Manish Kumar (21044) : Assignment-IV : E0-225

(1)

Form-I:
$$\max\left(\sum_{e\in\delta^+(s)} f_e\right) \longrightarrow 1.1$$

$$\sum_{e\in\delta^-(v)} f_e = \sum_{e\in\delta^+(v)} f_e \quad \forall\ v\neq s,t \longrightarrow 1.2$$

$$f_e \leq u_e \quad \forall\ e\in E \longrightarrow 1.3$$

$$f_e \geq 0 \quad \forall\ e\in E \longrightarrow 1.4$$

Form-II:
$$\max\left(\sum_{p\in P} f_p\right) \longrightarrow 2.1$$

$$\sum_{p\in P:e\in p} f_p \leq u_e \quad \forall\ e\in E \longrightarrow 2.2$$

$$f_p \geq 0 \quad \forall\ p\in P \longrightarrow 2.3$$

→ Proof strategy:

Lemma 1: If $\{f_p\}_{p\in P}$ is a feasible solution to 'Form-II', then there is a feasible solution for 'Form-I' of the same cost.

Lemma 2: If $\{f_e\}_{e\in\delta^+(s)}$ is a feasible solution to 'Form-I', then there is a feasible solution to 'Form-II' of the same cost.

→ Lemma 1 and 2 implies that these two linear program always have equal optimal objective function value.

→ Proof of Lemma 1:

In form-II, $\sum_{p\in P:e\in p} f_p$ means the sum of the flows of all paths that uses the edge 'e'

We alternatively define it in terms of nodes as:

$$f(u,v) := \sum_{P \in \mathcal{P} : (u,v) \in P} f_P$$

inequality 2.2 implies it satisfy capacity constraints

To show conservation constraint, we pivot node v,
Then sum over all edges incident on it:

$$\sum_{u : (u,v) \in E} f(u,v) = \sum_{P \in \mathcal{P} : v \in P} f_P \quad \text{— (a)}$$

Same holds for edges coming out of v,

$$\sum_{w : (v,w) \in E} f(v,w) = \sum_{P \in \mathcal{P} : v \in P} f_P \quad \text{— (b)}$$

Equation (a) = (b) implies conservation for edge v in the network.

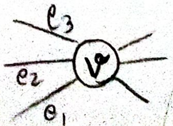
Diffrent s-t path passing through vertex v. Summing over these path gives equation (a) and (b)

→ Proof of Lemma 2:

This can be inferred from Flow decomposition theorem (FDT).

FDT: Let $f_1, \ldots, f_k$ be feasible flow while $P_1, \ldots, P_k$ be s-t path in a flow network.
Then $k \leq |E|$ and flow f_i sends positive flow only on the edges of P_i.

$$\sum_{e \in \delta^-(v)} f_e = \sum_{e \in \delta^+(v)} f_e$$



Through any vertex 'v' all the flow $\{f_i\}$ across it corresponds to some path $\{P_i\}$.

Thus sum of such s-t path amounts to $\sum_{P \in \mathcal{P} : e \in P} f_P \leq u_e$

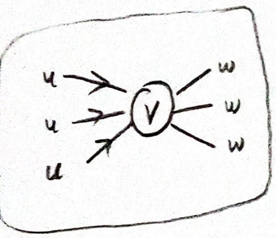
Also, $f_e \Rightarrow f_P$

② Maximum multi-commodity flow (MMCF)

$$\max \left(\sum_{i=1}^{K} f^{(i)} \right)$$

Capacity constraints : $\sum_{i=1}^{K} \underbrace{f_{uv}^{(i)}}_{\text{flow of commodity 'i' from vertex } u \text{ to } v.} \leq c(u,v)$; $\forall (u,v) \in V$

Conservation constraints :



$$\sum_{u \in V} f_{uv}^{(i)} = \sum_{w \in V} f_{vw}^{(i)} \quad ; \quad \forall i = \{1, \dots, K\} \quad \forall v \in V / \{s_i, t_i\}$$

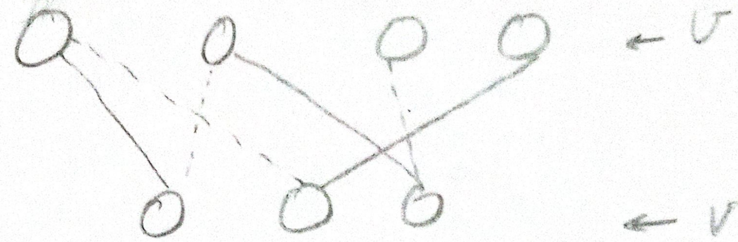
Positivity constraint :

$$f_{uv}^{(i)} \geq 0$$

The above linear program output maximum possible value of multicomodity flow.

③ Maximum Bipartile Matching (MBM)

(a) Given graph $G = (U \cup V ; E)$ where $E \subseteq U \times V$



Let 'M' be solution set for MBM

$x_e = 1$ iff edge 'e' $\in M$

objective : $\max\left(\sum_{e \in E} x_e\right)$

Constrain : for matching in Bipartile graph

$$\left\{\begin{array}{ll} \sum_{e \in \underbrace{\delta(v)}} x_e \leq 1 & \forall v \in (U \cup V) \\ x_e \geq 0 & \forall e \in E \\ x_e \leq 1 & \forall e \in E \end{array}\right.$$

Set of edges incident on vertex 'v'

③

(b)

Primal LP: (Part 3(a))	Dual LP
$\max\left(\sum_{e \in E} x_e\right)$	$\min\left(\sum_{v \in (U \cup V)} y_v\right)$
$\sum_{e \in \delta(v)} x_e \leq 1 \; ; \forall v \in (U \cup V)$	$y_v + y_u \geq 1$ $e = (u, v)$ and $\forall e \in E$
$\left.\begin{array}{l} x_e \geq 0 \\ \boxed{x_e \leq 1} \end{array}\right\} \forall e \in E$ This is not neccessary	$y_v, y_u \geq 0$

→ Meaning of dual variable 'y_v': It represents if a vertex $v \in (U \cup V)$ is included in the solution set or not.

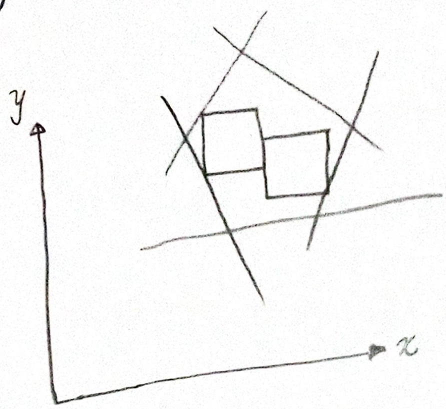
→ Objective function represents the minimum possible set of vertices that satisfy the associated Constrain.

→ Dual problem represents minimum vertex cover in the given Bipartile graph.

→ It can be inferred from the constraint $y_v + y_u \geq 1$. This makes sure that for every edge in the graph, atleast one of its vertex is included.

→ Minimization of the objective function ensures that we pick fewest vertices still covering all edges.

④



$a_i x + b_i y \le c_i \quad \forall i = \{1, \dots, n\}$

intersection of above lines form feasible region.

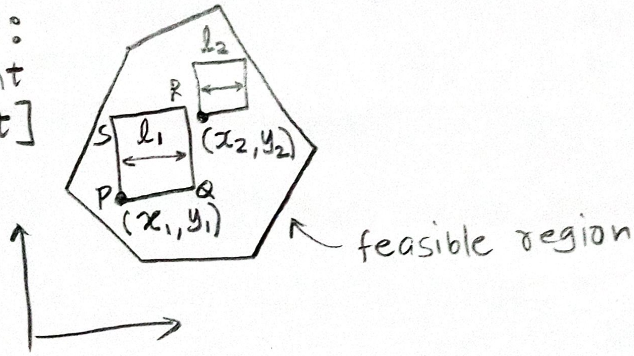
Since, squares are axis aligned,

Hence, we consider two cases.

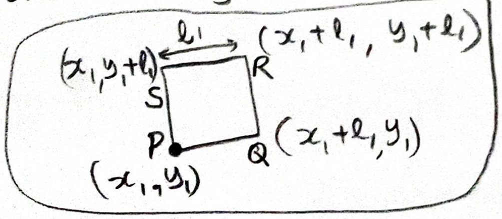
Both cases gives a linear program.

We compare the output of both cases and declare maximum among them as final output.

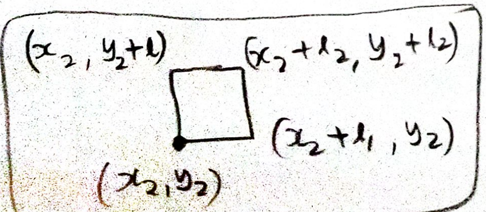
Case-I: [Left-Right alignment]



Let sides of squares are l_1 and l_2 as shown in the above diagram.



The left square lies in feasible region. [Constrain-I]



The right square shall lie inside feasible rgion, and its coordinate shall begin just after the left square finishes.

[Constraint - II and III]

objective: sum of areas $= 4l_1 + 4l_2$

$$\boxed{\max(4l_1 + 4l_2)} \longrightarrow (1)$$

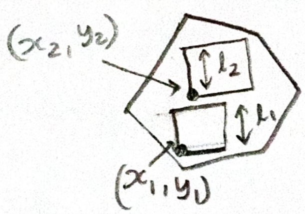
Square PQRS
$$\begin{cases} a_i x_1 + b_i y_1 \leq c_i & \longrightarrow (2.1) \\ a_i(x_1 + l_1) + b_i(y_1) \leq c_i & \longrightarrow (2.2) \\ a_i(x_1) + b_i(y_1 + l_1) \leq c_i & \longrightarrow (2.3) \\ a_i(x_1 + l_1) + b_i(y_1 + l_1) \leq c_i & \longrightarrow (2.4) \end{cases}$$

orientation of two squares
$$\{ x_1 + l_1 \leq x_2 \longrightarrow (3)$$

For other square
$$\begin{cases} a_i x_2 + b_i y_2 \leq c_i & \longrightarrow (4.1) \\ a_i(x_2 + l_2) + b_i y_2 \leq c_i & \longrightarrow (4.2) \\ a_i(x_2) + b_i(y_2 + l_2) \leq c_i & \longrightarrow (4.3) \\ a_i(x_2 + l_2) + b_i(y_2 + l_2) \leq c_i & \longrightarrow (4.4) \end{cases}$$

$$l_1, l_2 \geq 0 \longrightarrow 5$$

Case-II: Top and down orientation



All equation from (1) to (5) remain same except eqn (3)

Replace eqn (3) by

$$y_1 + l_1 \leq y_2 \quad (3')$$