Lecture 11

Today: start module on combinatorial optimization and integerer programmy

$$\min c^T x$$
$$\text{st } Ax \le b$$
$$x \in \mathbb{Z}^n$$

LP relaxation

assume $x \in \mathbb{R}^n$

solve LP

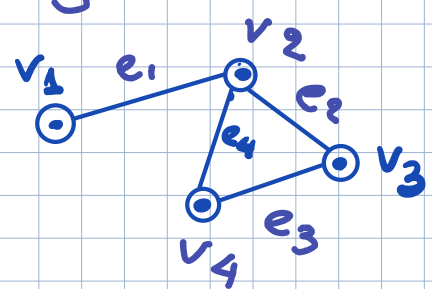
1) briefly review graph theory notation + basic facts
2) Min-cut max-flow theorem.

Graph Theory Notation

$G = (V, E)$

$E \subseteq [V^2]$

2 element subsets

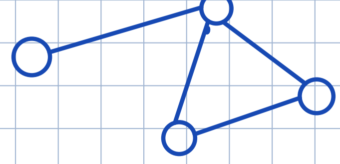


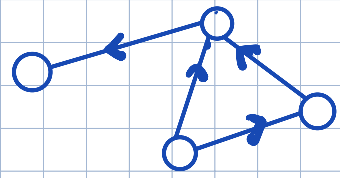
$e_1 = (v_1 v_2)$

$e_2 = (v_2 v_3)$

undirected: edges have no orientation

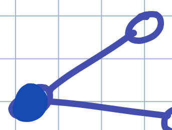
directed: edges have orientation.





Undirected graphs

degree of a vertex = # of neighbors (adjacent vertices)

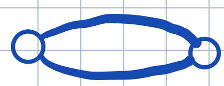


$d(v_1) = 2$

Simple graph: no self loops and no parallel edges

self loop:

parallel edges:

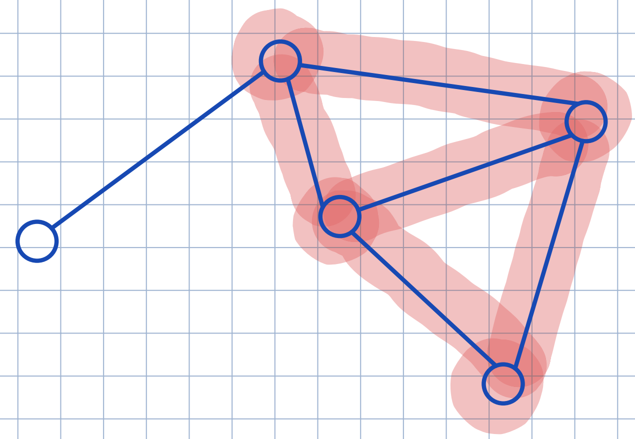


Consider two graphs $G = (V, E)$ and $G' = (V', E')$, we say

1) if $V' \subseteq V$, and $E' \subseteq E$ $\Leftrightarrow$ G' is a subgraph of G

2) if $V' = V$, $E' \subseteq E$, $\Leftrightarrow$ G' is a spanning subgraph

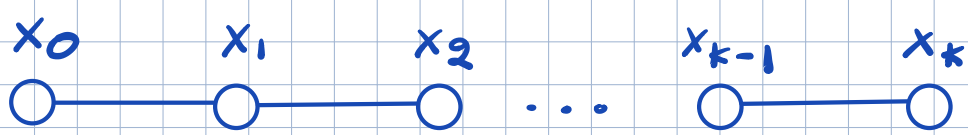
3) G' is a subgraph of G and contains all edges $xy \in E$ for each $x, y \in V'$ $\Leftrightarrow$ G' is an induced subgraph



Path: nonempty graph $P=(V,\mathcal{E})$

$V=\{x_0, x_1, x_2, \dots, x_k\}$ $\quad k \geq 1$ $\quad$ with all x_i distinct.

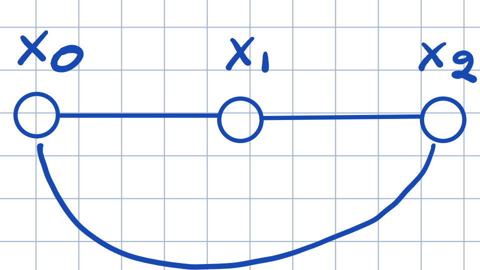
$\mathcal{E}=\{x_0x_1, x_1x_2, \dots, x_{k-1}x_k\}$



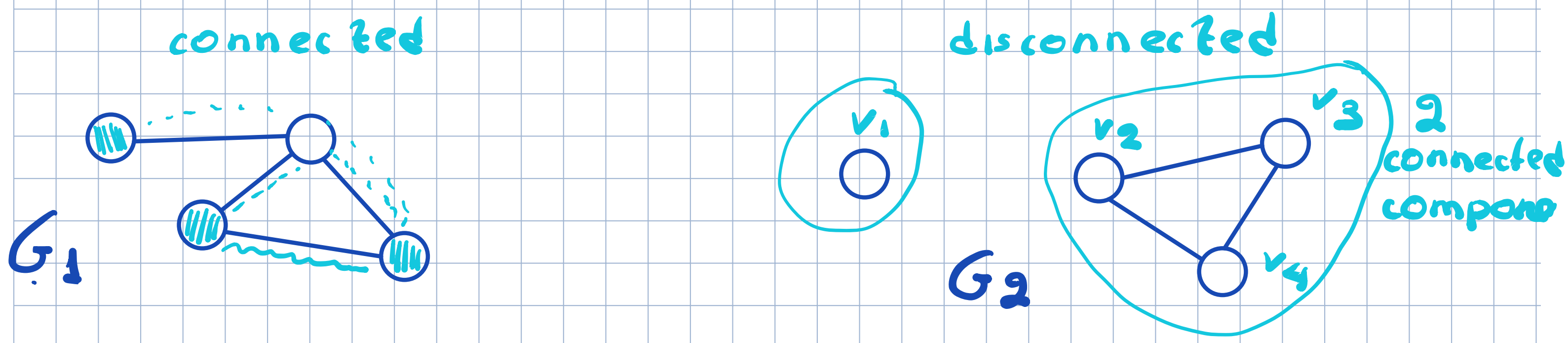
length of a path = # of edges in the path.

in the example length k.

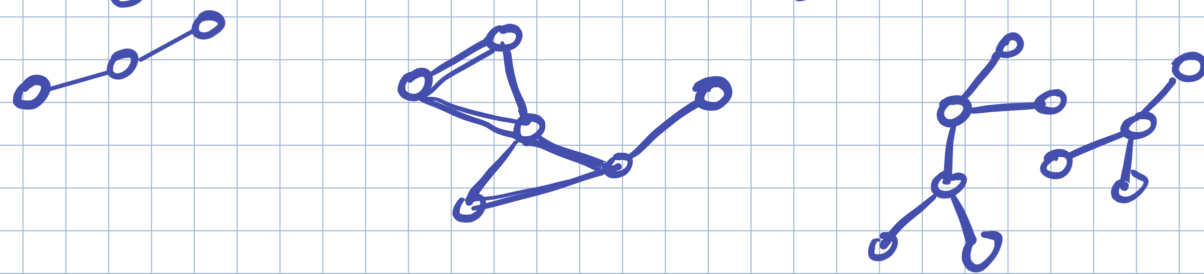
Cycle: if a path P has length at least two, the graph $P + x_kx_0$ is called a cycle.



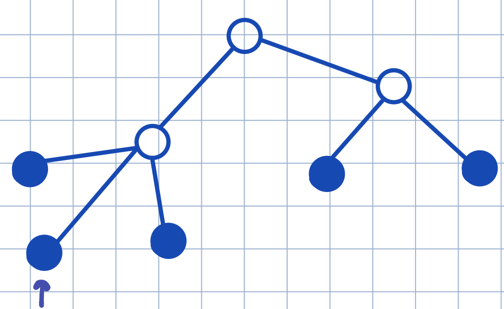
Connected graph: if any two vertices are connected by a path



Acyclic graph: contains no cycle



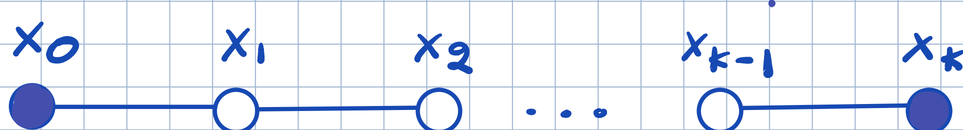
Connected acyclic graph = tree

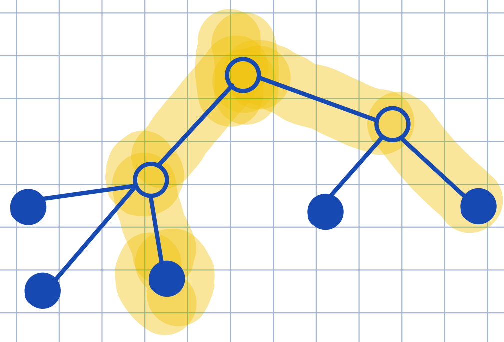


Trees : vertices of degree 1 are called "leaves"

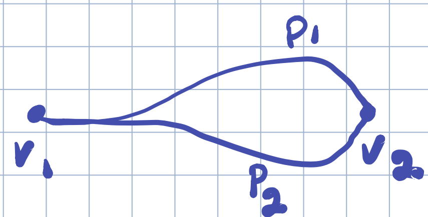
1) Each tree has at least <u>two leaves</u>

Path:



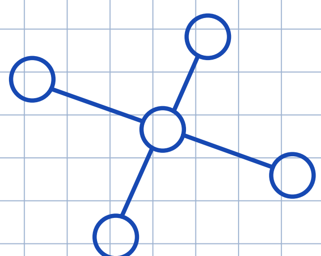


2) If T is a tree, any two vertices are connected by exactly one path.



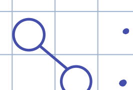
→ trees are acyclic.

3) A tree with n vertices has always $n-1$ edges.



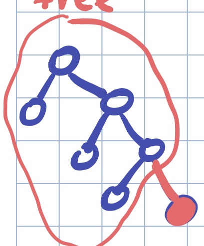
<u>Proof</u> induction on the number of vertices n

$n=2$, 1 edge ✓



assume that our statement holds $n=k$ vertice ($k-1$ edges)

tree

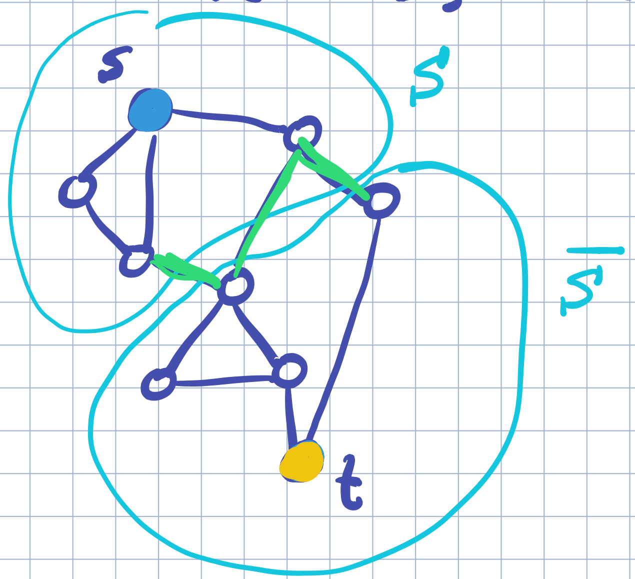


• consider $n=k+1$ vertices

remove one leaf, remaining graph is a tree with k vertices and $k-1$ edges ⇒ bringing back the leaf $\frac{k+1}{k}$

Cut:

Given a graph $G=(V, E)$, and two vertices, s (source), t (destination)



A cut is a partition of the vertices into two sets, S & $\overline{S}$, such that $s \in S$, and $t \in \overline{S}$

$$S \cup \overline{S} = V$$
$$S \cap \overline{S} = \emptyset$$

value of a cut: # of edges that connect S and $\overline{S}$

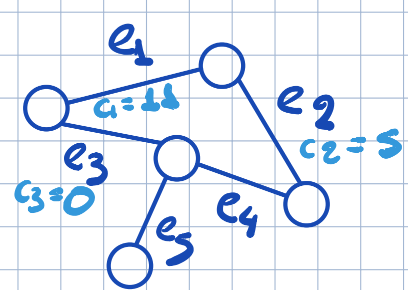
2^{n-2}

Capacity associated with every edge:

$$c: E \rightarrow \mathbb{R}^+$$

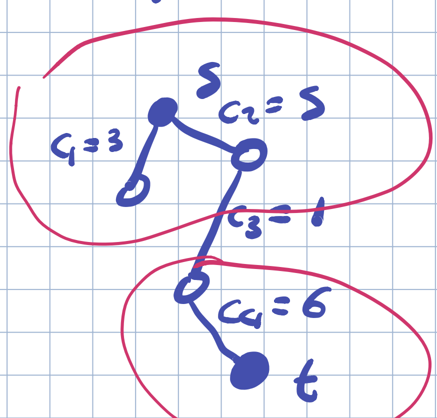
$c(e_1) = 11$ bits/sec

$c(e_2) = 5$ bits/sec.



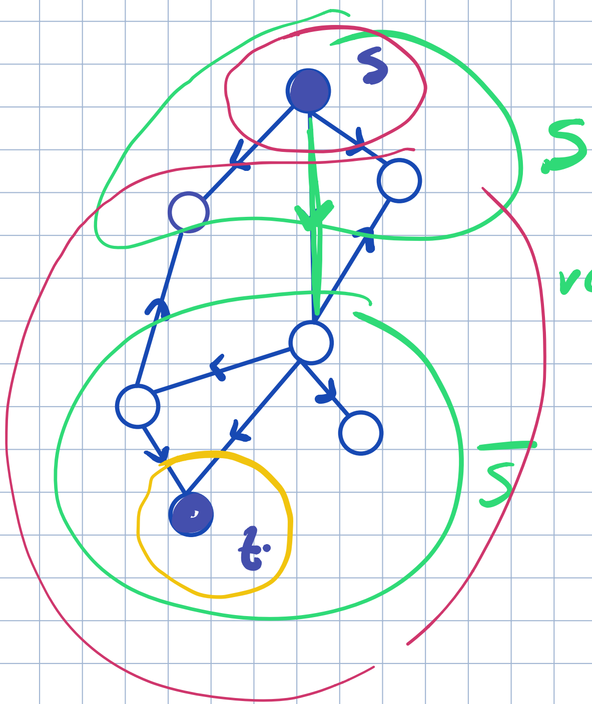
For capacitated graphs cut value:

sum of the capacities of the edges that cross the cut.



Directed graphs:

we only consider edges that go from S to $\bar{S}$



value of cut = 1

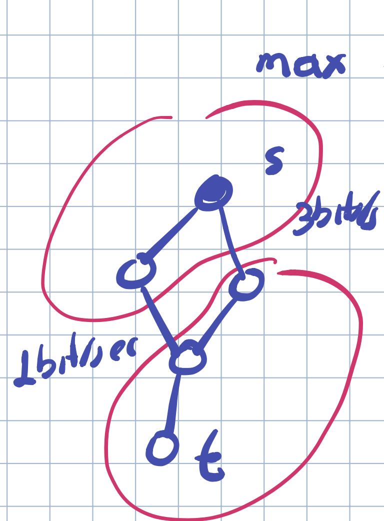
minimum cut = a cut that has the smallest possible value.

Min-cut max-flow theorem

max amount of information I can send from s to t = min among all values of cuts that separates s from t.

max flow $\leq$ min cut

$\geq$



$\sim$ 4 bits/sec

LP formulation for max-flow

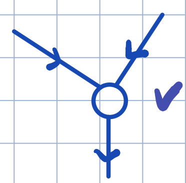
Consider a directed graph $G=(V,E)$, with capacited edges, and two distinct nodes s and t.
We want to find the maximum amount of flow that we can send from s to t subject to two constraints:

(1) capacity constraints.

---

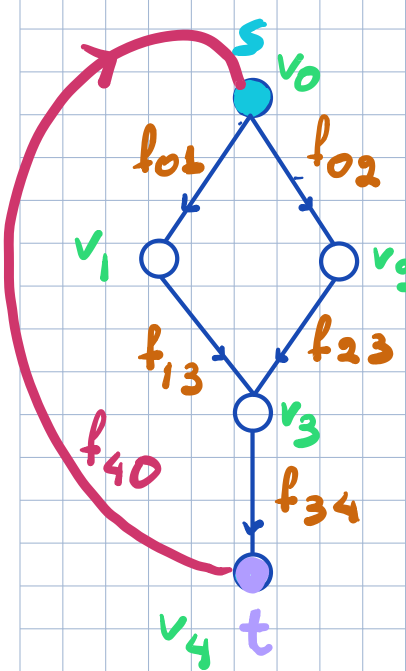
(2) flow conservation

Let v be any vertex (apart s & t)



it sends out as much flow as it receives.

add one artificial edge of infinite capacity



$v_0: \cancel{f_{40}} - \cancel{f_{01}} - f_{02} \leq 0$

$v_1: \cancel{f_{01}} - f_{12} \leq 0$

$v_2: f_{02} - f_{23} \leq 0$

$v_3: f_{13} + f_{23} - f_{34} \leq 0$

$v_4: f_{34} - \cancel{f_{40}} \leq 0$

$0 < 0$ contradiction

2nd trick: make equalities inequal.

assume $\exists$ feasible f_{ij}, $f_{01} - f_{12} < 0$

variables: amount of flow through every edge f_{ij}

objective function: $\max f_{ts}$

$$\max_{f_{ij}} \quad f_{ts}$$

s.t. $f_{ij} \leq c_{ij}$ capacity constraints for every edge e_{ij}

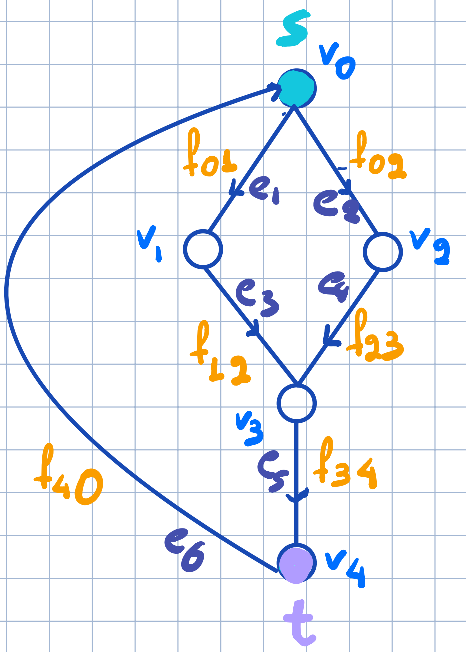
$$\sum_j f_{ji} - \sum_k f_{ik} \leq 0$$ for every vertex i flow conservation

$$f_{ij} \geq 0$$

We will write this LP in matrix format
introduce the edge-vertex adjacency matrix

$$M_{ij} = \begin{cases} 0 & \text{if edge } i \text{ is not adjacent to vertex } j \\ +1 & \text{if edge } i \text{ enters vertex } j \\ -1 & \text{if edge } i \text{ exits vertex } j \end{cases}$$

In our example $\qquad M \quad |E| \times |V|$



$$\begin{array}{c} \\ e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \end{array} \begin{array}{c} \begin{array}{ccccc} v_0 & v_1 & v_2 & v_3 & v_4 \end{array} \\ \begin{bmatrix} -1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \\ 1 & 0 & 0 & 0 & -1 \end{bmatrix} \end{array}$$

$$M^T \qquad \begin{array}{c|cccc} & e_1 & \cdots & & e_6 \\ \hline v_0 & -1 & -1 & \cdots & 1 \\ \vdots & & & & \\ v_4 & & & & \end{array} \qquad \begin{pmatrix} f_{01} \\ f_{02} \\ \vdots \\ \cdot \end{pmatrix}$$

max flow problem in vector notation

$$f = \begin{pmatrix} f_{01} \\ \vdots \\ f_{ts} \end{pmatrix} \in \mathbb{R}^n \qquad \min \; -e_n^T f$$

st

edges $\quad d_{ij} \leftarrow (I \;\; 0) f \leq c_{(n-1)}$

vertices $\quad p_i \leftarrow M^T f \leq 0$

$v_{ij} \leftarrow -f \leq 0$

to derive the dual, we associate dual

variables with the constraints of the primal.

Derive the dual:

$$\mathcal{L}(f, p, d, v) = -e_n^T f + d^T[(I \; 0) f - c] + p^T(m^T f) + v^T(-f)$$

Lagrange dual function.

$$g(p, d, v) = \inf_f \mathcal{L}(f, p, d, v) = \begin{cases} -d^T c & \text{if } (\quad) = 0 \\ -\infty & \text{otherwise} \end{cases}$$

for $g(\quad)$ to be a lower bound we need $d, p, v \geq 0$

Dual.

$$\max \; -d^T c$$

s.t.

$$e_n^T + p^T M^T + d^T(I \; 0) + v^T(-I) = 0$$

$$p, d, v \geq 0$$

---

steps expanding removing v slack variable.

$$\begin{array}{ll} \min & c^T d \\ \text{st} & P_s - P_t \geq 1 \\ & d_{ij} \geq P_i - P_j \\ & d_{ij} \geq 0,\ P_i \geq 0 \end{array}$$

LP2

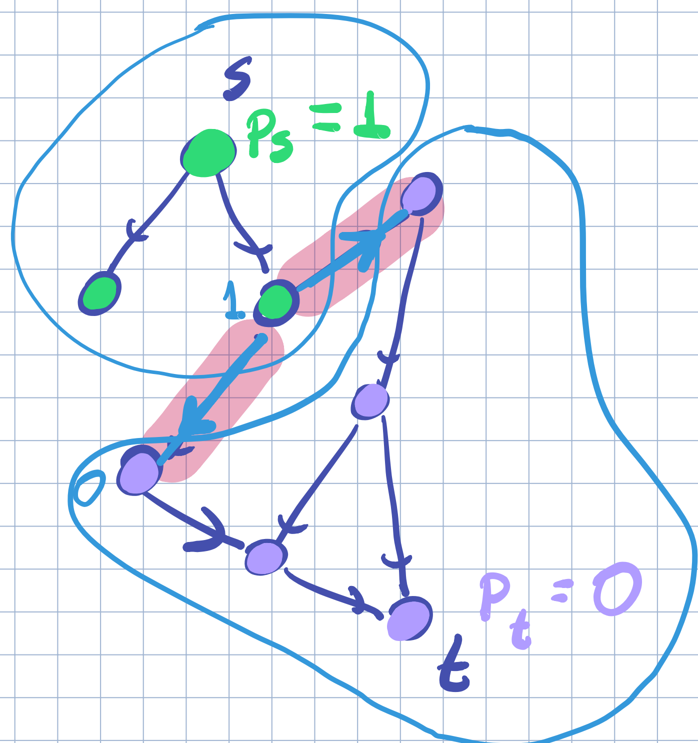
$d_{ij} \rightarrow$ "length" associated with edge ij

$P_i \rightarrow$ "potential" associate with vertex i.

Assume $d_{ij},\ P_i \in \{0, 1\}$

our program becomes an integer LP.

$$\begin{array}{ll} \min & c^T d \\ \text{st} & P_s - P_t \geq 1 \\ & d_{ij} \geq P_i - P_j \\ & d_{ij}, P_i \in \{0,1\} \end{array}$$



only feasible choice
$P_s = 1$
$P_t = 0$

some of the remain vertices have $P_i = 0$ and some $P_i = 1$.

for edges that cross the cut $d_{ij} \geq 1 \Rightarrow d_{ij} = 1$