Shortest Path LP.

Directed graph $G(V, E)$.
Each edge $e \in E$ has a non-negative length L_e.
(or no negative length cycles)

Source vertex $\longrightarrow s$
destination vertex $\longrightarrow t$

Find a shortest path from s to t.

$x_e \rightarrow 0$ e not present in the path
$x_e \rightarrow 1$ e present in the path.

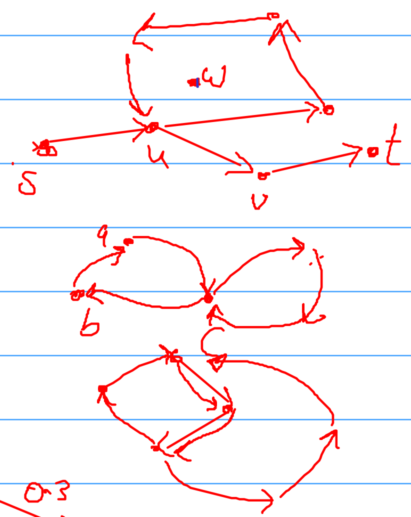
$$\text{Min} \sum_e L_e x_e$$

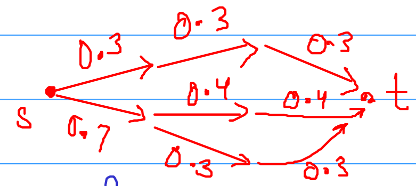
for each $e \in E$ $x_e \geq 0$

for $u \neq s, t$ $\displaystyle\sum_{e \in in(u)} x_e = \sum_{e \in out(u)} x_e$

$$\sum_{e \in out(s)} x_e = 1$$

$$\sum_{e \in In(t)} x_e = 1$$





→ Is it the right linear program?

Can LP optimal be smaller than the shortest path length?
No.

It turns out that there is always an integral optimal solution for this LP.

HW Try to prove via $\pm \varepsilon$ argument on cycle.

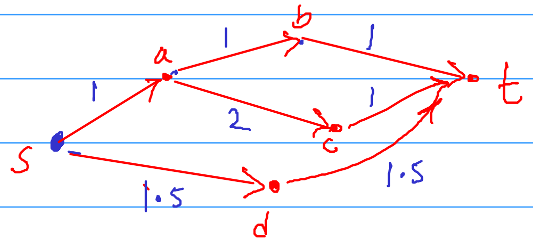
We will prove this via dual LP.

Dual LP for the shortest path

	Primal	Dual
	Min $\sum_e L_e x_e$	Max $y_t - y_s$
for each $e \in E$	$\cancel{1 \geq}\ x_e \geq 0$	y_u $\quad u \in V$
for $u \neq s, t$	$\sum_{e \in in(u)} x_e - \sum_{e \in out(u)} x_e = 0$	for each $e = (p,q) \in E$ $\quad p \xrightarrow{e} q$
$\rightarrow$	$-\sum_{e \in out(s)} x_e = -1$	$y_q - y_p \leq L_e$
$\rightarrow$	$\sum_{e \in In(t)} x_e = 1$	

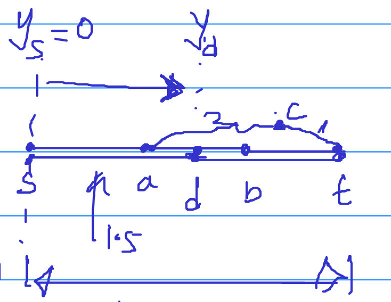
Physical interpretation of the dual:

Imagine a length l_e thread for edge e.
Pull t as far away from s as possible.



min $\sum L_e x_e$

$$-x_{sa} - x_{sd} = -1$$
$$x_{sa} - (x_{ab} + x_{ac}) = 0$$
$$x_{sd} - x_{dt} = 0$$
$$x_{ac} - x_{ct} = 0$$
$$x_{ab} - x_{bt} = 0$$
$$x_{bt} + x_{ct} + x_{dt} = 1$$



optimal solution

Max $y_t - y_s$

$y_a - y_s \leq 1$ $\qquad y_t - y_c \leq 1$

$y_b - y_a \leq 1$ $\qquad y_d - y_s \leq 1.5$

$y_t - y_b \leq 1$ $\qquad y_t - y_d \leq 1.5$

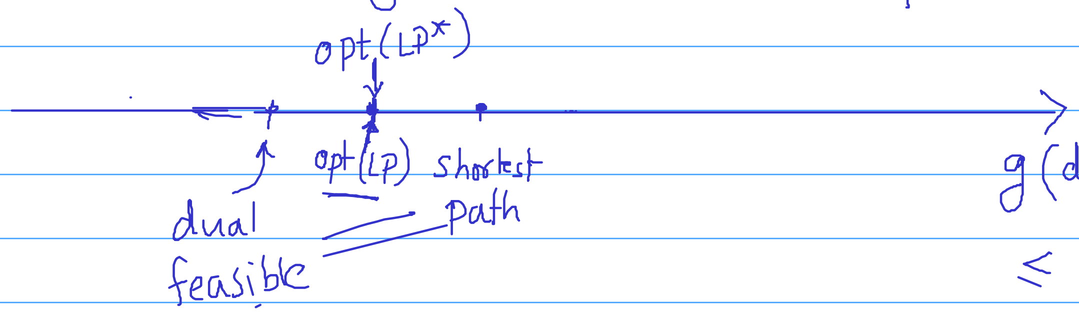
$y_c - y_a \leq 2$

$y_a \longrightarrow$ distance of a from s.

Tight threads correspond to tight dual constraints.

Claim The optimal value of shortest path LP is equal to the length of the shortest path.

proof: We will construct a dual feasible solution whose dual objective value is equal to the length of the shortest path.



$g(\text{dual feasible}) \leq g(\text{dual opt})$

Take $y_s = 0$ and $y_u = \text{dist}(s, u)$. $(\text{dist}(s,q) \leq L_{e(p,q)} + \text{dist}(s,p))$

Verify $y_q - y_p \leq L_e$ (feasibility)

$y_t - y_s = \text{dist}(t,s) - 0 = \text{shortest path } (s,t)$

Complementary Slackness

$$\begin{aligned} \max\ & w^T x \\ & Ax \leq b \\ & x \geq 0 \end{aligned} \qquad\qquad \begin{aligned} \min\ & b^T y \\ & A^T y \geq w \\ & y \geq 0 \end{aligned}$$

$$\max \sum_{i=1}^{n} w_i x_i \qquad\qquad \min \sum_j b_j y_j$$

for $1 \leq j \leq k$ $\sum_i a_{ji} x_i \leq b_j$ $\leftarrow$ for $1 \leq i \leq n$ $\sum_j a_{ji} y_j \geq w_i$ $\leftarrow$

$x_i \geq 0$ for $1 \leq j \leq k$ $y_j \geq 0$

For any feasible x for primal LP and feasible y for dual LP

$$\underbrace{\sum_i w_i x_i}_{\text{primal obj}} \leq \sum_i x_i \left(\sum_j a_{ji} y_j\right) = \sum_j y_j \left(\sum_i a_{ji} x_i\right) \leq \underbrace{\sum_j y_j b_j}_{\text{dual obj}}$$

weak duality

Strong Duality

For any primal optimal solution x^* and dual optimal solution y^*

$$\sum_i w_i x_i^* = \sum_i \sum_j a_{ji} y_j^* x_i^* = \sum_j \sum_i a_{ji} x_i^* y_j^* = \sum_j b_j y_j^*$$

$$\sum_i \underbrace{x_i^*}_{\geq 0} \underbrace{\left(\sum a_{ji} y_j^* - w_i\right)}_{\geq 0} = 0 \implies$$ every term in the summation is zero.

For each $1 \leq i \leq n$, $\quad x_i^* \left(\sum_j a_{ji} y_j^* - w_i \right) = 0$

$$\Rightarrow \qquad x_i^* = 0 \quad \text{or} \quad \sum_j a_{ji} y_j^* = w_i$$

the i-th primal variable is zero or
the i-th dual constraint is tight.

$$x_i^* > 0 \implies \sum_j a_{ji} y_j^* = w_i$$

$$\sum_j a_{ji} y_j^* > w_i \implies x_i^* = 0$$

Similarly,
for each $1 \leq j \leq k$ $\quad y_j^* = 0 \quad$ or $\quad \sum_i a_{ji} x_i^* = b_j$

the j-th dual variable is zero or
the j-th primal constraint is tight.

$$y_j^* > 0 \implies \sum_i a_{ji} x_i^* = b_i$$

$$\sum_i a_{ji} x_i^* < b_i \implies y_j^* = 0$$

Only those dual variables are nonzero for which the corresponding primal constraint is tight.

Complementary slackness.

Let x & y be some primal feasible and dual feasible solutions.

They are optimal if and only if they satisfy complementary slackness i.e.,

① for each i, $x_i > 0 \Rightarrow \sum_j a_{ji} y_j = w_i$

② for each j, $y_j > 0 \Rightarrow \sum_i a_{ji} x_i = b_j$

We can use complementary slackness to show optimality of solutions.

Approximate complementary slackness

Let x and y be primal feasible and dual feasible solutions s.t.

$$\left.\begin{array}{l} \text{① for each } i,\ x_i > 0 \Rightarrow \alpha \cdot w_i \geq \sum_j a_{ji} y_j \geq w_i \\ \text{② for each } j,\ y_j > 0 \Rightarrow \frac{b_j}{\alpha} \leq \sum_i a_{ji} x_i \leq b_j \end{array}\right\} \alpha > 1$$

HW Prove that x and y are approximately optimal.

HW LP for maximum matching in a bipartite graph

Dual LP $\longleftarrow$ minimum vertex cover

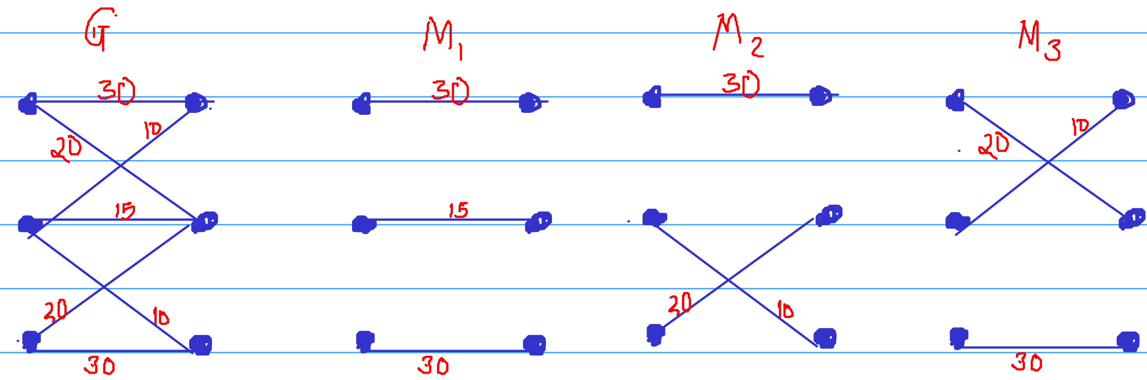
König Theorem: In a bipartite graph, |maximum matching| = |minimum vertex cover|

Prove it via LP duality.

Primal Dual algorithm for weighted bipartite matching.

Given weights on the edges the goal is to find a minimum weight perfect matching.

Other versions like maximum weight perfect matching, maximum matching can be converted to this.



An example with three perfect matchings

LP	Dual LP
$\min \sum_{e \in E} w_e x_e$	$\max \sum_{u \in V} y_u$
$x_e \geq 0$ for $e \in E$	y_u for $u \in V$
$\sum_{e \in \delta(u)} x_e = 1$ for $u \in V$	$y_u + y_v \leq w_e$ for $e = (u,v) \in E$

$\delta(u) \leftarrow$ edges incident on u

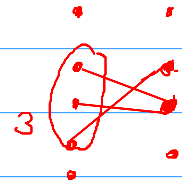
complementary slackness

for each $e \in E$, $x_e > 0 \Rightarrow y_u + y_v = w_e$ $\quad e = (u,v)$

$y_u + y_v < w_e \Rightarrow x_e = 0$

We will assume we have an algorithm that tests existence of a perfect matching and if there is no perfect matching then output a Hall's block.

a subset $S \subseteq L$ s.t. $|N(S)| < |S|$.



Think of it as a reduction from minimum weight perfect matching to existence of a perfect matching. (bipartite graphs)

Algorithm outline

① Start with a dual feasible solution y (zero on the left, min weight on the right)

② Try to construct a primal feasible solution x s.t. (x,y) satisfy complementary slackness (construct a PM using tight edges $y_u + y_v = w_e$)

$\xrightarrow{\text{succeed}}$ optimal solution (stop)

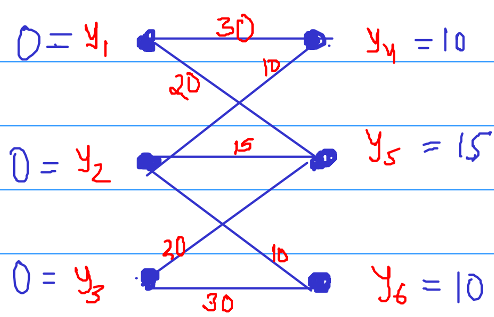
$\xrightarrow{\text{Fail}}$ Modify the dual solution y s.t. dual objective value increases

Go to ②

$\exists\, S,\ |N(S)| < |S|,$ increase y variables for S by ε, decrease y variables for $N(S)$ by ε

Starting feasible solution

$y_u + y_v \leq w_e$ for each e



$y_1 + y_5 \leq 20$

Assign y variables to zero for one side of vertices. For the other side, assign y_u the min weight of any edge incident on u.

Primal feasible with complementary slackness

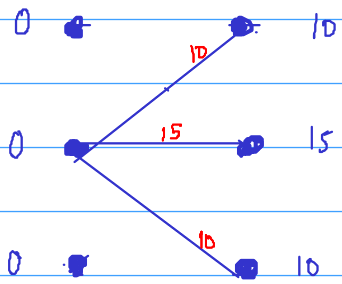
For $e=(u,v) \in E$ $\quad y_u + y_v < w_e \implies x_e = 0$

non-tight edges

Tight edges $\longrightarrow$ $e \in E$ s.t. $y_u + y_v = w_e$

will work with only tight edges.

Feasible solution exists if and only if there is perfect matching (not obvious)



$$x_e \geq 0 \qquad e \in E$$

$$\sum_{e \in \delta(u)} x_e = 1 \qquad u \in V$$

Try to construct a perfect matching using only the tight edges.

The tight subgraph

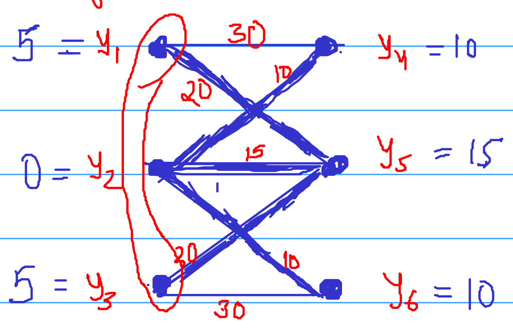
If there is no perfect matching, $\exists\ S \subseteq L$ s.t. $|N(S)| < |S|$.

$N(S)$: tight neighborhood of S.

Hall's block.

No primal feasible solution means we can improve the dual solution

Improving the dual while maintaining feasibility.

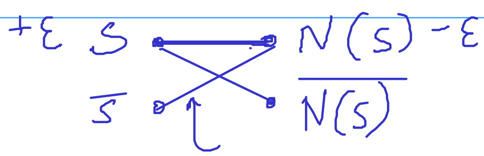


$S = \{u_1, u_3\}$

$N(S) = \{u_5\}$

$|N(S)| < |S|$

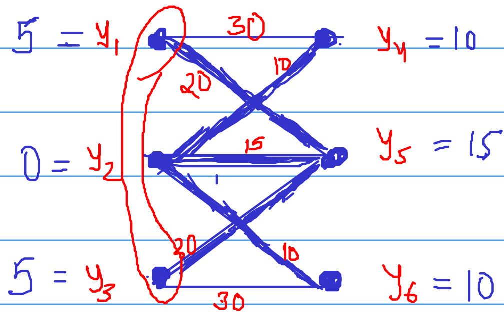
[Increase the dual variables for S by ε and decrease the dual variables for $N(S)$ by ε.



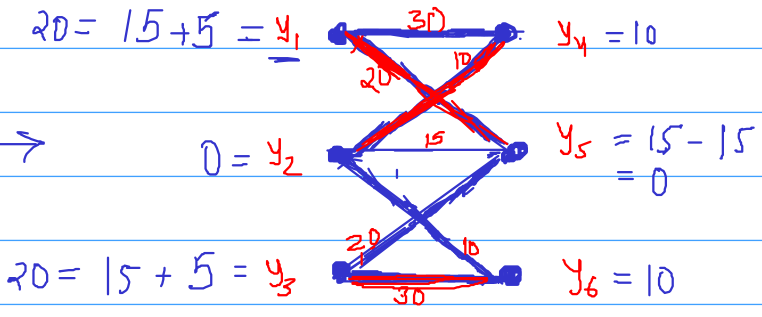
Lecture 11 (Feb 11)

$$\underset{+\varepsilon}{y_1} + \underset{-\varepsilon}{y_5} \le 20$$

$$\underset{+\varepsilon}{y_1} + y_4 \le 30 \implies \varepsilon \le 15$$



$\longrightarrow$



$\varepsilon \le 15$

Try to construct a PM using tight edges

$$PM = \{ \underset{20}{(u_1, u_5)} \ \underset{10}{(u_2, u_4)} \ \underset{30}{(u_3, u_6)} \} \quad = 60$$

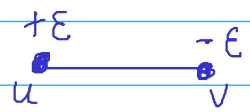
Choice of ε.

We need to maintain feasibility. $(y_u + y_v \le w_e)$

Three kinds of edges (u, v)

① $u \in S, \quad v \in N(S)$

$$\underset{+\varepsilon}{y_u} + \underset{-\varepsilon}{y_v} \le w_e \qquad \text{remains valid.}$$



② $u \notin S$, v might or might not be in $N(S)$

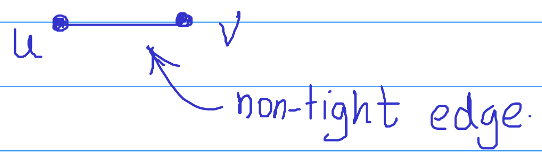
$$\underset{+0}{y_u} + \underset{-\varepsilon/0}{y_v} \le w_e$$

The sum $y_u + y_v$ either remains same or it decreases

The constraint remains satisfied.

③ $u \in S \qquad v \notin N(S)$

$$\underset{+\varepsilon}{y_u} + \underset{0}{y_v} < w_e$$



$$\varepsilon = \min_{\substack{e=(u,v): \\ u \in S \\ v \notin N(S)}} \{ w_e - y_u - y_v \}$$

With this choice of ε, every edge constraint remains satisfied.

Change in the dual objective in an iteration.

dual objective $\sum_{u \in V} y_u$

$$\text{change} = |S| \cdot \varepsilon - |N(S)|\varepsilon = \varepsilon \underbrace{\left(|S| - |N(S)|\right)}_{>0}$$

Why does the algorithm terminate?

Whenever there is no PM among the tight edges, there is a way to increase the dual objective value.

Dual objective is always upper bounded by minimum weight of a PM.

Number of iterations?

If all weights are integral, then ε is also integral.
$\Rightarrow \varepsilon \geq 1$

No. of iterations $\leq$ min wt of a PM

Pseudopolynomial time. Poly (n, W) $\rightarrow$ max absolute value of any weight.

Input size $\leftarrow$ $n, m, \log W$

A different analysis can show strongly polynomial time.

no. of arithmetic operations $=$ Poly (n).

- Primal dual approach can be applied on any LP.

HW

- Show that whenever there is no primal solution satisfying complementary slackness with the current dual solution, there is a way to improve the dual solution.

- Finding this improvement is not trivial. (itself an LP problem)

Primal-dual approach can be viewed as a reduction from an LP to a simpler LP.

Minimum weight perfect matching in general graphs.

→ LP description non-trivial (exponential size) [1965 Edmonds]

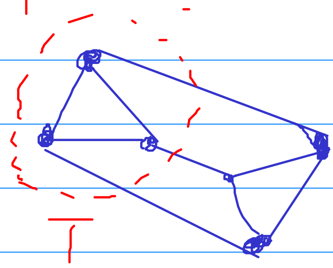
→ the primal dual algorithm runs in polynomial time even though LP description is exponential size.

combinatorial problems for which an exact LP description is known.

- Matching

$$x_e \geq 0 \qquad e \in E$$

$$\sum_{e \in \delta(v)} x_e = 1 \qquad v \in V$$

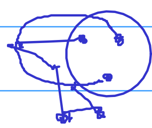


T $\bar{T}$

$\forall$ $T \subseteq V$, $|T|$ is odd $\qquad \sum_{e \in \delta(T, \bar{T})} x_e \geq 1$

edges connecting T with $\bar{T}$.

- Spanning trees



$$x_e \geq 0 \qquad e \in E$$

for all $S \subseteq V$ $\qquad \sum_{e \in E[S]} x_e \leq |S| - 1$

- Odd/even subset

- Independent set in interval graphs, bipartite graphs, more generally perfect graphs.

- vertex cover in bipartite graphs

- It's unreasonable to expect an exact LP description for TSP / Independent set / vertex cover in general graphs. even with exponentially constraints.

Why?

Because a nice LP description will put these problems in coNP.

And people believe NP $\neq$ coNP.

Lecture 12, Feb 14

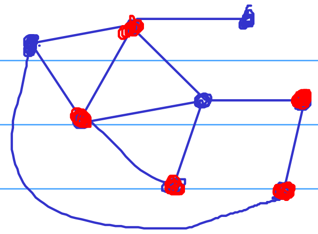
Primal dual schema

Exact Algorithms

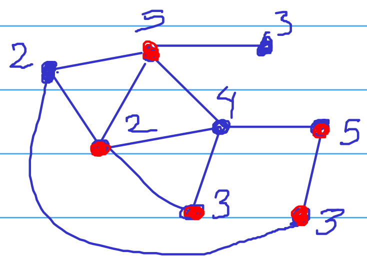
- Min weight perfect matching in bipartite graphs (also known as assignment problem)
- Min weight perfect matching in non-bipartite graphs
- Dijkstra's algorithm
- Ford Fulkerson algorithm for maximum flow.

Approximation algorithms for NP-hard problems

Vertex Cover : a set of vertices such that every edge is covered by it.



- Minimum size vertex cover
- Minimum weight vertex cover (given weights on vertices)



Easy on bipartite graphs.

2- approx algorithm for minimum size vertex cover.

1. $S \leftarrow$ empty set of vertices
2. While there is an edge e that is not covered by S
   put both endpoints of e in S.

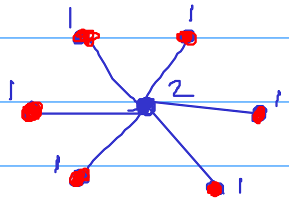
Claim: S is at most twice of the min vertex cover.

Proof: HW

Minimum weight vertex cover:

The simple greedy algorithm performs very badly.

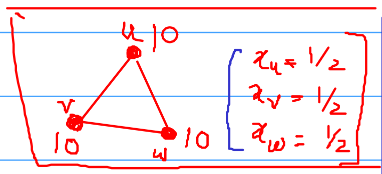
Simple greedy : select the minimum weight vertex that still has uncovered edges incident on it.



optimal $\leftarrow$ center vertex.

Primal Dual Approach:

We will work with an LP formulation even though it's not an exact formulation.



LP

$$\min \sum_{u \in V} w_u x_u$$

$x_u + x_v \geq 1$ for each edge $(u,v) \in E$

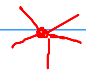
$x_u \geq 0$ for each $u \in V$

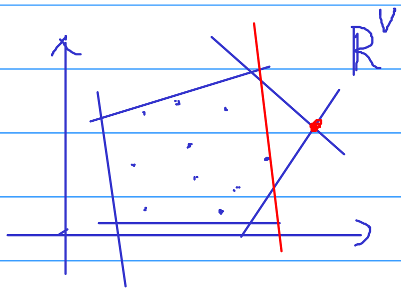
Dual LP

$$\max \sum_{e \in E} y_e$$

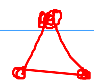
$y_e \geq 0$

$$\sum_{e \in \delta(u)} y_e \leq w_u$$ for each $u \in V$





for every triangle u, v, w



$$x_u + x_v + x_w \geq 2$$

Not needed

The choice of LP is not obvious. It's a part of the algorithm design process.

Complementary slackness conditions:

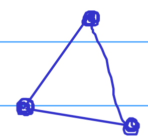
1. for $e \in E$, $y_e > 0 \implies x_u + x_v = 1$ where $e = (u,v)$

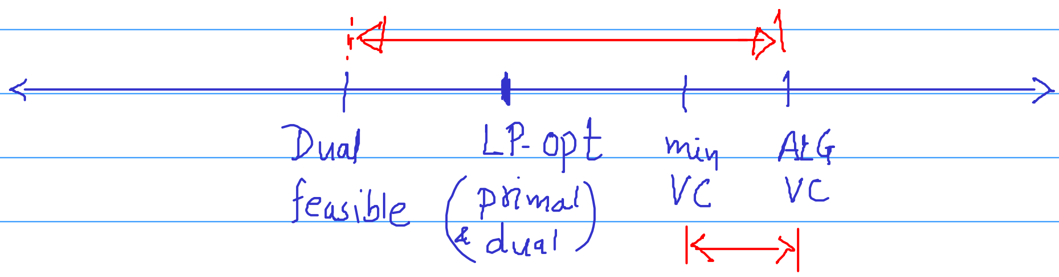
2. for $u \in V$, $x_u > 0 \implies \sum_{e \in \delta(u)} y_e = w_u$

$$\sum_{e \in \delta(u)} y_e < w_u \implies x_u = 0$$

Most likely it will not be possible to find an integral primal feasible solution (vertex cover) satisfying complementary slackness conditions with a dual solution.

Because it can be that all primal optimal solutions are fractional.





We will construct a vertex cover (primal feasible) and a dual feasible solution and bound the gap between the two.

$$\text{ALG-VC} \leq 2 \cdot \text{Dual feasible} \leq 2 \cdot \text{LP opt} \leq 2 \cdot \text{min VC}$$

We will do this by showing approximate complementary slackness.

Approximate complementary slackness

1. for $e \in E$, $y_e > 0 \Rightarrow \underbrace{2 \geq x_u + x_v \geq 1}$ where $e = (u,v)$ → True automatically

2. for $u \in V$, $\sum_{e \in \delta(u)} y_e < w_u \Rightarrow x_u = 0$

$$x_u > 0 \Rightarrow \sum_{e \in \delta(u)} y_e = w_u$$

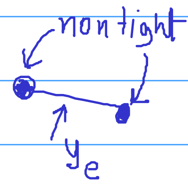
Primal Dual Algorithm

0: $S = \emptyset$

1. Start with a feasible dual solution. (all $y_e = 0$)

2: For tight vertices (tight dual constraints)
→ set $x_u = 1$ (include u in S)

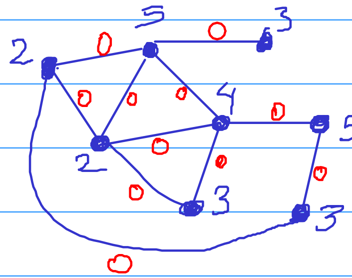
3. Check whether we have a vertex cover.
If yes, output S. ($S = \{u \in V : x_u = 1\}$)

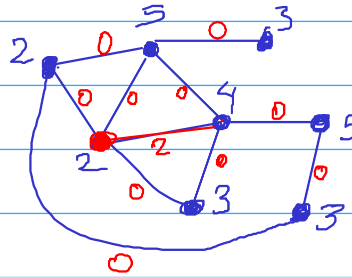


If no, take an uncovered edge e
and increase the dual variable y_e
(till one of the dual constraints become tight)

4. Go to 2.

Algorithm on an example



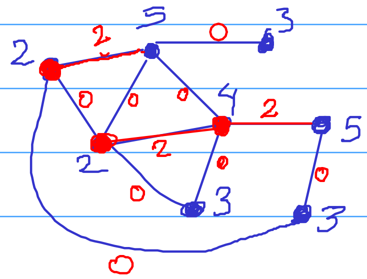


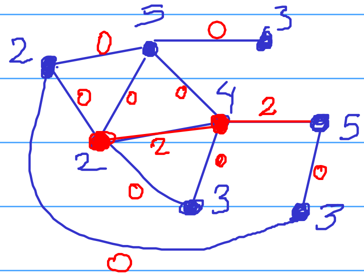
$y_1 + y_2 + y_3 + y_4 \le 2$

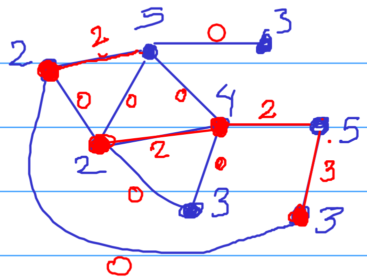
0 0 0 0

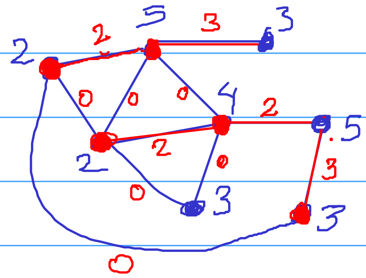
2

$y_1 + y_2 + y_3 + y_4 = 2$









ALG VC = 16.

min VC = 14.

Dual obj = 12

Claim: The algorithm gives a 2-approximate vertex cover.

$$\left(ALGVC \le 2 \cdot \min VC \right)$$

Proof:

$$ALGVC = \sum_{u \in V} x_u w_u = \sum_{u \in S} w_u$$

complementary slackness 2.

x is output by ALG

$$= \sum_{u \in S} \left(\sum_{e \in \delta(u)} y_e \right).$$

Approximate Complementary slackness 1

$$\le \sum_{e \in E} y_e \cdot 2 = 2 \cdot \text{dual obj}$$

$$ALG\ VC \le 2 \cdot \underset{\text{(feasible dual)}}{\text{dualobj}} \le 2 \cdot LP\text{-}OPT \le 2 \cdot \min VC.$$

- There is no polynomial time algorithm known that achieves 1.99-approximation for min weight vertex cover.

Lecture 13 Thu Feb17

Application of Linear programming duality in Game theory

- Multiple agents / players
- Each player takes own decisions
- Payoff's for any player are dependent on everyone's decision.

Example : Prisoner's dilemma

Two prisoners interrogated separately.
They can remain silent or turn approver.

- If both remain silent → 1 year jail for each
- If one silent, one approves → freedom for approver, 3 years jail for silent
- If both approve → 2 years jail for each.

A ↓ \ B →	silent	Approve
Silent	-1, -1	-3, 0
Approve	0, -3	-2, -2

Payoff matrix

Both will become approvers.

Nash equilibrium : A tuple of strategies such that no player can get any advantage by unilaterally changing her/his strategy.

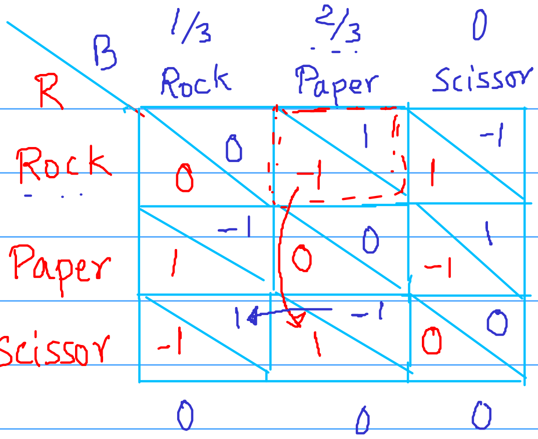
(Approve, Approve) → nash equilibrium.

Rock > Scissor

Scissor > paper

paper > Rock

Example 2



		B	$1/3$	$2/3$	0
	R		Rock	Paper	Scissor
$1/2$	Rock		0, 0	-1, 1	1, -1
$1/4$	Paper		1, -1	0, 0	-1, 1
$1/4$	Scissor		-1, 1	1, -1	0, 0
			0	0	0

Nash equilibrium

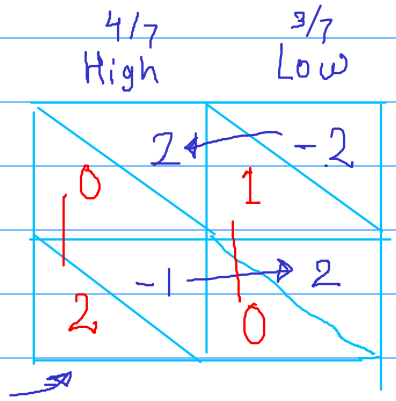
$\rightarrow (1/3, 1/3, 1/3) \leftarrow$

$(1/3, 1/3, 1/3) \leftarrow$

Payoff $\rightarrow$ 0

Payoff $\rightarrow$ 0

Is there a Nash equilibrium?



		$4/7$ High	$3/7$ Low
$\frac{1}{3}$	High	0, 2	1, -2
$\frac{2}{3}$	Low	2, -1	0, 2

$$\frac{1}{3}\left(\frac{4}{7}\times 2 - \frac{3}{7}\times 2\right)$$

$$+\frac{2}{3}\left(\frac{4}{7}\times -1 + \frac{3}{7}\times 2\right)$$

$$\underbrace{\begin{bmatrix} 1/3 & 2/3 \end{bmatrix}}_{x^T} \underbrace{\begin{bmatrix} 2 & -2 \\ -1 & 2 \end{bmatrix}}_{C} \underbrace{\begin{bmatrix} 4/7 \\ 3/7 \end{bmatrix}}_{y} =$$

Mixed strategies: a probabilistic combination of pure strategies.

Let R be the payoff matrix of the row player and C be the payoff matrix of the column player

$R \leftarrow k \times n$ matrix

$C \leftarrow k \times n$ matrix

Mixed strategy for row player $(x_1, x_2, \cdots, x_k) \in \mathbb{R}^k$

$\sum x_i = 1, \quad x_i \geq 0$

for column player $(y_1, y_2, \cdots, y_n) \in \mathbb{R}^n$

$\sum y_i = 1, \quad y_i \geq 0$

Def

(x, y) is said to be Nash equilibrium if

- $x^T R y \geq x'^T R y$ for any x' s.t. $\sum x'_i = 1$, $x'_i \geq 0$
- $x^T C y \geq x^T R y'$ for any y'

Claim: (x, y) is Nash equilibrium if and only if

- $x^T R y \geq e_i^T R y$ for each $1 \leq i \leq k$ $\quad e_i = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}$
- $x^T C y \geq x^T R e_j$ for each $1 \leq j \leq n$.

Theorem: [1951] There is always a tuple of mixed strategies that is a Nash equilibrium.

Two player zero sum Game.

$$R + C = 0$$

[1940s] John von Neuman & Morgenstern

Nash equilibrium for two player zero sum games.

Equivalent to LP duality.

	A	B	C	
1	20, −20	10, −10	30, −30	1/2
2	−10, 10	20, −20	10, −10	1/2
	−5	−15	−20	

Suppose row player first decides the strategy and then it will be revealed to column player.

What should be the row player's strategy?

R \ B	1	2
1	30, −30	−10, 10
2	−20, 20	10, −10
	−5	$\boxed{0}$

Row player 1 $\rightarrow$ Column $\rightarrow$ 2

Payoff −10

Row player 2 $\rightarrow$ Column $\rightarrow$ 1

Payoff −20

Row player (1/2, 1/2) $\rightarrow$ Column 2

Payoff $\rightarrow$ 0

Row player (x_1, x_2) $\qquad x_1 + x_2 = 1$

Column Player 1 $\quad -30x_1 + 20x_2$

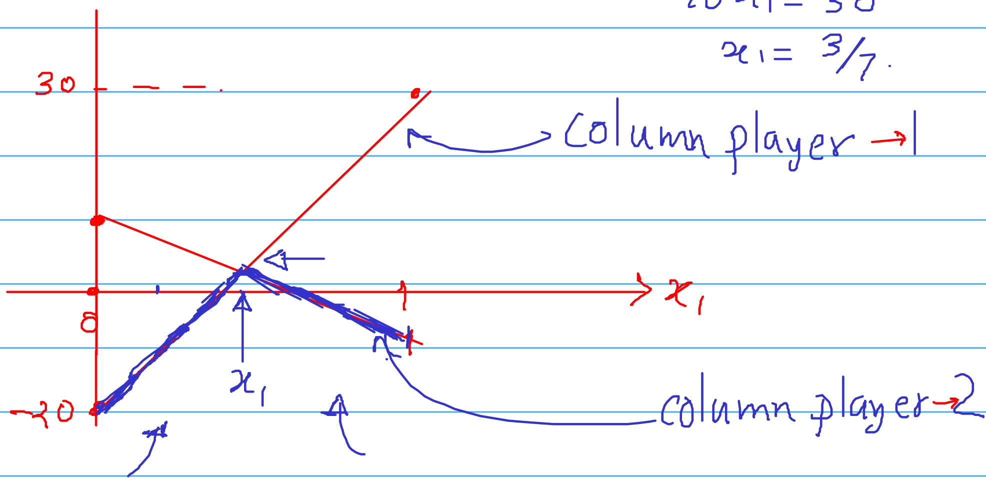
column player 2 $\rightarrow$ $10x_1 - 10x_2$

$$\max\left(-30x_1 + 20x_2,\ 10x_1 - 10x_2\right)$$

$$\max_{x_1, x_2}\left(\min\left(\underbrace{30x_1 - 20x_2}_{=50x_1 - 20},\ \underbrace{-10x_1 + 10x_2}_{-20x_1 + 10}\right)\right)$$

$x_1 + x_2 = 1$

$50x_1 - 20$
$= -20x_1 + 10$
$70x_1 = 30$
$x_1 = 3/7.$



blue plot payoff (x_1)

$\left(\frac{3}{7}, \frac{4}{7}\right)$ Payoff $= \frac{10}{7}$

Column player → (y_1, y_2)

↳ Row player 1 → $-30y_1 + 10y_2$
Row player 2 → $20y_1 - 10y_2$

$$\frac{-10}{7} = \max_{y_1, y_2} \underbrace{\min\left(-30y_1 + 10y_2,\ 20y_1 - 10y_2\right)}_{\max(30y_1 - 10y_2,\ -20y_1 + 10y_2)} \Rightarrow x^T C y$$

$-30y_1 + 10y_2 = 20y_1 - 10y_2$
$-50y_1 = -20y_2$
$y_1 + y_2 = 1$
$y_1 + \frac{5}{2} y_1 = 1$
$\Rightarrow y_1 = 2/7$
$y_2 = 5/7$

$\left(\frac{2}{7}, \frac{5}{7}\right)$ $= \frac{10}{7}$ $\frac{10}{7}$ $\frac{10}{7}$

Row player $\left(\frac{3}{7}, \frac{4}{7}\right)$
Column player $\left(\frac{2}{7}, \frac{5}{7}\right)$
Nash equilibrium

Note: $\max_y \min_x x^T C y = - \min_y \max_x x^T R y$

Why is the above pair of strategies a Nash equilibrium?

Because when column player fixes the strategy $(\frac{2}{7}, \frac{5}{7})$ the whatever the row player does, the payoff is always at most $10/7$.

Hence, there is no advantage in unilaterally changing.

Minimax Theorem

$$\max_x \min_y x^T R y = \min_y \max_x x^T R y$$

The above equality equivalent to Strong LP duality.

LP_1

$$\max \; Z$$
$$30x_1 - 20x_2 \geq Z$$
$$-10x_1 + 10x_2 \geq Z$$
$$x_1 + x_2 = 1$$
$$x_1 \geq 0$$
$$x_2 \geq 0$$

LP_2

$$\max \; \omega$$
$$-30y_1 + 10y_2 \geq \omega$$
$$20y_1 - 10y_2 \geq \omega$$
$$y_1 + y_2 = 1$$
$$y_1 \geq 0$$
$$y_2 \geq 0$$

Prove that LP_2 is equivalent to dual of LP_1 (upto −)

Hence the two optimal values are related.

Monday, Feb 28

Generalization of vertex cover: covering problems.

- Hitting set:

Given set of objects A with weights.
given subsets of A,
$T_1, T_2, \dots, T_k \subseteq A$

find a subset $S \subseteq A$ with minimum weight
that hits every T_i (take at least one element from T_i)

LP $\min \sum_{a \in A} x_a w_a$

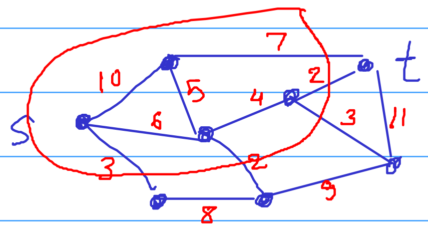
$x_a \geq 0$ for $a \in A$

$|T_i| \geq \sum_{a \in T_i} x_a \geq 1$

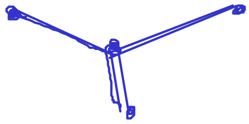
Approximation factor guarantee: $\max_i |T_i \cap S|$

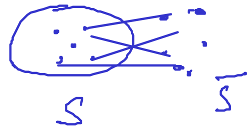
Example: Feedback vertex set
- Many problems can be expressed as a hitting-set problem, but not always immediately clear.

example 1: Minimum (s,t)-cut



Want a set of edges that hits every path from s,t

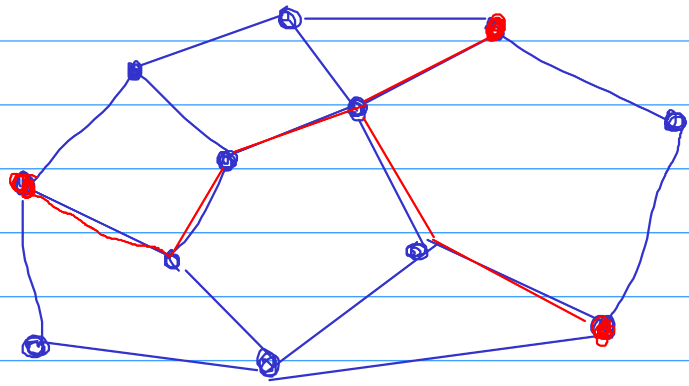




Example 2 Minimum Spanning tree / min Connected subgraph

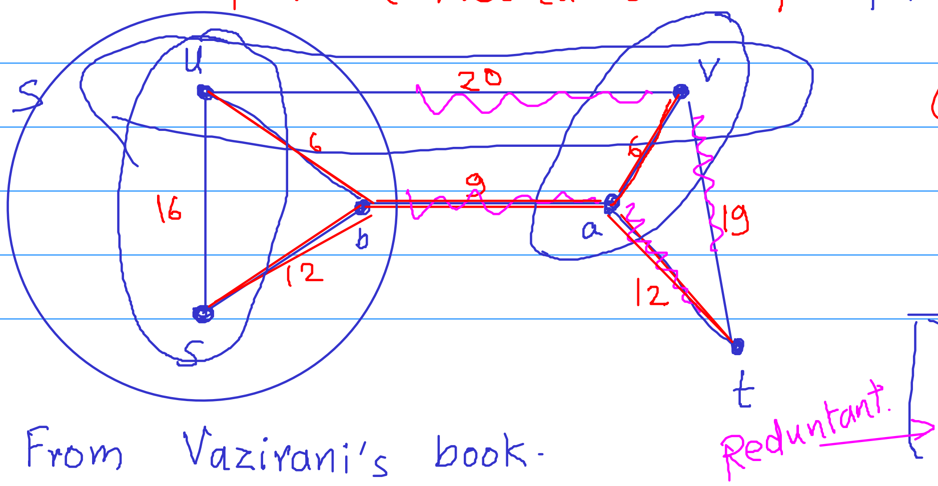
Want a set of edges that hits every set of cut-edges $\delta(S)$ for every $S \subseteq V$
$S \neq \emptyset$ $S \neq V$.

- Steiner tree : given a graph with edge weights and a set of terminal vertices, find a minimum weight tree s.t. all terminal vertices are connected to each other.



NP-hard

Steiner forest : given a graph with edge weights and pairs of terminal vertices
(s_1, t_1) (s_2, t_2) .... (s_k, t_k)
find the minimum weight subgraph where
s_i is connected with t_i for $i = 1, 2, \ldots, k$.



(s, t) (u, v)

$\underline{45}$

$33 + 20 = 53$

$C = \{v, a\}$

$\delta(C)$ disconnects s from t

(v,u) (v,t) (a,b) (a,t)

From Vazirani's book.

$C' = \{t\} \quad \delta(C') = \{(v,t), (a,t)\}$

- Shortest path and MST are special cases.

How to frame Steiner forest as a hitting set problem?

Def: Terminal separating cut: a cut $C \subseteq V$ s.t.
$s_i \in C$ and $t_i \notin C$ for some i
OR $s_i \notin C$ and $t_i \in C$ for some i

$$\delta(C) \leftarrow \{(a,b) : a \in C, b \in \bar{C}\}$$

Claim: A forest F is a Steiner forest for $(s_1,t_1), (s_2,t_2), \ldots, (s_k,t_k)$ — HW

iff $|F \cap \delta(C)| \geq 1$ for terminal separating cut C.

LP

$$\min \sum_{e \in E} x_e w_e$$

$x_e \geq 0 \quad e \in E$

$\sum_{e \in \delta(C)} x_e \geq 1$ for separating cut C.

Dual LP

$$\max \sum_{C} y_C$$

$y_C \geq 0$

$\underline{\sum_{C : e \in \delta(C)} y_C} \leq w_e$ for $e \in E$

Complementary Slackness

- $x_e > 0 \Rightarrow \sum_{C: e \in \delta(C)} y_C = w_e$ ← maintain exactly

  $\sum_C y_C < w_e \Rightarrow x_e = 0$

- $y_C > 0 \Rightarrow \underline{\sum_{e \in \delta(C)} x_e = 1}$ ← relax.

Primal Dual Algorithm (high level)

- Start with
  $F=$ empty set
  a dual feasible solution $y_C = 0$ for all C

- Loop
  Increase some dual variables

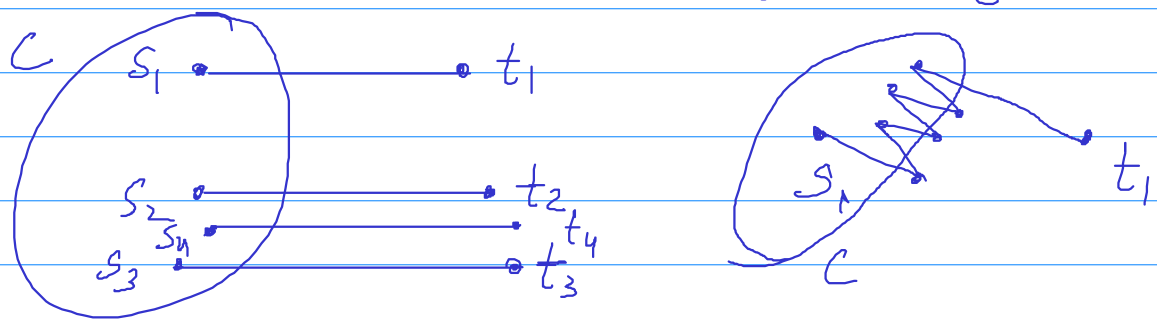
  If an edge e becomes tight
  $$\left(\sum_{C: e\in\delta(C)} y_C = w_e\right)$$
  then set $x_e = 1$
  put $e \in F$

Stop when every pair of terminals is connected in F.

Naive implementation of the algorithm will give approximation factor

$$\sum_{e\in\delta(C)} x_e = |F \cap \delta(C)| \le k \,?$$

max over all C.

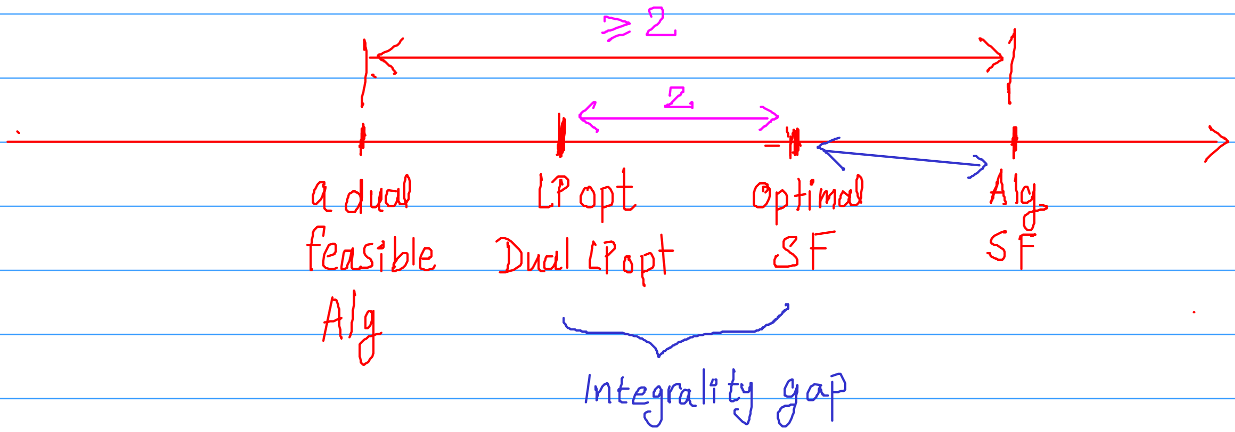


We need some better implementation and analysis.

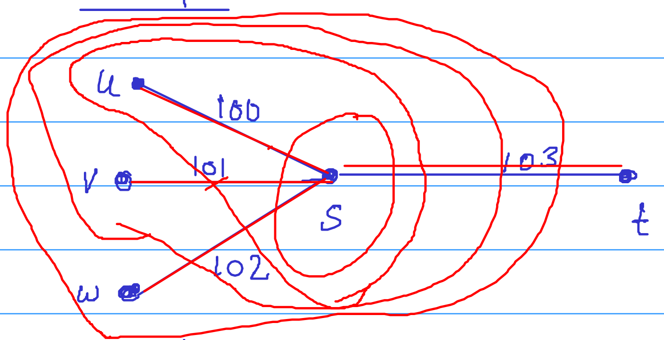
But we cannot hope to get an approx factor better than 2.

There is a problem instance where

Optimal steiner forest cost $\approx 2 \cdot$ LP-opt



Example :



Terminal pairs : (s, t)

$C = \{s\}$

$y_{\{s\}} \leftarrow 100$

$(u, s) \rightarrow F$

$\rightarrow \quad y_{\{s\}} + y_{s,w} \leq 100$

$y_{\{s\}} + y_{s,w} + y_{u,s} \leq 101$

$y_{\{s\}} + y_{u,s} \leq 102$

$y_{\{s\}} + y_{s,w} + y_{u,s} \leq 103$

$C = \{s, w\}$

$y_{\{s,w\}} \leftarrow$ cannot increase

$C \leftarrow \{u, s\}$

$y_{\{u,s\}} \leftarrow 1$

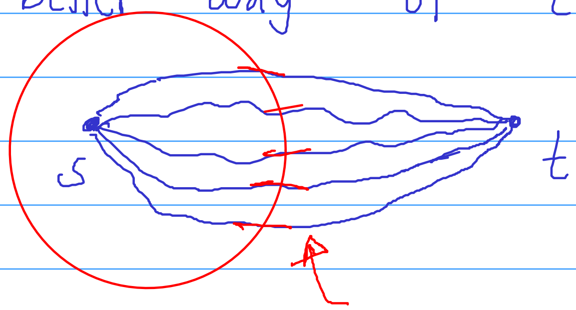
$(v, s) \rightarrow F$

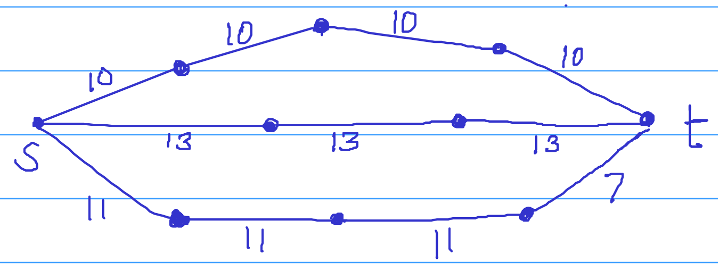
$(w, s) \rightarrow F$

$(s, t) \rightarrow F$

March 3, Thursday

Is there a better way of choosing dual variables





In the above graph we have only one terminal pair (s,t)

No matter in which order you increase the dual variables, at the end of the algorithm almost all edges will be tight (except one edge on each of the two longer paths).

Hence, we have to adopt the idea of Pruning.

Idea 1 : Pruning

At the end, throw out any edge that is not used in connecting any s_i with t_i

Note that in any forest, there is a unique path between s_i and t_i.

We can throw any edge not lying on any (s_i, t_i) path.

Now, how to choose which dual variable to increase.

- y_C can be increased only if $\delta(C)$ does not have tight edges.

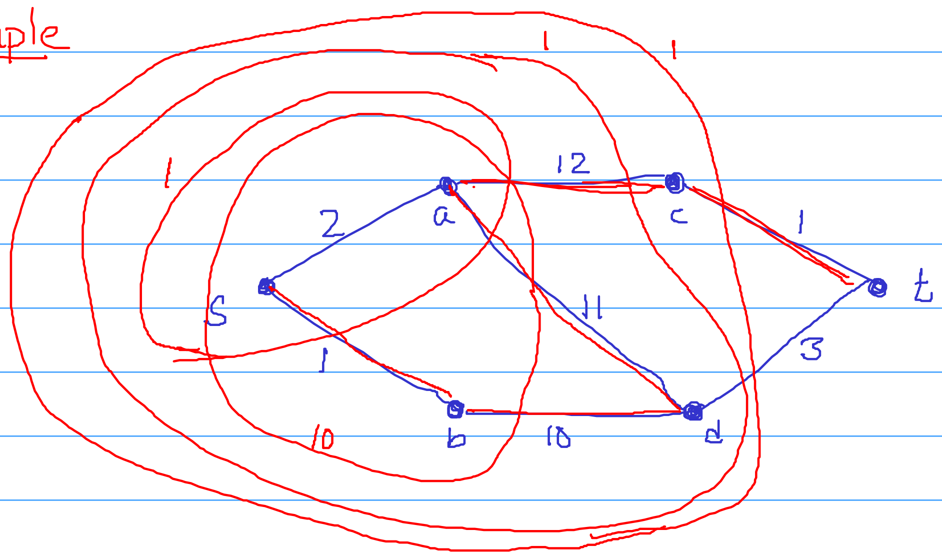
  Recall the dual constraint $\sum_{C: e \in \delta(C)} y_C = w_e$

- C should be such that any tight edge "does not cross" C.

  i.e., tight edges should be inside or outside C.

Can we keep picking such cuts C arbitrarily?

Example



Obtained path $= 35$

shortest path $= 14$

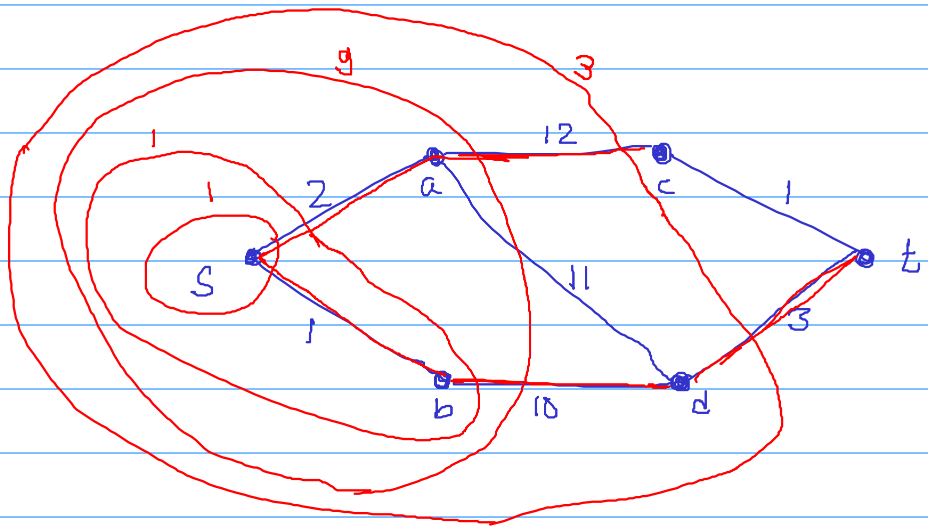
- Take $C = \{s, a, b\}$, increase y_C till 10.
  $e = (b, d)$ becomes tight.

- Take $C = \{s, a\}$, increase y_C till 1
  two edges (s, b) and (a, d) become tight.

- Take $C = \{s, a, b, d\}$ increase y_C till 1
  edge (a, c) becomes tight.

- Take $C = \{s, a, b, c, d\}$ increase y_C till 1
  edge (c,t) becomes tight.
  And we get a path between s and t.

Is there some natural order which gives a better result?



- Take $C = \{s\}$. Increase y_C till 1.
  edge (s,b) becomes tight.

- Take $C = \{s, b\}$. Increase y_C till 1.
  edge (s,a) becomes tight.

- Take $C = \{s, a, b\}$. Increase y_C till 9.
  edge (b,d) becomes tight.

- Take $C = \{s, a, b, d\}$. Increase y_C till 3.
  edges (a,c) and (d,t) become tight.

  s and t are connected.

  Pruning: throw out (s,a) and (a,c).
  We get the shortest path.

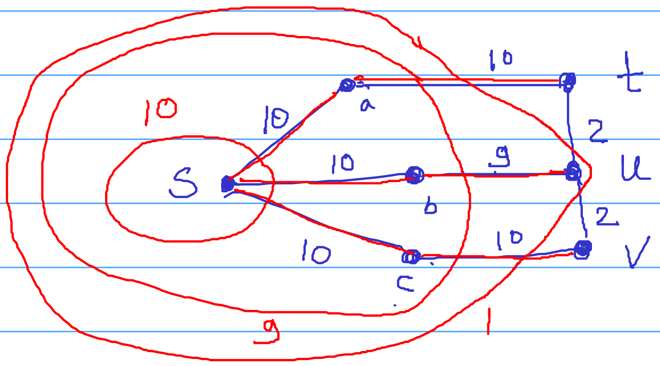
Start from $C = \{s\}$ and gradually enlarge it.

Idea2: Always take C to the minimal set so that $\delta(C)$ does not have tight edges.

Generalizable to more than one terminal pair?

Let's take an example with more terminal pairs

(s,t), (t,u), (s,u), (s,v), (t,v), (u,v)

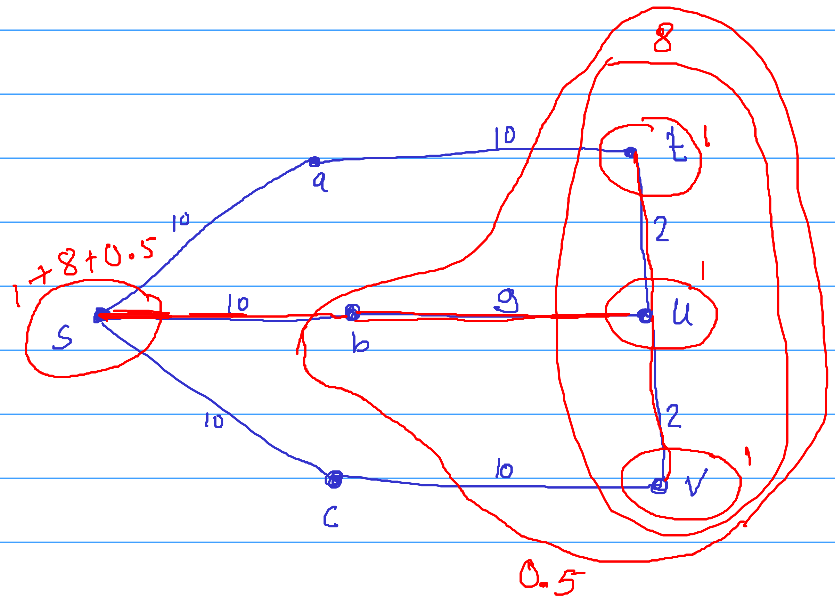


obtained steiner forest 59
optimal steiner forest 23

- Take $C = \{s\}$ Increase y_C till 10
  (s,a), (s,b), (s,c) become tight
- Take $C = \{s,a,b,c\}$. Increase y_C till 9
  (b,u) become tight.
- Take $C = \{s,a,b,u\}$. Increase y_C till 1
  (a,t), (c,v) become tight.

Stop.

Idea 3: Take cuts around all the terminals
Increase these dual variables simultaneously.



- Take $C_1 = \{s\}$ $C_2 = \{t\}$, $C_3 = \{u\}$, $C_4 = \{v\}$.

  Increase y_{C_1}, y_{C_2}, y_{C_3}, y_{C_4} simutaneously.

  Increment = 1
  (t,u) and (u,v) become tight.

- Take $C_1 = \{s\}$ $C_2 = \{t, u, v\}$

  Increase y_{C_1}, y_{C_2} simultaneously.
  Increment = 8
  (b,u) becomes tight.

- Take $C_1 = \{s\}$ $C_2 = \{t, u, v, b\}$

  Increase y_{C_1}, y_{C_2} simultaneously.
  Increment = 0.5
  (s,b) becomes tight.

All terminals are now connected.

Algorithm:

Invariants: F will be maintained as a forest with tight edges.

0. $F = \phi$, $y_C = 0$ for all C

1. Let $C_1, C_2, \cdots, C_q$ be those connected components in F which separate some terminal pair.

2. Increase $y_{C_1}, y_{C_2}, \ldots, y_{C_q}$ simultaneously till some edges becomes tight.

3. Add the tight edges to F (one by one, avoiding any cycle formation).

4. Stop if all terminal pairs are connected in F otherwise go to 1.

5. Pruning: throw out any edges that are not used in connecting any terminal pair.

Analysis:

via approximate complementary slackness does not work

for c with $y_c > 0$ $\quad \sum_{e \in \delta(c)} x_e = \gamma$?

Can we upper bound

In above examples, γ can be large.

But for some other cuts $\gamma = 1$.

Need some kind of averaging argument.

Different analysis:

Final dual value $D = \sum_{c} y_c$

Final primal value $P = \sum_{e \in F} w_e = \sum_{e \in F} \sum_{c : e \in \delta(c)} y_c$

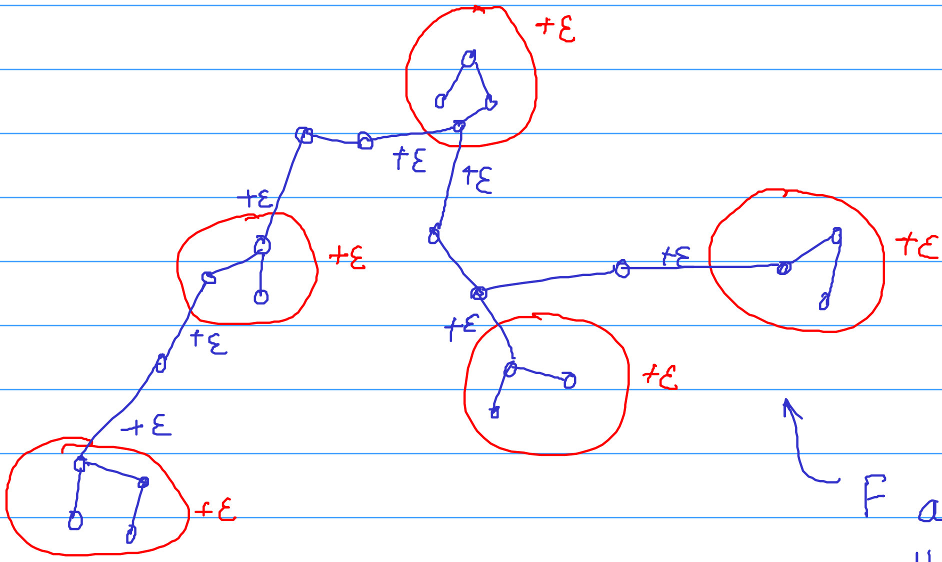
In any iteration, compare the increment in the above two quantities.

In an iteration, let's say we have $C_1, C_2, \ldots, C_q$ as the active cuts.

$y_{C_1}, y_{C_2} \ldots, y_{C_q}$ all increased by ε (say)

$$\Delta D = q\,\varepsilon$$

What about ΔP ?



F at the end of the algorithm

Claim: $\Delta P \leq 2(q-1)\varepsilon \leq 2\Delta D$

Hence, at the end,

$$P \leq 2D \leq 2 \cdot \text{opt steiner Forest}$$

2-approximation algorithm. ■

Questions:

- Take all pairs of vertices as terminal pairs.
  That means minimum spanning tree.

  Does this algorithm give exact or 2-approx?

- Take a single pair of terminals.
  That means shortest (s,t) path

  Does this algorithm give exact or 2-approx?

- Exercise Survivable Network Design

Given terminal pairs (s_1, t_1) (s_2, t_2) $\cdots$ (s_k, t_k)
and numbers $n_1, n_2, \cdots, n_k$

Find the minimum cost subgraph such that

s_i and t_i have at least n_i edge-disjont paths
for each i.