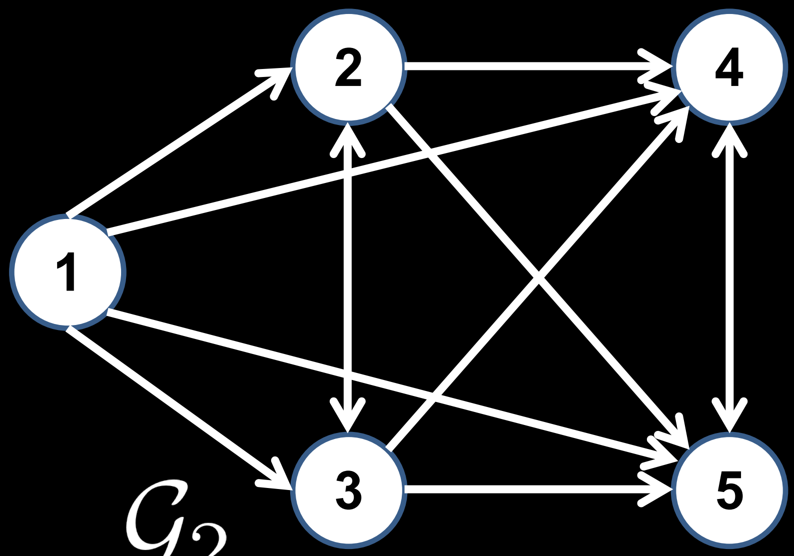
Spanning k-tree

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ contains a spanning k-tree if there exists a subdigraph of $\mathcal{G}$ that is a spanning k-tree

example:



$\mathcal{G}_1$

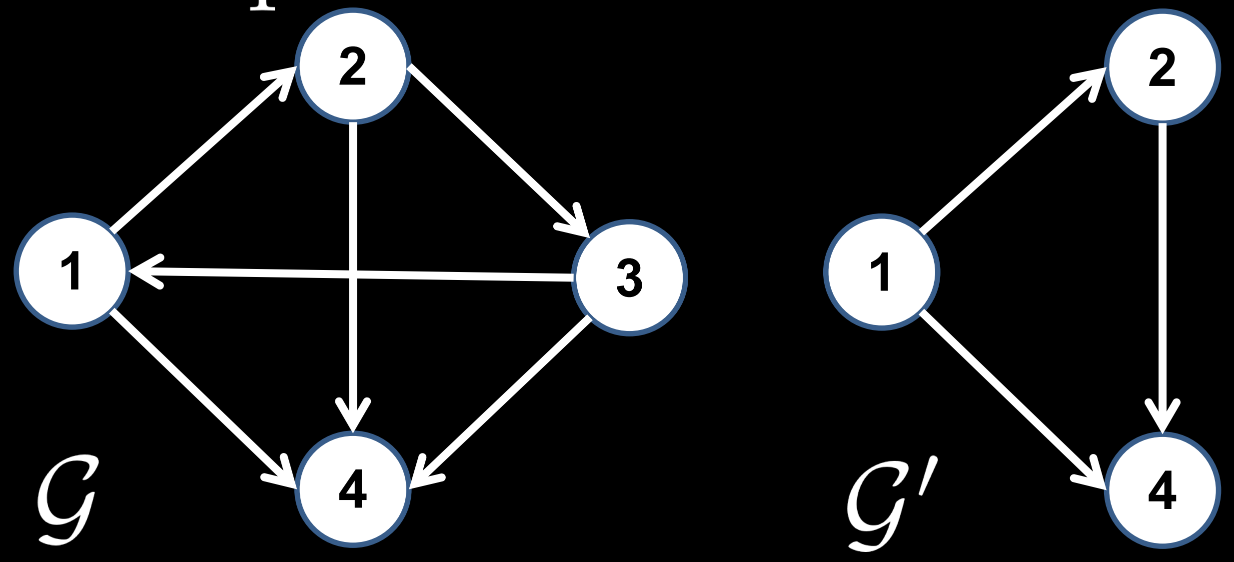


$\mathcal{G}_2$

Spanning *k*-tree

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ contains a spanning with k-root set $\mathcal{R} = \{r_1, \ldots, r_k\}$
Remove one node $r_i \in \mathcal{R}$ from $\mathcal{G}$
induced subgraph of $\mathcal{G}$ by $\mathcal{V} \setminus \{r_i\}$
contains a spanning $(k-1)$-tree

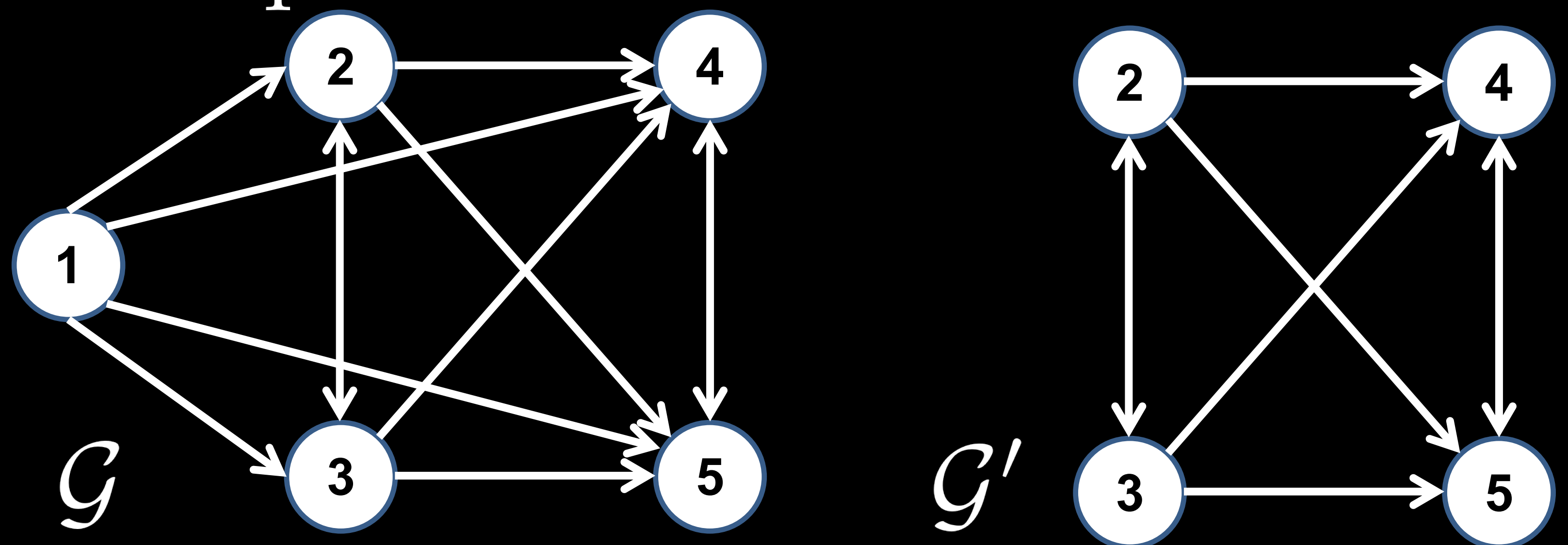
example:



Spanning *k*-tree

graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ contains a spanning with k-root set $\mathcal{R} = \{r_1, \ldots, r_k\}$
Remove one node $r_i \in \mathcal{R}$ from $\mathcal{G}$
induced subgraph of $\mathcal{G}$ by $\mathcal{V} \setminus \{r_i\}$ contains a spanning $(k-1)$-tree

example:



Graph theory: matrices

Weighted graph

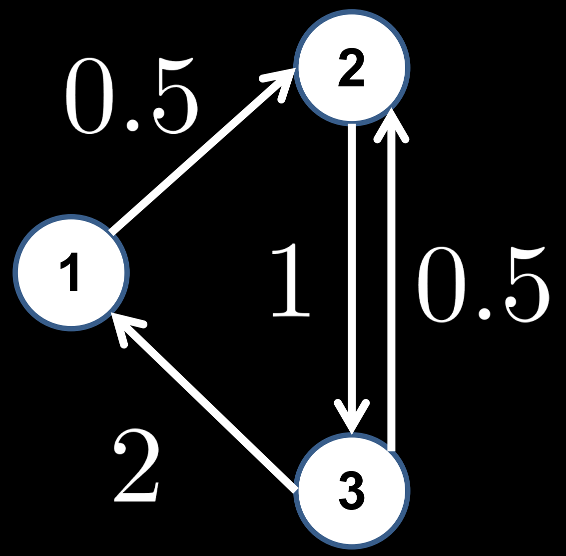
graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

node set $\mathcal{V} = \{v_1, \ldots, v_n\}$

edge set $\mathcal{E} = \{(v_i, v_j), \ldots\}$

edge (v_i, v_j) has weight a_{ji}

example:



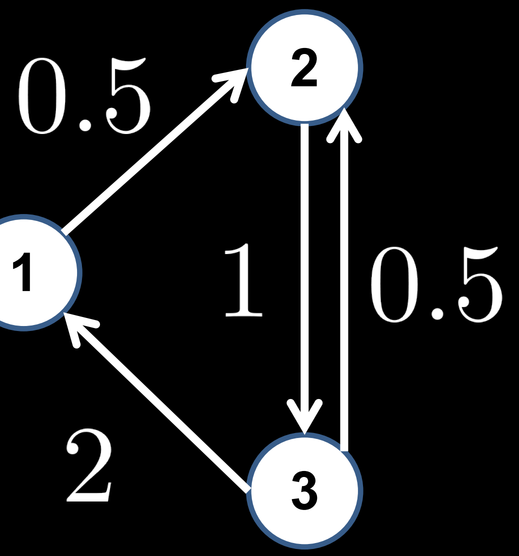
Weighted graph

weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

convention:

- $(\forall i \in [1, n])(v_i, v_i) \notin \mathcal{V}$
- weight $a_{ji} = 0$ iff edge $(v_i, v_j) \notin \mathcal{V}$

example:



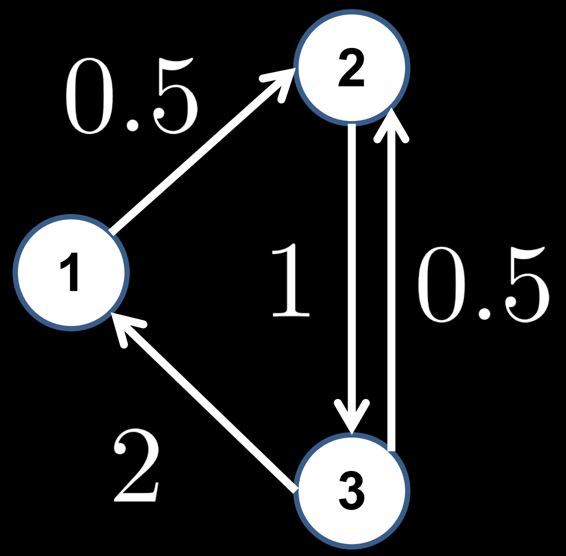
Weighted degree

weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

edge (v_i, v_j) has weight a_{ji}

weighted degree of v_j is $d_{v_j} = \sum_{i \in \mathcal{N}_j} a_{ji}$

example:



Weighted degree

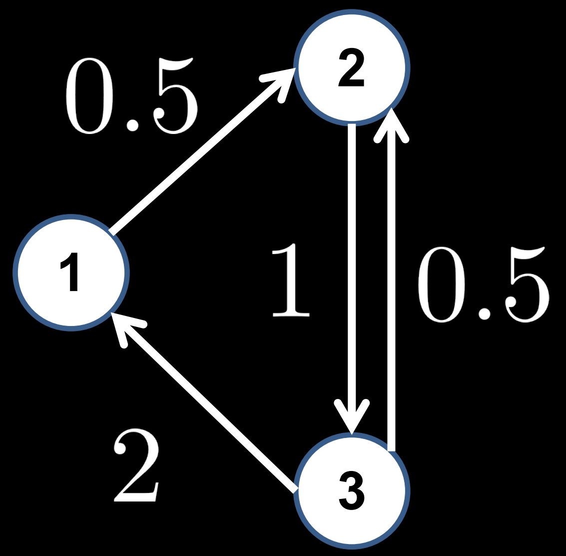
weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

edge (v_i, v_j) has weight a_{ji}

weighted out-degree of v_j is

$$d^o_{v_j} = \sum_{i \in \mathcal{N}^o_j} a_{ij}$$

example:



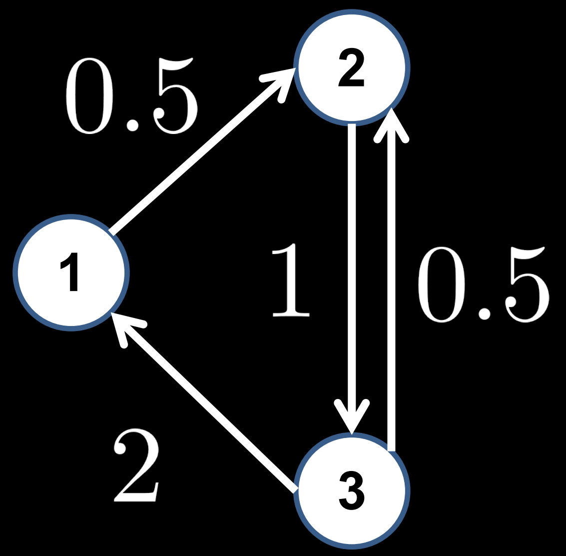
Balanced weighted graph

weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

edge (v_i, v_j) has weight a_{ji}

node v_j is weight-balanced if $d_{v_j} = d^o_{v_j}$

example:



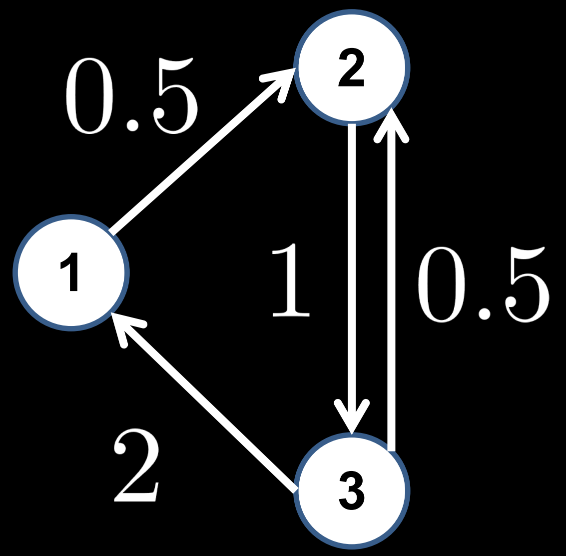
Balanced weighted graph

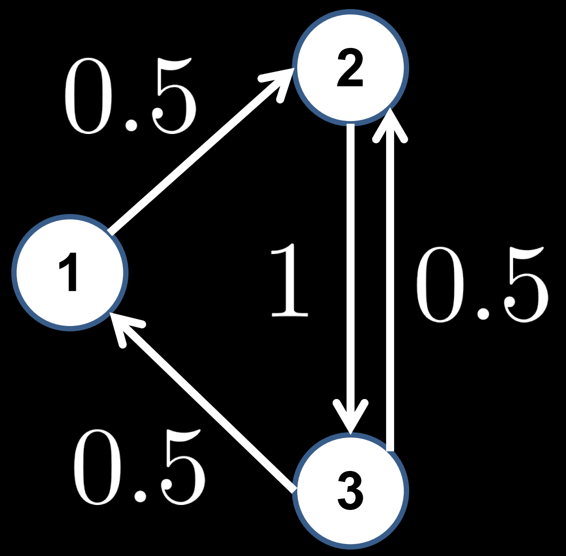
weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

edge (v_i, v_j) has weight a_{ji}

$\mathcal{G}$ is weight-balanced if
every v is weight-balanced

example:





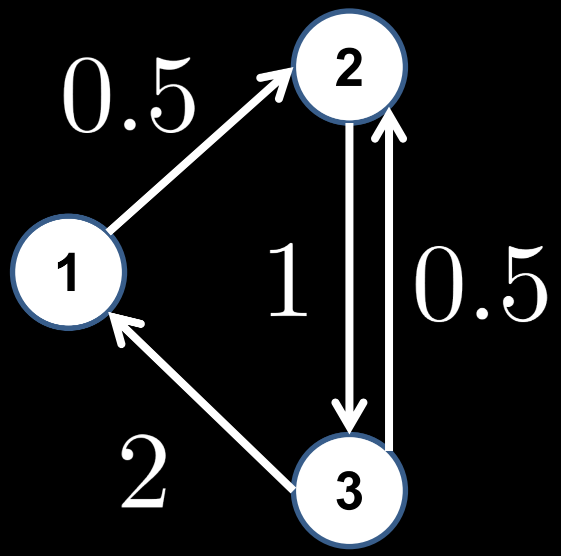
Adjacency matrix

weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = n$

edge (v_i, v_j) has weight a_{ji}

adjacency matrix $A = [a_{ij}]$

example:



Degree matrix

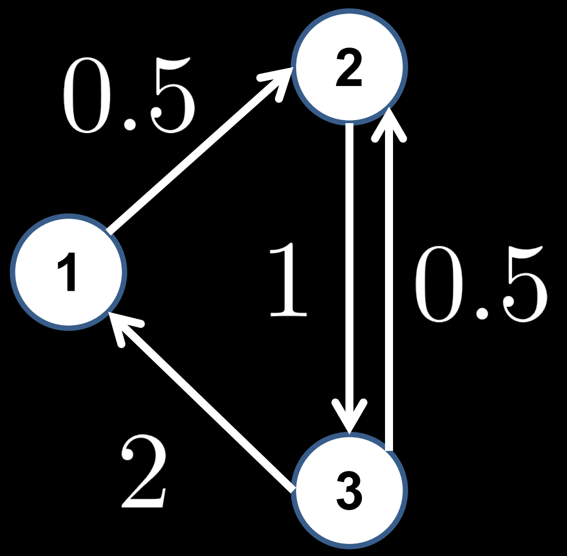
weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = n$

edge (v_i, v_j) has weight a_{ji}

degree matrix $D = \text{diag}(d_{v_1}, \ldots, d_{v_n})$

('diag' means diagonalization)

example:



Adjacency & degree matrix

$$\underset{A}{\begin{bmatrix} 0 & 0 & 2 \\ 0.5 & 0 & 0.5 \\ 0 & 1 & 0 \end{bmatrix}} \qquad \underset{D}{\begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}$$

$$\operatorname{diag}(A\mathbf{1}) = D,\ \mathbf{1} = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}$$

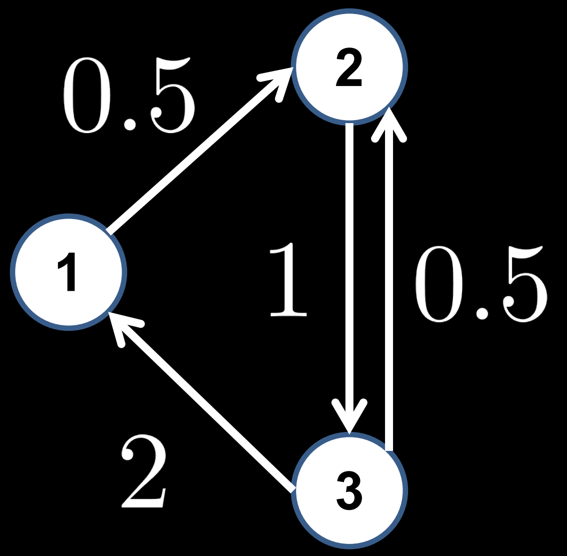
Laplacian matrix

weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = n$

edge (v_i, v_j) has weight a_{ji}

Laplacian matrix $L = D - A$

example:



Laplacian matrix

$$L = \begin{bmatrix} 2 & 0 & -2 \\ -0.5 & 1 & -0.5 \\ 0 & -1 & 1 \end{bmatrix}$$

Every row sums up to zero

$$L\mathbf{1} = (D - A)\mathbf{1}$$

Eigenvalue & eigenvector

weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = n$

Laplacian matrix $L = D - A$

L has an eigenvalue 0,
with eigenvector $\mathbf{1}$ (?)

Rank

weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = n$

Laplacian matrix $L = D - A$

$\text{rank}(L) \leq n - 1$

if $\mathcal{G}$ contains a spanning tree

$$\text{rank}(L) \geq n - 1$$

if $\mathcal{G}$ contains a spanning 2-tree

$$\text{rank}(L) \geq n - 2$$

if $\mathcal{G}$ contains a spanning k-tree

$$\text{rank}(L) \geq n - k$$

Rank vs. spanning tree

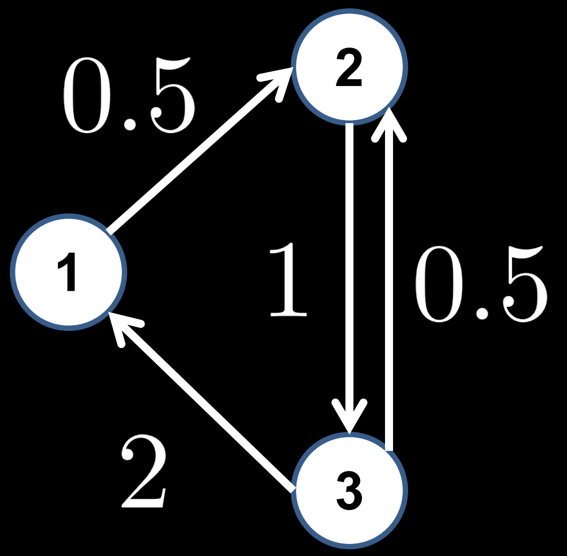
weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = n$

Laplacian matrix $L = D - A$

if $\mathcal{G}$ contains a spanning tree

$$\text{rank}(L) = n - 1$$

example:



$$L = \begin{bmatrix} 2 & 0 & -2 \\ -0.5 & 1 & -0.5 \\ 0 & -1 & 1 \end{bmatrix}$$

Rank vs. spanning 2-tree

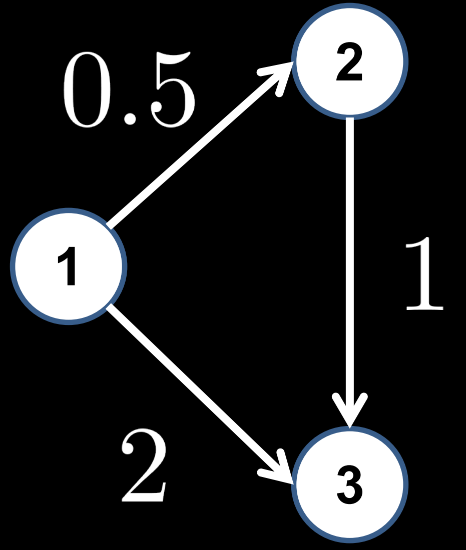
weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = n$

Laplacian matrix $L = D - A$

if $\mathcal{G}$ contains a spanning 2-tree

$$n - 2 \leq \text{rank}(L) \leq n - 1$$

example:



$$L = \begin{bmatrix} 0 & 0 & 0 \\ -0.5 & 0.5 & 0 \\ -2 & -1 & 3 \end{bmatrix}$$

Rank vs. spanning 2-tree

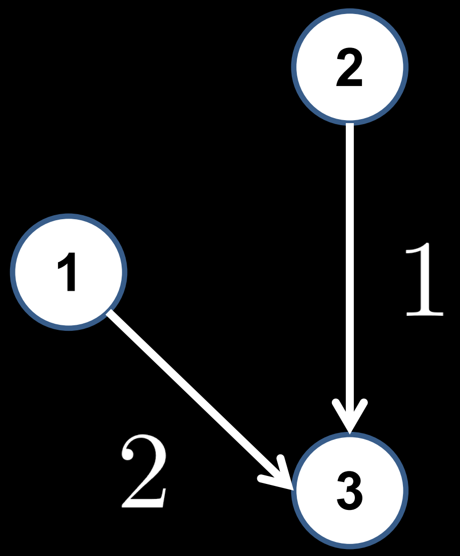
weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = n$

Laplacian matrix $L = D - A$

if $\mathcal{G}$ contains a spanning 2-tree

$$n - 2 \leq \mathrm{rank}(L) \leq n - 1$$

example:



$$L = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -2 & -1 & 3 \end{bmatrix}$$

Rank vs. spanning *k*-tree

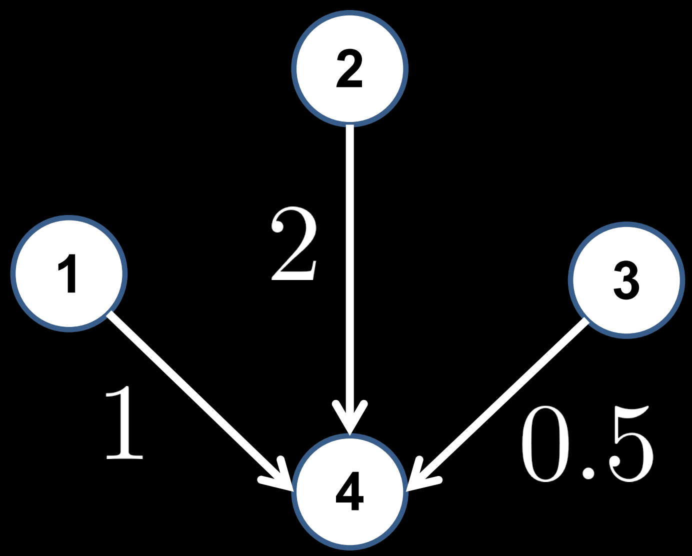
weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = n$

Laplacian matrix $L = D - A$

if $\mathcal{G}$ contains a spanning k-tree

$$n - k \leq \text{rank}(L) \leq n - 1$$

example:



$$L = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & -2 & -0.5 & 3.5 \end{bmatrix}$$

Rank vs. spanning *k*-tree

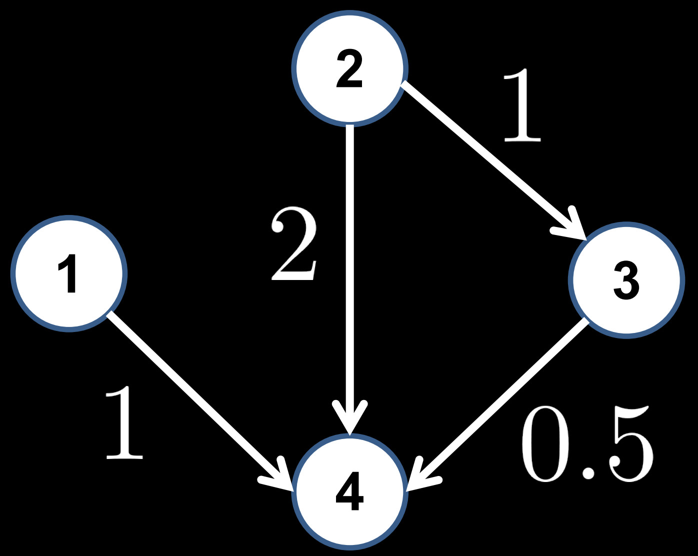
weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = n$

Laplacian matrix $L = D - A$

if $\mathcal{G}$ contains a spanning k-tree

$$n - k \leq \text{rank}(L) \leq n - 1$$

example:



$$L = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ -1 & -2 & -0.5 & 3.5 \end{bmatrix}$$

Rank vs. spanning *k*-tree

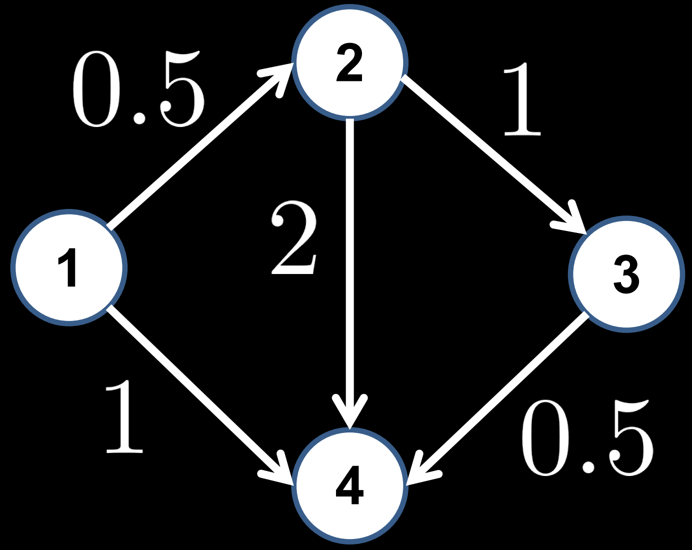
weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = n$

Laplacian matrix $L = D - A$

if $\mathcal{G}$ contains a spanning k-tree

$$n - k \leq \text{rank}(L) \leq n - 1$$

example:



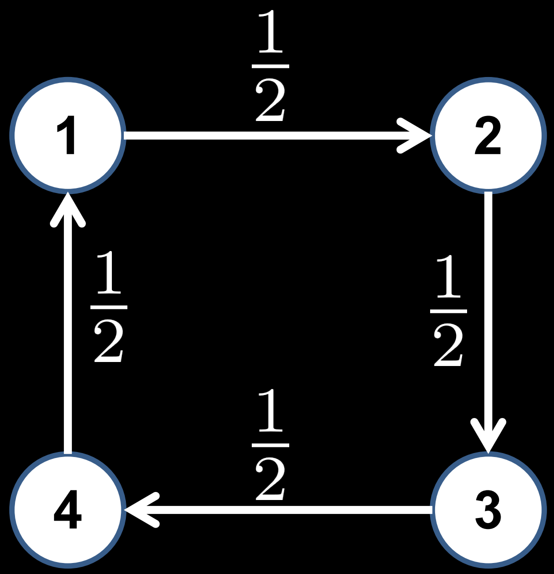
$$L = \begin{bmatrix} 0 & 0 & 0 & 0 \\ -0.5 & 0.5 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ -1 & -2 & -0.5 & 3.5 \end{bmatrix}$$

Nonnegative matrix

matrix A is nonnegative, $A \geq 0$

if every entry $a_{ij} \geq 0$

example:



adjacency matrix

$$A = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \end{bmatrix} \geq 0$$

Positive matrix

matrix A is positive, $A > 0$

if every entry $a_{ij} > 0$

example:

$$A = \begin{bmatrix} 1 & 1 & 2 & \frac{1}{2} \\ \frac{1}{2} & 2 & 1 & 1 \\ 1 & \frac{1}{2} & 2 & 2 \\ 3 & \frac{1}{5} & \frac{1}{2} & 1 \end{bmatrix} > 0$$

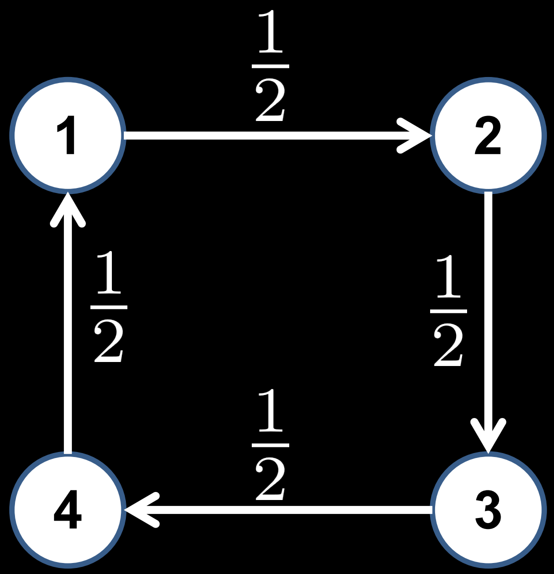
Irreducible matrix

$A \geq 0$ is an irreducible matrix

if $I + A + \cdots + A^{n-1} > 0$

(n is size of A)

example:



$$A = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \end{bmatrix} \geq 0$$

$$I + A + A^2 + A^3 =$$

$$\begin{bmatrix} \frac{15}{8} & \frac{1}{8} & \frac{5}{8} & \frac{11}{8} \\ \frac{11}{8} & \frac{15}{8} & \frac{1}{8} & \frac{5}{8} \\ \frac{5}{8} & \frac{11}{8} & \frac{15}{8} & \frac{1}{8} \\ \frac{1}{8} & \frac{5}{8} & \frac{11}{8} & \frac{15}{8} \end{bmatrix} > 0$$

Irreducible matrix

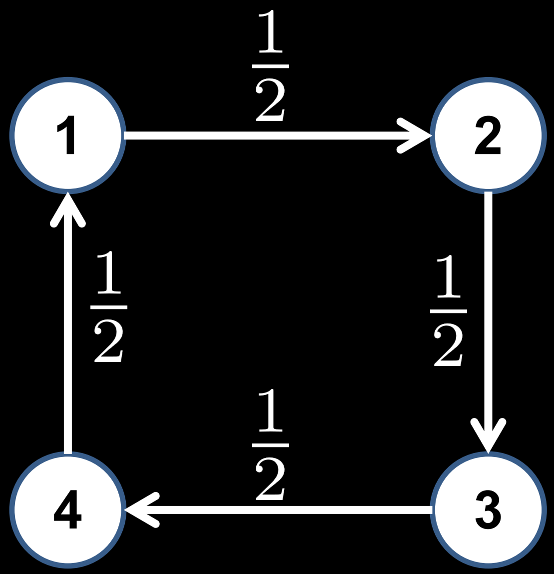
Fact: $A \geq 0$ is adjacency matrix of $\mathcal{G}$.

A is an irreducible matrix

iff $\mathcal{G}$ is strongly connected

example:

$$A = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \end{bmatrix} \geq 0$$



$$I + A + A^2 + A^3 =$$

$$\begin{bmatrix} \frac{15}{8} & \frac{1}{8} & \frac{5}{8} & \frac{11}{8} \\ \frac{11}{8} & \frac{15}{8} & \frac{1}{8} & \frac{5}{8} \\ \frac{5}{8} & \frac{11}{8} & \frac{15}{8} & \frac{1}{8} \\ \frac{1}{8} & \frac{5}{8} & \frac{11}{8} & \frac{15}{8} \end{bmatrix} > 0$$

Primitive matrix

$A \geq 0$ is a primitive matrix

if $(\exists k > 0) A^k > 0$

example:

$$A = \begin{bmatrix} \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \geq 0$$

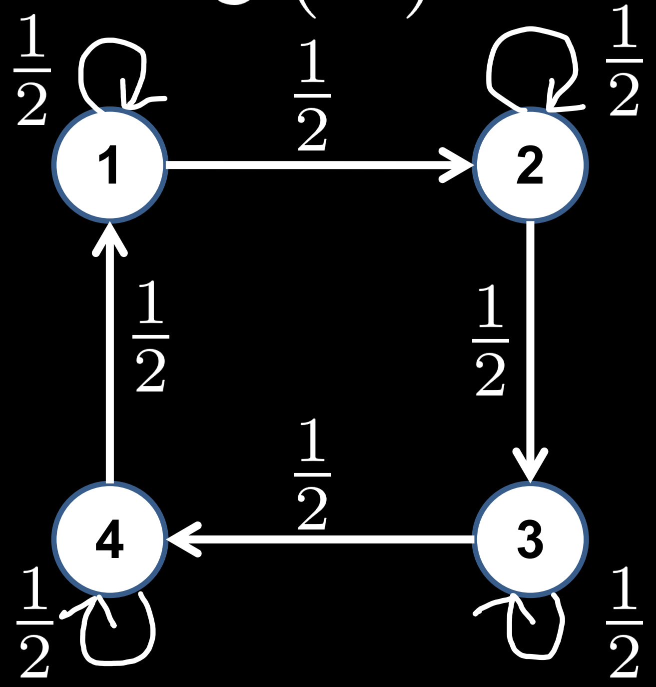
$$A^3 = \begin{bmatrix} \frac{1}{8} & \frac{1}{8} & \frac{3}{8} & \frac{3}{8} \\ \frac{3}{8} & \frac{1}{8} & \frac{1}{8} & \frac{3}{8} \\ \frac{3}{8} & \frac{3}{8} & \frac{1}{8} & \frac{1}{8} \\ \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{bmatrix} > 0$$

Primitive matrix

Fact: $A \geq 0$ is adjacency matrix of $\mathcal{G}$ (every node has a selfloop edge).
A is a primitive matrix
iff $\mathcal{G}$ is strongly connected

example:

$\mathcal{G}(A)$



$$A = \begin{bmatrix} \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \geq 0$$

$$A^3 = \begin{bmatrix} \frac{1}{8} & \frac{1}{8} & \frac{3}{8} & \frac{3}{8} \\ \frac{3}{8} & \frac{1}{8} & \frac{1}{8} & \frac{3}{8} \\ \frac{3}{8} & \frac{3}{8} & \frac{1}{8} & \frac{1}{8} \\ \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{bmatrix} > 0$$

Types of Laplacian matrix

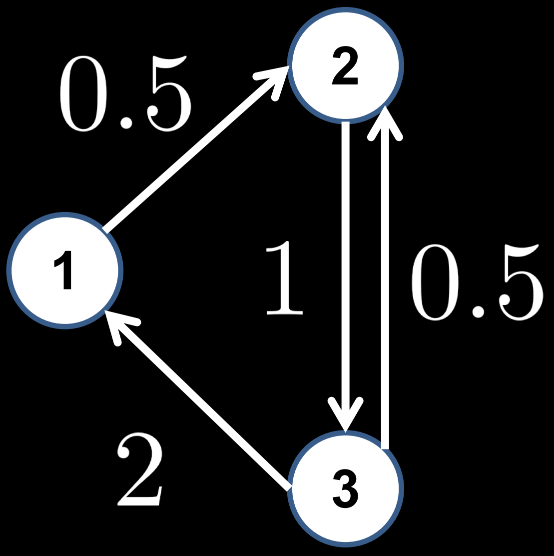
weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = n$

Laplacian matrix $L = D - A$

$a_{ij} \geq 0 \Rightarrow A$ nonnegative

$\Rightarrow L$ ordinary

example:



$$A = \begin{bmatrix} 0 & 0 & 2 \\ 0.5 & 0 & 0.5 \\ 0 & 1 & 0 \end{bmatrix}$$

$$L = \begin{bmatrix} 2 & 0 & -2 \\ -0.5 & 1 & -0.5 \\ 0 & -1 & 1 \end{bmatrix}$$

Types of Laplacian matrix

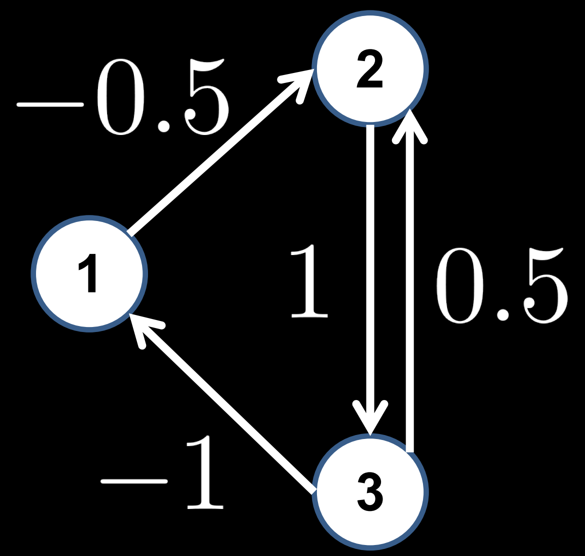
weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = n$

Laplacian matrix $L = D - A$

$a_{ij} \in \mathbb{R} \Rightarrow A$ real

$\Rightarrow L$ signed

example:



A

L

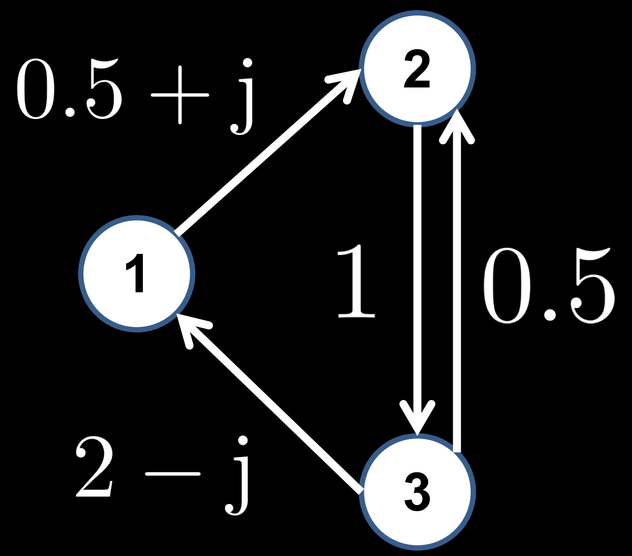
Types of Laplacian matrix

weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = n$

Laplacian matrix $L = D - A$

$$a_{ij} \in \mathbb{C} \Rightarrow A \text{ complex}$$
$$\Rightarrow L \text{ complex}$$

example:



$$A = \begin{bmatrix} 0 & 0 & 2-\mathrm{j} \\ 0.5+\mathrm{j} & 0 & 0.5 \\ 0 & 1 & 0 \end{bmatrix}$$

$$L = \begin{bmatrix} 2-\mathrm{j} & 0 & \mathrm{j}-2 \\ -0.5-\mathrm{j} & 1+\mathrm{j} & -0.5 \\ 0 & -1 & 1 \end{bmatrix}$$

Types of Laplacian matrix

weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $|\mathcal{V}| = n$

Laplacian matrix $L = D - A$

$a_{ij} \geq 0 \Rightarrow A$ nonnegative $\Rightarrow L$ ordinary

averaging, optimization, consensus

$a_{ij} \in \mathbb{C} \Rightarrow A$ complex $\Rightarrow L$ complex

2D formation control

$a_{ij} \in \mathbb{R} \Rightarrow A$ real $\Rightarrow L$ signed

3D formation control