Lecture 6A

Stochastic & Robust Optimization

Facility location problem

A company need to decide where to open facilities to serve client efficien

$F = \{1, \dots, n\}$: candidate facility locations.

Opening a facility i incurs a one-time cost f_i and capacity $u_i \geq 0$

$C = \{1, \dots, m\}$: set of clients

Each client j has demand $d_j \geq 0$ must be assigned to an open facility

Assign client j to facility i, incurs an associated cost C_{ij}

Goal : min sum of facility opening and client assigning cost.

LP model

Var : y_i = indicates if facility i is open $\forall i \in F$ ≥ 0

x_{ij} = indicates if client j is assigned to facility i

Objective $\min \sum_{i \in F} f_i y_i + \sum_{j \in C} \sum_{i \in F} C_{ij} x_{ij}$

Constraint Every client is assigned to facility

$\sum_{i \in F} x_{ij} = 1 \qquad \forall j \in C$

Client is only assigned to open facility

$x_{ij} \leq y_i \qquad \forall i \in F, \ \forall j \in C$

capacity constraint

$\sum_{j \in C} d_j x_{ij} \leq u_i y_i \qquad \forall i \in F$

Integrity constraints

$0 \leq x_{ij}, y_i \leq 1$

Client demands $\{d_j\}_{j \in C}$ are often uncertain

1) demand is modeled by a probability distribution (stoch. opt)

2) Specifying uncertainty set representing possible outcome for d_js

ex. $D = \{ (d_j)_{j \in C} : d_j \in [d_j^0 - \Delta_j, d_j^0 + \Delta_j] \ \forall j \in C \}$

(Robust optimization)

Stochastic Optimization

Scenario presentation

How is the demand distribution specified

Distribution specified by a finite collection A of scenarios

Each scenario $s \in A$ specifies a realization of client demands (d_j^s)

and has a probability of occurrence P_s

$$\sum_{s \in A} P_s = 1 \qquad P_s \geq 0 \quad \forall s \in A$$

Distribution : with probability P_s, scenario S occurs

i.e. demand = $(d_j^{\ s})_{j \in C}$

Decisions are made in 2 stages :

1) First stage : decisions are made given only uncertain info
decisions about y_i's (where to open facilities)
given scenario representation

2) Second stage : decision made after a scenario is realized
We know $\{d_j^s\}_{j \in C}$ and decide how to assign client to open facilities and possibly have option of open extra facilities / augulatory capacity incurring costs

Modeled by X_{ij}^s $\forall s \in A$, $\forall i \in F$, $j \in C$
in scenario S, if client j is to facility i

This is called 2 stage stochastic optimization

Farmer problem deterministic LP (yields are determinstic)

Farmer raises wheat, corn, sugar between his 500 acres farm requirement for a corp (for farm use) can be met by farm-yield or by purchasing the corp. Any surplus is sold.

Assume yields are deterministic.

	wheat	corn	sugarbeet
yield (ton/acre)	2.5	3	20
planting cost	150	230	260
sales price ($/ton)	170	150	36 if < 6000T, 10 if 6000T
purchase price	238	210	
min req	200	240	

Var x_1 : # acres of wheat planted x_2: corn x_3 = sugarbeet

y_1: # tons of wheat bought y_2: corn

w_1: # tons of wheat sold w_2 = corn w_3 = SB at high w_4 = SB at low

Obj; $\max (170w_1+150w_2+36w_3+10w_4) - \overbrace{(150x_1+230x_2+260x_3)}^{\text{plant cost}} - \overbrace{(238y_1+210y_2)}^{\text{buying cost}}$

constraints $x_1+x_2+x_3 \le 500$

(wheat) $0 \le w_1 \le 2.5x_1+y_1-200$

(corn) $0 \le w_2 \le 3x_2+y_2-240$

(SB) $w_3+w_4 \le 20x_3$, $w_3 \le 6000$ all vars ≥ 0

Optimal sol = 118,600

acre planted	120 wheat	80 corn	300 SB
yield	300	240	6000
purchase	—	—	—
sales	100	—	6000

opt sol satisfies wheat + corn requir. Then devotes avg in ↓ profit

SB profit ↑ price = 20x36−260=400

SB profit ↓ price = 20x10−260=−60

wheat = 2.5x170−150=275

corn = 3x 150−230=220

Lecture 6B

Hedging against uncertainty

Due to uncertainty, we won't be able to take optimal decisions every scenario. There is cost associated with not having precise info.

2 stage stochastic LP farmers prob with uncertain yield

	Wheat (ton/acre)	corn	SB	
Low yield ($0.8 \times$ Avg)	2	2.4	16	
avg yield	2.5	3	20	
high yield ($1.2 \times$ Avg)	3	3.6	24	prob = 1/3 each

Goal: max expected profit

Decision var: same

Stochastic var: Scenarios $\{L, A, H\}$, $\forall s \in A$

$$\left\{ \begin{array}{l} y_1^s, y_2^s \text{ tons of wheat and corn bought} \\ w_1^s, w_2^s, w_3^s, w_4^s \text{ tons of wheat, corn, SBhigh, SBlow sold} \\ \text{Let } P_s = \text{prob of occurrence of scenario } S \ \forall s \in A \end{array} \right\} = \frac{1}{3}$$

Objective max $\underbrace{-(150x_1 + 230x_2 + 260x_3)}_{\text{stage 1 planting cost}} + \sum_{s \in \{L,A,H\}} P_s \underbrace{[170w_1^s + 150w_2^s + 36w_3^s + 10w_4^s]}_{\text{scenarios revenue}} \underbrace{- (238y_1^s + 210y}_{\text{scenario S purchase}}$

constraints: $x_1 + x_2 + x_3 \leq 500$ (Area)

$$\forall s \in \{L, A, H\} \left\{ \begin{array}{l} \text{Let yield } i^s \text{: yield of crop } i \text{ in scenario } S_i{}^2 \\ \quad (\text{e.g. yield}^H = 3.6) \\ 0 \leq w_1^s \leq \text{yield}_1^s x_1 + y_1^s - 200 \\ 0 \leq w_2^s \leq \text{yield}_2^s x_2 + y_2^s - 240 \\ w_3^s + w_4^s \leq \text{yield}_3^s x_3, \quad w_4^s \leq 6000 \qquad (3, S) \end{array} \right.$$

Stochastic LP

max $x_1 + x_2 + x_3 \leq 500$ (Area)

$0 \leq w_1^s \leq \text{yield}_1^s x_1 + y_1^s - 200$ (1, S)

$0 \leq w_2^s \leq \text{yield}_2^s x_2 + y_2^s - 240$ (2, S)

$w_3^s + w_4^s \leq \text{yield}_3^s x_3$, $w_4^s \leq 6000$ (3, S)

all vars ≥ 0

Optimal solution to deterministic LP with:

Low yields total profit = 59,950 | High yields total profit = 167,667

	Low: wheat	Low: corn	Low: SB	High: wheat	High: corn	High: SB
Acres planted	100	25	375	183.33	66.67	250
Yield	200	60	6000	550	240	6000
Purchase	—	180	—	—	—	—
Sales	—	—	6000	350	—	6000

Expected Value of Perfect Info (EVPI)

Q: What is the best one would do if farmer had perfect info about which scenario will occur at the time of 1^{st} stage decisions i.e. at the time of planting?

A: Then farmer would implement opt. sol for scenario that will occur, and hence can expect profit $= \frac{1}{3}(118{,}600 + 59{,}950 + 167{,}667) = \$115{,}406$

$\text{EVPI} = \text{Diff} = \$115{,}406 - \underbrace{\$105{,}390}_{\text{opt value of stoch. LP}} = \$7{,}016$

EVPI = |(expected value of opt value of deterministic LP for each scenario − opt value of stochastic LP)|

Value of stochastic soln (VSS) opt value of stochastic LP = \$108,310

Here is a simple, but potential subsptional solution:

1) look at expected yields : $\forall$ crop i $\sum_{s \in \{L,A,H\}} P_s \cdot \text{yield}_i^s$

 then W = 2.5 , C = 3, SB = 20

2) Solve deterministic LP with expected yields to get planting decision i.e. first-stage decisions gives W = 120, C = 80, SB = 300

3) Plug in these planting decisions in stochastic LP and refer corresponding second stage decisions to get feasible sol.

 $\forall s \in \{L,A,H\}$, $0 \le W_i^s \le \text{yield}_i^s x_i + y_i^s - 200$

 compute obj value of this feasible solution to this stochastic LP

 Here, we get \$107,240

 VSS = 108390 − 107240 = 1150

Lecture 7A

<u>Hospital Staffing problem</u>

Hospital ER has to plan staffing for Sunday in face of uncertain patient load.
Regular-time nurses: work at \$300/day, can handle ≤ 10 patients/day
After observing sunday's patient load, can bring in at most 5 "rush-nurses" at a cost of \$400/nurse. Rush-nurse's can handle ≤ 10 patients/day

Scenarios for Sunday's patient load:

Scenario	1	2	3	4	
Load (# patients)	90	100	100	80	each scenario w/ prob 0.25

Goal: min total expected cost

<u>Decision var</u> 1st stage vars: X_{Reg}: # reg nurses called in knowing scenario distrib
2nd stage vars: X^s_{rush}: # rush-nurses called in on Sunday in Scen. S

Objective: min $300X_{Reg} + \sum_{s=1}^{4} \frac{1}{4} \cdot 400 X^s_{rush}$

constraints: $\forall$ scenario $S = 1,2,3,4$ $X^s_{rush} \leq 5$

$$10(X_{Reg} + X^1_{rush}) \geq 90$$
$$10(X_{Reg} + X^2_{rush}) \geq 100$$
$$10(X_{Reg} + X^3_{rush}) \geq 100$$
$$10(X_{Reg} + X^4_{rush}) \geq 80 \qquad \text{all vars} \geq 0$$

Opt value of stochastic LP is \$2900 ($X^*_{reg} = 8, X^{*1}_{rush} = 1, X^{*2}_{rush} = 2, X^{*3}_{rush} = 3, X^{*4}_{rush}$

<u>Expected Value of Perfect Infor</u> (EVPI)

If we knew each scenario s that would occur, we would call in $\frac{Load^s}{10}$ regular nurse
This incurs expected cost $= \frac{1}{4} \cdot 300\left(\frac{90}{10} + \frac{100}{10} + \frac{100}{10} + \frac{80}{10}\right) = 2775$
so EVPI $= 2900 - 2775 = \$125$

<u>Value of stochastic soln</u> (VSS)

Look at expected load $370/4 = 92.5$
1) Solve deterministic LP with expected load to get 1st stage decisions $X_{Reg} = 92.5/10 = 9.2$
2) Plug in X_{Reg} in 2 stage LP to get 2nd stage decisions. $X^1_{rush} = 0$, $X^2_{rush} = X^3_{rush} = 0.75$, $X^4_{rush} = C$
This solution has cost $= 300 \times 9.25 + \frac{1}{4} \cdot 400(0 + 0.75 + 0.75 + 0) = 2925$
VSS $= |2925 - 2900| = \$25$

Robust Optimization

consider a generic LP

$$\max \quad c^T x$$
$$\text{s.t.} \quad Ax \le b \qquad\qquad A: m\times n \text{ matrix}$$
$$\underbrace{l \le x \le u}_{\text{bound constraints}} \qquad\qquad l_j, u_j \text{ could be -ve}$$

A robust optimization problem is specified by an uncertainty set A_i for every row $i = 1, \dots, m$ of A, where

$$A_i = \{(\tilde{a}_{ij})_{j=1}^n : \tilde{a}_{ij} \in \mathbb{R}\} \subseteq \mathbb{R}^n$$

Each vector $(\tilde{a}_{ij})_{j=1}^n$ is a realization of coefficients for i^{th} inequality of $Ax \le b$

Robust Optimization problem arising from A_i, $i = 1, \dots, n$

$$\max \ c^T x$$
$$\text{s.t. } \forall i = 1, \dots m \quad \forall (\tilde{a}_{ij})_{j=1}^n \in A_i, \ \sum_{j=1}^n \tilde{a}_{ij} x_j \le b_i$$
$$l \le x \le u$$

Example

$n=4$, uncertainty set for 1^{st} inequality of $Ax \le b$ is

$$A_1 = \begin{cases} \tilde{a} = (\tilde{a}_{11}, \tilde{a}_{12}, \tilde{a}_{13}, \tilde{a}_{14}) \\ \tilde{a}_{11} \in [3, 5] \\ \tilde{a}_{12} \in [90, 120] \\ \tilde{a}_{13} \in [3.5, 5.2] \\ \tilde{a}_{14} \in [-2.8, 2.8] \end{cases}$$

Another uncertainty set

$$A_1(2) = \{\tilde{a} = (\tilde{a}_{11}, \tilde{a}_{12}, \tilde{a}_{13}, \tilde{a}_{14}) \in A_1 : \text{at most 2 } \tilde{a}_{ij}\text{'s deviate from mid-point of their interval}\}$$

$\tilde{a} = (3, 95, 4, 0)^T \in A$, but not in $A_1(2)$

Soysters set

Look at uncertainty set

$$A_i^{soy} = \{ \tilde{a}_i = (\tilde{a}_{ij})_{j=1}^n : \tilde{a}_{ij} \in [a_{ij} - \Delta_{ij}, a_{ij} + \Delta_{ij}] \ \forall j = 1, \dots, n \}$$

So correct robust programs

$$\max \ C^T x$$
$$\text{s.t. } l \leq x \leq u$$
$$\sum_{j=1}^{n} \tilde{a}_{ij} x_j \leq b_i \quad ①$$
$$\forall i = 1, \dots, m \quad \forall \tilde{a}_i = (\tilde{a}_{ij})_{j=1}^n \in A_i^{soy}$$

suppose $\tilde{A}^{soy} = \{ \tilde{a}_{ij} \in \{ a_{ij} - 1, a_{ij}, a_{ij} + 1 \} \ \forall j = 1, \dots, n \}$ set has 2^n vectors

Goal: convert (Rob-P) as an equivalent compact LP

Rewrite ① as $\max_{\tilde{a}_i \in A_i} \sum_{j=1}^{n} a_{ij} x_j \leq b_i = \sum_{j=1}^{n} a_{ij} x_j + \max_{\tilde{a}_i \in A_i^{soy}} \left(\sum_{j=1}^{n} (\tilde{a}_{ij} - a_{ij}) x_j \right) \leq b_i$

This is maximized by setting

$$\tilde{a}_{ij} = \begin{cases} a_{ij} + \Delta_{ij}, & \text{if } x_j \geq 0 \\ a_{ij} - \Delta_{ij}, & \text{if } x_j < 0 \end{cases}$$

So we can write ① as

$$\sum_{j=1}^{n} a_{ij} x_j + \underbrace{\sum_{j, x_j \geq 0} \Delta_{ij} x_j + \sum_{j, x_j < 0} \Delta_{ij} x_j}_{\sum_{j=1}^{n} \Delta_{ij} |x_j|} \leq b_i$$

$$\Rightarrow \sum_{j=1}^{n} a_{ij} x_j + \sum_{j=1}^{n} \Delta_{ij} |x_j| \leq b_i$$

$$\xRightarrow{\text{linearize}} \quad y_j \geq x_j, \ y_j \geq -x_j, \ \sum_{j=1}^{n} a_{ij} x_j + \sum_{j=1}^{n} \Delta_{ij} y_j \leq b_i$$

Compact LP: $\max \ C^T x$

$$\text{s.t. } l \leq x \leq u$$
$$y_j \geq x_j, \ y_j \geq -x_j \quad \forall j = 1, \dots, n$$

Equivalent to (Prob-P)

$$\sum_{j=1}^{n} a_j x_j + \sum_{j=1}^{n} \Delta_{ij} y_j \leq b_i \quad \forall i = 1, \dots, n$$

Lecture 7B

Bertsimas-Sim model

We introduce products Γ_i $\forall i = 1, \dots, m$ and uncertainty set for row i

$A_i(\Gamma_i) = \{ \tilde{a}_i = (\tilde{a}_{ij})_{j=1}^n : \tilde{a}_{ij} \in \{a_{ij} - \Delta_{ij}, a_{ij} + \Delta_{ij}\}$ $\forall j = 1, \dots, n\}$

and at most Γ_i of $\tilde{a}_{ij}$'s deviate from a_{ij}s $\}$

$\uparrow \Gamma_i$: more robust, more conservative model

Γ_i : nicely allows us to control robustness vs. conservation

Robust problem $\max c^T x$ (BS-Rob-P)

s.t. $l \le x \le u$

$\forall \tilde{a}_i : (\tilde{a}_{ij})_{j=1}^n \in A_i(\Gamma_i) \sum_{j=1}^n \tilde{a}_{ij} x_j \le b_i$

Converting (BS-Rob-P) to a compact LP (pg 69-71)

step 1 rewrite LP $= \sum_{j=1}^n a_{ij} x_j + \max_{\tilde{a}_i \in A_i(\Gamma_i)} \sum_{j=1}^n (\tilde{a}_{ij} - a_{ij}) x_j \le b_i$, $B_i(x, \Gamma_i) = \max \sum_{j=1}^n (\tilde{a}_{ij} - a_{ij}) x_j$

difficult to write a closed-form expression for $B_i(x, \Gamma_i)$ because at most Γ_i $\tilde{a}_{ij}$'s can deviate from a_{ij}

An opt sol to $B_i(x, \Gamma_i)$ includes choosing a set S_i of at most Γ_i indices from $\{1, \dots, n\}$ = indices for which $\tilde{a}_{ij}$ can deviate from a_{ij} and then setting

$$\tilde{a}_{ij} = \begin{cases} a_{ij} & , j \notin S_i \\ a_{ij} + \Delta_{ij} , & j \in S_i, x_j \ge 0 \\ a_{ij} - \Delta_{ij} , & j \in S_i, x_j < 0 \end{cases} \quad \text{so } B_i(x, \Gamma_i) = \max_{\substack{S_i \subseteq \{1, \dots, n\} \\ |S_i| \le \Gamma_i}} \sum_{j \in S_i} \Delta_{ij} |x_j|$$

step2 Write $B_i(x_i, \Gamma_i)$ as an LP.

Introduce decision variable z_{ij} $\forall j = 1, \dots, n$ to indicate

if $j \in S_i$ $\quad z_{ij} = 1 = j \in S_i$

$z_{ij} = 0 = j \notin S_i$

we can see that
$$B_i(x, \Gamma_i) = \max \sum_{j=1}^n \Delta_{ij} |x_j| z_{ij}$$
$$\text{s.t. } \sum_{j=1}^n z_{ij} \le \Gamma_i$$
$$0 \le z_{ij} \le 1 \quad \forall j = 1, \dots, n$$

$$\max \sum_{j=1}^n \Delta_{ij} |x_j| z_{ij}$$
$$\text{s.t. } \sum_{j=1}^n z_{ij} \le \Gamma_i$$
$$0 \le z_{ij} \le 1$$

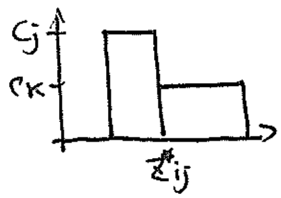
<u>Claim</u> there is always an opt sol to (Bi-LP) where all z_{ij} are 0 or 1

<u>proof</u> Let $(z^*_{ij})_{j=1}^{n}$ be an opt soln that $\sum_{j=1}^{n} z^*_{ij} = r_i$. r_i is an integer.

So if $z^*_{ij} \in (0,1)$, then there exists $k \neq j$ s.t. $z^*_{ik} \in [0,1]$

Let $c_j = \Delta_{ij}|x_j|$

$c_k = \Delta_{ik}|x_k|$



we want to make sure width of both bars is certain amount so we can adjust width of shorter rectangle until one becomes 0, the other

If $c_j \geq c_k$, $\uparrow z^*_{ij}$, $\downarrow z^*_{ik}$ until $z^*_{ij} = 1$, $z^*_{ij} = 0$

$c_j < c_k$, $\downarrow z^*_{ij}$, $\uparrow z^*_{ik}$ until $z^*_{ik} = 1$ or $z^*_{ij} = 0$

By repeatedly making these pertributions, we will get to an opt sol where all z_{ij}s $\in \{0,1\}$ $\Rightarrow$ $OPT_{Bi\text{-}LP} = B_i(x, r_i)$

$$(Bi\text{-}LP)\quad \max \sum_{j=1}^{n} \Delta_{ij}|x_j| z_{ij}$$
$$\text{s.t.} \sum_{j=1}^{n} z_{ij} \leq r_i$$
$$0 \leq z_{ij} \leq 1 \qquad \forall j = 1, \dots, n$$

$\sum_{j=1}^{n} a_{ij}x_j + B_i(x, r_i) \leq b_i$ so we get equivalent constraint

$$OPT_{Bi\text{-}LP} \leq b_i - \sum_{j=1}^{n} a_{ij}x_j \qquad (2)$$

Step 3 take duplex of Bi-LP and plug it into (2)

$$\min\ r_i\alpha_i + \sum_{j=1}^{n} p_{ij}$$
$$\text{s.t.}\ \alpha_i + p_{ij} \geq \Delta_{ij}|x_j| \qquad \forall j = 1, \dots, n$$
$$\alpha_i, p_{ij} \geq 0 \qquad \forall j = 1, \dots, n$$

So $r_i\alpha_i + \sum_{j=1}^{n} p_{ij} \leq b_i - \sum_{j=1}^{n} a_{ij}x_j$ since $OPT_{Bi\text{-}LP} = \alpha_i r_i + \sum_{j=1}^{n} p_{ij}$

Step 4 linearize $|x_j|$ by introducing var y_j and constants

$y_j \geq x_j$, $y_j \geq -x_j$, replace (2) by $\alpha_i + p_{ij} \geq \Delta_{ij} y_j \quad \forall j = 1, \dots, n$

Final LP equivalent to (BS-Rob-P):

$$\max\ c^T x \quad \text{s.t.}\ l \leq x \leq u,\ y_j \geq x_j,\ y_j \geq -x_j \quad \forall j = 1, \dots, n$$
$$r_i\alpha_i + \sum_{j=1}^{n} p_{ij} \leq b_i - \sum_{j=1}^{n} a_{ij} - x_j \qquad \forall i = 1, \dots, n$$
$$\alpha_i + p_{ij} \geq \Delta_{ij} y_j,\ \alpha_i \geq 0,\ p_{ij} \geq 0 \qquad \forall j = 1, \dots, n$$

Lecture 8A

Example Resource constraint problem

product	m/c 1 (hrs)	m/c 2	revenue
1	11	4	300
2	7	6	260
3	6	5	220
4	5	4	180
Availability	700	500	

LP : $\max 300x_1+260x_2+220x_3+180x_4$

m/c 1 $\quad$ s.t. $11x_1+7x_2+6x_3+5x_4 \le 700$ $\quad$ (1)

m/c 2 $\quad$ $4x_1+6x_2+5x_3+4x_4 \le 500$ $\quad$ (2)

$x_1, \dots, x_4 \ge 0$ $\quad$ (3)

Suppose times on m/c 1 may vary by ± 1 hr and times of at most 2 products may deviate. We have uncertainty set

$A_1(2) = \{(\tilde{a}_{11}, \tilde{a}_{12}, \tilde{a}_{13}, \tilde{a}_{14}): \tilde{a}_{11} \in [10,12], \tilde{a}_{12} \in [6,8], \tilde{a}_{13} \in [5,7], \tilde{a}_{14} \in [4,6],$
at most 2 $\tilde{a}_{ij}$ may deviate from mid points of their ranges$\}$

(Bre S-P) $\max 300x_1+260x_2+220x_3+180x_4$

s.t. $\sum_{j=1}^{4} \tilde{a}_{ij} x_j \le 700 \quad \forall (\tilde{a}_{11}, \tilde{a}_{12}, \tilde{a}_{13}, \tilde{a}_{14}) \in A_1(2)$ $\quad$ (1')

$4x_1+6x_2+5x_3+4x_4 \le 500, \quad x_1, \dots, x_4 \ge 0$

Step 1 Rewrite (1') with a non-expression on LHS

$$11x_1+7x_2+6x_3+5x_4+\underbrace{\max[(\tilde{a}_{11}-11)x_1+(\tilde{a}_{12}-7)x_2+(\tilde{a}_{13}-6)x_3+(\tilde{a}_{14}-5)x_4]}_{B_i(x,2)} \le 700$$

$\Rightarrow B_1(x,2) \le 700-11x_1-7x_2-6x_3-5x_4$ $\quad$ (1'')

Step 2 write $B_1(x,2)$ as an LP

$B_1(x,2) = \max_{S \subseteq \{1,\dots,4\},\ |S| \le 2} \sum_{j \in S} x_j = \max \sum_{j=1}^{4} x_j z_{ij}$ $\quad$ (B_1-LP)

s.t. $\sum_{j=1}^{4} z_{ij} \le 2$, $\quad 0 \le z_{ij} \le 1 \quad \forall j=1,\dots,4$

(1'') $= OPT_{B_1\text{-}LP} \le 700-11x_1-7x_2-6x_3-5x_4$ $\quad$ (1''')

Dual of B_1-LP $= \max 2\alpha_1 + \sum_{j=1}^{4} r_{ij}$ $\quad$ (B_i-D)

s.t. $\alpha_1 + P_{1j} \ge x_j$, $\alpha_1 \ge 0$, $P_{1j} \ge 0 \quad \forall j=1,\dots,4$

Step 3 replace (B_i-LP) by its dual and plug in (1''')

(1''') $= 2\alpha_1 + \sum_{j=1}^{4} \le 700-11x_1-7x_2-6x_3-5x_4$

Final LP $\quad \max 300x_1+260x_2+220x_3+180x_4$

s.t. $2\alpha_1 + \sum_{j=1}^{4} \le 700-11x_1-7x_2-6x_3-5x_4$

$\alpha_1 + P_{ij} \ge x_j$, $\quad \alpha_1 \ge 0$, $P_{ij} \ge 0 \quad \forall i,j$

$4x_1+6x_2+5x_3+4x_4 \le 500$, $\quad x_1, \dots, x_4 \ge 0$

Chapter 4 Integer programming

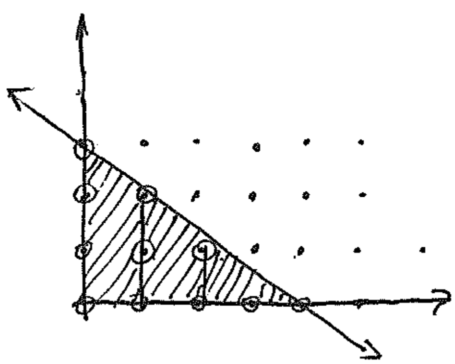
Defn. An integer program (IP) is an LP with additional constraints requiring some (or all) var to be integer valued.

pure IP : all vars are integers
mixed IP : some subset of vars are integers

Example

$$\begin{aligned} \max \quad & 3x_1 + 4x_2 \\ \text{s.t.} \quad & x_1 + x_2 \le 4 \\ & x_1 + 2x_2 \le 6 \\ & x_1, x_2 \ge 0 \\ & x_1, x_2 \text{ int } (*) \\ & x_1 \text{ integer } (**) \end{aligned}$$

pure LP : LP + *
mixed LP : LP + **



////. feasible region to LP
• feasible region to pure IP
— feasible region to mixed IP

Lecture 8B

Foundations & Modeling

Example: investment problem

WatCo is considering in 4 projects. (Yes / No decision). Max return.

	P_1	P_2	P_3	P_4	cash avail
year 1	40	20	25	80	150
year 2	10	30	30	40	80
Final return	100	90	120	160	

Decision var: (binary) $x_i = \begin{cases} 1 & \text{if invest in } P_i \\ 0 & \text{o/w} \end{cases}$

Objective: $\max 100x_1 + 90x_2 + 120x_3 + 160x_4$

constraints: (year 1) $40x_1 + 20x_2 + 25x_3 + 80x_4 \leq 150$

(year 2) $10x_1 + 30x_2 + 30x_3 + 40x_4 \leq 80$

$x_1, \ldots, x_4 \in \{0,1\} \;=\; 0 \leq x \leq 1$ x is integer

- Cardinality constraints: we can invest in at most 3 projects
  $x_1 + x_2 + x_3 + x_4 \leq 3$
- Binary Logic: P_1 or P_2 must be invested in
  $(x_1 = 1)$ or $(x_2 = 1) \Rightarrow x_1 + x_2 \geq 1$
- Implications a) if P_2 is selected then P_3 is selected
  If $(x_2 = 1)$ then $(x_3 = 1) \Rightarrow x_2 \leq x_3$

  b) if P_2 is selected then P_3 is Not selected
  If $(x_2 = 1)$ then $(1 - x_3 = 1) \Rightarrow x_2 \leq 1 - x_3 \Rightarrow x_2 + x_3 \leq 1$

  c) P_1 can't be selected unless both P_2 and P_3 are selected
  If $(x_1 = 1)$ then $(x_2 = 1$ and $x_3 = 1) \Rightarrow x_1 \leq \frac{x_2 + x_3}{2} \Rightarrow 2x_1 \leq x_2 + x_3$

  OR: If $(x_1 = 1)$ then $\min(x_2, x_3) = 1 \Rightarrow x_1 \leq x_2, x_1 \leq x_3$

  D) If P_1 is selected then P_2 or P_3 is selected
  If $(x_1 = 1)$ then $(P_2 = 1$ or $P_3 = 1) \Rightarrow x_1 \leq x_2 + x_3$

  E) If P_2 or P_3 is selected then P_4 is selected
  If $(x_2 = 1$ or $x_3 = 1)$ then $(x_4 = 1) \Rightarrow x_2 + x_3 \leq 2x_4$

  OR: $\max(x_2, x_3) \leq x_4 \Rightarrow x_2 \leq x_4, x_3 \leq x_4$

  F) if both P_2 and P_3 are selected then P_4 is selected
  $x_2 + x_3 \leq x_4 + 1$ proof:
  if $(\min(x_2, x_3) = 1)$ then $(x_4 = 1) \Rightarrow \min(x_2, x_3) \leq x_4$
  $\Rightarrow \max(1 - x_2, 1 - x_3) \geq 1 - x_4 \Rightarrow (1 - x_2) + (1 - x_3) \geq 1 - x_4$
  $\Rightarrow x_2 + x_3 \leq x_4 + 1$

Fixed charge Problem

	Barbie	Dipsy	Avail
material	12	8	600
labour	120	100	10000
machine	10	15	800
Revenue	28	19	

- each unit of material costs \$2
- production of barbie incur fixed cost \$100
- Decision var: x_B, x_D : # barbie, # of dipsy

  y_R : # of material used

  z_B : Binary var to encode

  $z_B = \begin{cases} 1 & \text{if } x_B > 0 \\ 0 & \text{o/w} \end{cases}$

  if $x_B = 0$, $z_B = 0$

$\max\ 28x_B + 19x_D - 2y_R - 100z_B$

s.t (raw) $12x_B + 8x_D \le y_R,\ y_R \le 600$ (1)

(labour) $120x_B + 100x_D \le 10000$ (2)

(m/c) $10x_B + 15x_D \le 800$ (3)

$x_B, x_D, y_R \ge 0$, Integer (4)

$\underbrace{x_B \le M_B z_B}_{\text{Big-M constraint}}$

M_B : suitable large positive value

How big should M_B be?

M_B should be $\ge$ max value x_B can take when $x_B > 0$

can take $M_B = \min\left(\frac{600}{12}, \frac{1000}{120}, \frac{800}{10}\right) = 50$

IP : $\max\ 28x_B + 19x_D - 2y_R - 100z_B$

s.t. (1)(2)(3)(4)

$z_B = \{0, 1\}$, $x_B \le 50z_B$ (Big-M)

Q: suppose using m/c incurs a fixed cost of \$50

A: Introduce binary variable $z_{MC} \in [0, 1]$ encode

$10x_B + 15x_D > 0 \Rightarrow z_{MC} = 1$, add $-50z_{MC}$ to obj function

Big-M : $10x_B + 15x_D \le 800z_{MC}$

(Another way : $x_B > 0 \Rightarrow z_{MC} = 1 \Rightarrow x_B \le M_1 z_{MC}$

$x_D > 0 \Rightarrow z_{MC} = 1 \Rightarrow x_D \le M_2 z_{MC}$)

Disjunctive (OR) constraints

Q: suppose we have a new doll: Twinky dolls

	Twinky
Raw	10
Labour	150
M/C	10
Revenue	25

Twinky dolls can only be produced if its production is ≥ 40 dolls

A: X_T : # of Twinky dolls ≥ 0, Integer

$$\max\ 28X_B + 19X_D - 2y_R - 100Z_B + 25X_T$$

S.T. $12X_B + 8X_D + 10X_T \leq y_R$ (1')

$120X_B + 100X_D + 150X_T \leq 10000$ (2')

$10X_B + 15X_D + 10X_T \leq 800$ (3')

$X_B \leq 50Z_B$ (Big-M)

$X_B, X_D, X_T \geq 0$, Integer, $Z_B \in \{0,1\}$

Need to encode $(X_T = 0)$ or $(X_T \geq 40)$

$$X_T \geq 0 \Rightarrow Z_T = 1 \Rightarrow X_T \geq 40$$

Big-M: $X_T \leq M_T Z_T$ (4)

$X_T \geq 40 Z_T$ (5)

(4) + (5): $X_T > 0 \Rightarrow X_T \geq 40$

Go back to problem without Twinky's

Q: suppose we have another m/c, m/c2; call old m/c m/c 1

we have to exclusively use one of the 2 m/c for entire production

The requirements of barbie, dipsy on m/c 2

	barbie	dipsy	availability
m/c2	8	10	600

When is a production plan (x_B, x_D) feasible/realizable?

(of course they should satisfy (raws)(labour) constraints)

ignoring (raw)(labour) constraints, are the following plans realizable?

- (80, 0) : Yes, can use m/c 1
- (0, 60) : Yes, can use m/c 2
- (50, 10) : yes, can us m/c 1 or m/c 2

A: mathematically

$10x_B + 15x_D \leq 800$ or $8x_B + 10x_D \leq 600$ (*)

Introduce vars $z_{mc1}, z_{mc2} \in \{0,1\}$

a) $z_{mc1} = 1 \Rightarrow 10x_B + 15x_D \leq 800$

b) $z_{mc2} = 1 \Rightarrow 8x_B + 10x_D \leq 600$

Given a and b, can encode (*) by $z_{mc1} + z_{mc2} \geq 1$

i) $10x_B + 15x_D \leq 800 z_{mc1} + M_1(1 - z_{mc1})$

ii) $8x_B + 10x_D \leq 600 z_{mc2} + M_2(1 - z_{mc2})$

iii) $z_{mc1} + z_{mc2} \geq 1$

M_1 is large enough so that $10x_B + 15x_D \leq M_1$ whenever $z_{mc1} = 0$

M_2 is large enough so that $8x_B + 10x_D \leq M_2$ whenever $z_{mc2} = 0$

When $z_{mc1} = 0$, we know $z_{mc2} = 1 \Rightarrow 8x_B + 10x_D \leq 600 \} \times 1.5 \Rightarrow$

$\Rightarrow 10x_B + 15x_D \leq 12x_B + 15x_D \leq 900$

exercise: figure out a suitable bound on M2

IP: some obj function (raw)(labour)(i)(ii)(iii), $z_{mc1}, z_{mc2} \in [0,1]$

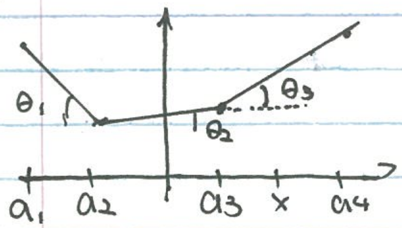
exercise: Suppose we have constraints $a_i^T x \leq b_i$, $0 \leq a_i^T x \leq M_i$ $\forall$ feasible x.

encode: $\geq k$ of constraints $a_i^T x \leq b_i$. what if $\leq k$ of the constraints $a_i^T x \leq b_i$.

Lecture 9B

Piecewise linear Functions $\quad x \in [a_1, a_4]$

$\theta_1 > 0$



we want to express $f(x)$ as a linear expression using some fractional / integer vars.

we write $x = a_1 + \delta_1 + \delta_2 + \delta_3$

$$(*)\begin{cases} \delta_1 = \text{"portion of } x \text{ in } [a_1, a_2]\text{"} \\ \delta_2 = \text{"portion of } x \text{ in } [a_2, a_3]\text{"} \\ \delta_3 = \text{"portion of } x \text{ in } [a_3, a_4]\text{"} \end{cases}$$

Then $f(x) = b_1 + \delta_1\theta_1 + \delta_2\theta_2 + \delta_3\theta_3$

To enforce (*):

$\delta_i \geq 0 \quad \forall i = 1, \ldots, 3$

$\delta_1 \leq a_2 - a_1, \quad \delta_2 \leq a_3 - a_2, \quad \delta_3 \leq a_4 - a_3$

$$\underbrace{\begin{aligned} &\delta_2 > 0 \Rightarrow \delta_1 \text{ in at its max value of } a_2 - a_1 \\ &\quad \Rightarrow u_1 = 1 \Rightarrow \delta_1 \geq a_2 - a_1 \end{aligned}}$$

Big-M constraint $\delta_2 \leq (a_3 - a_2)u_1, \ \delta_1 \geq (a_2 - a_1)u_1 \quad u_1 \in \{0, 1\}$

$$\underbrace{\begin{aligned} &\delta_3 > 0 \Rightarrow \delta_2 \text{ in at its max value of } a_3 - a_2 \\ &\quad \Rightarrow u_2 = 1 \Rightarrow \delta_2 \geq a_3 - a_2 \end{aligned}}$$

Big-M constraint $\delta_3 \leq (a_4 - a_3)u_2, \ \delta_2 \geq (a_3 - a_2)u_2 \quad u_2 \in \{0, 1\}$

Entire set of constraints

$x = a_1 + \delta_1 + \delta_2 + \delta_3$ (δ_i's integer if x is integer. fractional o/w)

$f(x) = b_1 + \delta_1\theta_1 + \delta_2\theta_2 + \delta_3\theta_3$

$\delta_1, \delta_2, \delta_3 \geq 0$

$\delta_1 \leq a_2 - a_1, \ \delta_2 \leq a_3 - a_2, \ \delta_3 \leq a_4 - a_3$

$\delta_1 \geq (a_2 - a_1)u_1, \ \delta_2 \leq (a_3 - a_2)u_1$

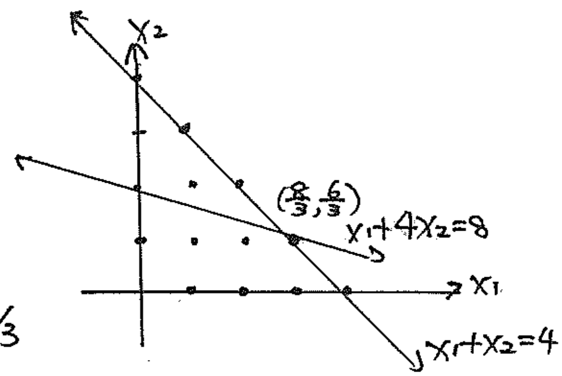
$\delta_2 \geq (a_3 - a_2)u_2, \ \delta_3 \leq (a_4 - a_3)u_2$

$u_1, u_2 \in \{0, 1\}$

Solution method for IPs

$$\begin{aligned}
\max\ & 3x_1+10x_2 && \text{(IP)}\\
\text{s.t. } & x_1+4x_2 \le 8 && (1)\\
& x_1+x_2 \le 4 && (2)\\
& x_1, x_2 \ge 0 && (3)\\
& x_1, x_2 \text{ integer} && (4)
\end{aligned}$$

$F_0 = \{(x_1,x_2) : (1)(2)(3)\}$, $Opt_{LP_0} = 64/3$



Defn The LP relaxation of an integer program in the LP obtained by dropping the integerity constraints

$$\begin{aligned}
\max\ & 3x_1+10x_2\\
\text{s.t. } & (1)(2)(3)
\end{aligned}$$

F_0 = feasible region of (LP_0)

Theorem : $OPT_{IP} \le OPT_{LP\text{-relaxation of } IP}$

Corollary: If the LP-relaxation of an IP has optimal soln X^* that is integral then X^* is an optimal solution to IP

proof : X^* is the feasible soln to IP since its feasible for LP-relaxation and an integral value of X^* $= OPT_{LP\text{ relaxation of } IP} \ge OPT_{IP}$
$\Rightarrow X^* =$ opt sol to IP

Strategy : solve LP-relaxation to get opt soln X^*. if integral then done. if not, branch and bound uniqu opt sol to (LP_0) in $X_0^* = (\frac{8}{3}, \frac{6}{3})$ with $OPT\ LP_0 = \frac{64}{3}$, we know $OPT_{IP} \le \frac{64}{3}$

Notation $\lfloor x \rfloor$: floor of x = largest int $\le x$, $\lceil x \rceil$: ceiling of x = smallest int $\ge x$

e.g. $OPT_{LP} \le \frac{64}{3} \Rightarrow OPT_{IP} \le \lfloor \frac{64}{3} \rfloor = 21$ (UB)

$\lfloor 2.1 \rfloor = 2$, $\lceil 2.1 \rceil = 3$, $\lfloor -1.8 \rfloor = -2$, $\lceil -1.8 \rceil = -1$

Trivial : any int $X = (x_1, x_2)$ will have $x_1 \ge \lceil \frac{8}{3} \rceil = 3$ or $x_1 \le \lfloor \frac{8}{3} \rfloor = 2$

Every int sol must lie in $F_1 = F_0 \cap \{(x_1,x_2) : x_1 \ge \lceil \frac{8}{3} \rceil = 3\}$, $F_2 = F_0 \cap \{(x_1,x_2) : x_1 \le \lfloor \frac{8}{3} \rfloor\}$

Opt soln to IP : better of best int sol in F_1 and best int sol in F_2

Lecture 10A

We want to solve

$\max\ 3x_1 + 10x_2$

$$F_0 \begin{cases} x_1 + 4x_2 \le 8 & (1) \\ x_1 + x_2 \le 4 & (2) \\ x_1, x_2 \ge 0 & (3) \\ x_1, x_2 \text{ integer} & (4) \end{cases}$$

<u>Branch and bond</u>

$LP_0 : \max\ 3x_1 + 10x_2 \quad \text{s.t. } x \in F_0$

$OPT_{LP_0} = \frac{64}{3} \quad x_0^* = \left(\frac{8}{3}, \frac{6}{3}\right) \quad LB = -\infty,\ UB = \lfloor \frac{63}{3} \rfloor = 21$

$x_1 \ge \lceil \frac{8}{3} \rceil = 3$ $\qquad$ $x_1 \le \lfloor \frac{8}{3} \rfloor = 2$

$LP_1 : LP_0 + \text{"}x_1 \ge 3\text{"}$

$OPT_{LP_1} = 19, \quad \underbrace{x_1^* = (3,1)}_{LB = 19}$

$LP_2 : LP_0 + \text{"}x_1 \le 2\text{"}$

$OPT_{LP_2} = 21, \quad x_2^* = (2, 1.5)$

$x_2 \ge \lceil 1.5 \rceil = 2$ $\qquad$ $x_2 \le \lfloor 1.5 \rfloor = 1$

$LP_3 : LP_2 + \text{"}x_2 \ge 2\text{"}$

$OPT_{LP_3} = 20, \quad \underbrace{x_3^* = (0,2)}_{LB = 20}$

$LP_4 : LP_2 + \text{"}x_2 \le 1\text{"}$

$OPT_{LP_4} = 16$

$\Rightarrow$ no benefit in completing F_4 since $OPT_{LP_4} \le LB$

$\Rightarrow$ pruning a node

Primal (Normal) simplex	Dual simplex
• works with feasible bases (i.e. RHS ≤ 0) and moves towards optimality i.e. reduced costs ≤ 0	• works with dual feasible bases i.e. where all reduced costs ≤ 0 move towards, primal feasibility, RHS $\ge$
• First choose entering vars then corresponding leaving vars.	• First chose a leaving vars then choose a corresponding entering vars.

Branch and Band Implementation

we solve new LPs encountered in BnB using dual simplex method

solving (LP₀) gives following simplex tableau.

$$z + \tfrac{7}{3} s_1 + \tfrac{2}{3} s_2 = \tfrac{64}{3}$$

(T₀) $$x_1 - \tfrac{1}{3} s_1 + \tfrac{4}{3} s_2 = \tfrac{8}{3} \qquad \text{Basis} = \{x_1, x_2\}$$

$$x_2 + \tfrac{1}{3} s_1 - \tfrac{1}{3} s_2 = \tfrac{4}{3}$$

$$-x_1 + s_3 = -3$$

we solve (LP_1) by reusing info from (T_0) : $LP_1 : LP_0 + $ "$x_1 \geq 3$"

Add $x_1 \geq 3 \Rightarrow x_1 - s_3 = 3 \Rightarrow -x_1 + s_3 = -3$ $\quad$ $s_3 \geq 0$ is slack var

old basis $\{x_1, x_2\}$ extends to new basis $\{x_1, x_2, s_3\}$. convert (T_0) + "$-x_1+s_3=3$" to tableau by adding x_1-row of (T_0) to $-x_1+s_3=-3$ which gives

$$z + \tfrac{7}{3} s_1 + \tfrac{2}{3} s_2 = \tfrac{64}{3}$$

$$x_1 - \tfrac{1}{3} s_1 + \tfrac{4}{3} s_2 = \tfrac{8}{3} \qquad (T_1) \text{ tableau for } \{x_1, x_2, x_3\}$$

$$x_2 + \tfrac{1}{3} s_1 - \tfrac{1}{3} s_2 = \tfrac{4}{3}$$

$$-\tfrac{1}{3} s_1 + \tfrac{4}{3} s_2 + s_3 = -\tfrac{1}{3}$$

Leaving var : choose basic var x_r where RHS of x_r-row < 0

Entering var : choose non-basis var x_k where coefficient < 0 or $\frac{\bar{c}_k}{\bar{a}_{rk}} = \min_{\bar{a}_{rk} \leq 0} \frac{\bar{c}_i}{\bar{a}_{ri}} \geq 0$

pivot on (s_3, s_1) : $z + 10 s_2 + 7 s_3 = 19$

$$x_1 - s_3 = 3$$

$$x_2 + s_2 + s_3 = 1 \qquad (T_2) \text{ optimal for } (LP_1)$$

$$s_1 - 4 s_2 - s_3 = 1$$

* suppose instead of $-\frac{1}{3} s_1 + \frac{4}{3} s_2 + s_3 = -\frac{1}{3}$ in (T_1) we had $-\frac{1}{3} s_1 - \frac{4}{3} s_2 + s_3 = -\frac{1}{3}$
  choose s_3 as leaving var but for entering var use $\frac{\bar{c}_k}{\bar{a}_{rk}} = \min_{\bar{a}_{rk} \leq 0} \frac{\bar{c}_i}{\bar{a}_{ri}} \geq 0$

In general : $z - \sum_{j \in N} \bar{c}_j x_j = \bar{v}$, $\quad \left[x_r + \sum_{j \in N} \bar{a}_{rj} x_j = \bar{b}_r < 0 \right] \times \frac{\bar{c}_k}{\bar{a}_{rk}}$

If we choose x_k with $\bar{a}_{rk} < 0$ as entering var, pivot on (x_r, x_k) gives

$$z + \frac{\bar{c}_k}{\bar{a}_{rk}} x_r - \sum_{j \in N} \left(\bar{c}_j - \frac{\bar{c}_k}{\bar{a}_{rk}} \bar{a}_{rj} \right) x_j = \bar{v} + \bar{b}_r \cdot \frac{\bar{c}_k}{\bar{a}_{rk}}$$

So new reduced cost of x_j where $j \in N$ is $\left[\bar{c}_j - \frac{\bar{c}_k}{\bar{a}_{rk}} \bar{a}_{rj} \right] < 0$
if $\frac{\bar{c}_j}{\bar{a}_{rj}} \geq \frac{\bar{c}_k}{\bar{a}_{rk}}$ if $\bar{a}_{rj} < 0$.

* If no leaving var $\Rightarrow$ OPT solution
  If no entering var $\Rightarrow$ all non-basic vars have ≥ 0 coefficients

$$(x_r\text{-row}) : \underbrace{x_r + \sum_{j \in N} \bar{a}_{rj} x_j}_{LHS \geq 0,\ \forall x \geq 0} = \bar{b}_r < 0$$

Lecture 10B

Definition

An inequality $\alpha^T x \leq \beta$ is:

- a valid inequality for a set X if it is satisfied by all $\alpha \in X$
- a valid for an IP/LP if it is a valid inequality for feasible region

Cutting-Plane Algorithm for solving IPs

Given $\max c^T x$

$$\text{s.t. } Ax \leq b$$
$$x \geq 0 \quad x \text{ integer}$$

assume that LP-relaxation of IP is not unbounded

Let $F = \{x : Ax \leq b, x \geq 0\}$, feasible region of LP relax of IP

$X = \{x \in F, x \text{ integer}\}$ feasible region of IP

1) Initialize $i = 0$, $F_0 \leftarrow F_i$ $\quad (X = F_i \cap \{x : x \text{ integer}\})$

2) Solve LP_i : $\max c^T x$ s.t. $x \in F_i$

 If LP_i is infeasible, then STOP, return IP is infeasible

 Else let $x^{*(i)}$ be optimal solution to LP_i

3) If $x^{*(i)}$ is integral, then STOP, return $x^{*(i)}$ opt solution to IP

4) Otherwise find an inequality $\alpha^T x \leq B$ that is

 a) valid for X

 b) violated by $x^{*(i)}$ (i.e. $\alpha^T x^{*(i)} > B$)

5) Update $F_{i+1} \leftarrow F_i \cap \{x : \alpha^T x \leq \beta\}$ Note $x^{*(i)} \notin F_{i+1}$

 $i \leftarrow i+1$

6) Go back to step 2

Generating Valid Inequalities

A) Exploiting Problem specific Structure $(\times) = 3x_1 - 4x_2 + 2x_3 - 3x_4 + x_5 \le -2$

Let $X = \{x \in \{0,1\}^5 : 3x_1 - 4x_2 + 2x_3 - 3x_4 + x_5 \le -2\}$

$F = \{x \in [0,1]^5 : 3x_1 - 4x_2 + 2x_3 - 3x_4 + x_5 \le -2\}$

1) can't have $x_2 = x_4 = 0$, for any point $x \in X$

o/w, $0 \le$ LHS, so $(\times)$ is not satisfied

so every $x \in X$ must satisfy $(x_2 = 1)$ or $(x_4 = 1)$ enforced by $x_2 + x_4 \ge 1$

This is not valid for F (e.g. $x_2 = \frac{1}{2}$, $x_i = 0 \ \forall \ i \ne 2$)

2) If $x_1 = 1$ then can't have $x_2 = 0$ for any $x \in X$

(if $x_1 = 1$, $x_2 = 0$ then LHS of $(\times) = 3 - 3 \ge 0 \Rightarrow (\times)$ is violated

so if $(x_1 = 1)$ then $(x_2 = 1)$, $x_1 \le x_2$ is violated for X

B) Chvátal – Gomory Procedure

consider $X = \{(x_1, x_2, x_3)$ $\begin{cases} (a)\ 3x_1 + 2x_2 \le 3 \\ (b)\ x_2 + x_3 \le 1 \\ (c)\ x_1 + x_3 \le 1 \\ (d)\ x_1, x_2, x_3 \ge 0, \text{int} \end{cases}$

$F = \{(x_1, x_2, x_3) : (a) \leftrightarrow (d)\}$

consider inequity: $\frac{1}{4} \times (a) + \frac{1}{2} \times (b) + \frac{1}{2} \times (c)$

$1.25x_1 + x_2 + x_3 \le 1.75$ $(*)$

$\Rightarrow x_1 + x_2 + x_3 \le 1.75$ $(**)$

All points even in F satisfy $(*)$ and since $x \ge 0$, all points in F satisfy $(**)$

For any $x \in X$, LHS of $(**)$ is an integer, so LHS of $(**) \le 1.75$

$x_1 + x_2 + x_3 \le 1$ $(***)$. $(0.5, 0.5, 0.5) \in F$, but violates $(***)$ so not violated.

In general, suppose $X = \{x : Ax \le b, x \ge 0 \ x \text{ int}\}$, $F = \{x : Ax \le b, x \ge 0\}$

1) Generate a valid inequality, even for F, by taking a nonneg linear comb $Ax \le$

$a_1^T x \le b_1] \times y_1 \ge 0$ ...... $a_n^T x \le b_n] \times y_n \ge 0$

$\Rightarrow \sum_{i=1}^{n} (y_i a_i^T) x \le \sum_{i=1}^{n} b_i y_i \Rightarrow \alpha^T x \le B$

2) $\alpha^T x \le B$ valid for F, $x \ge 0 \Rightarrow \sum_{i=1}^{n} [\alpha_j] x_j \le B$ $(**)$

CG cut inequality → 3) LHS of $(**)$: integer $\forall x \in X \Rightarrow \sum_{i=1}^{n} \lfloor \alpha_j \rfloor x_j \le \lfloor B \rfloor$ valid for X $(***)$

example for IP : $\overbrace{y_1}^{\ge 0} \times (1) + \overbrace{y_2}^{\ge 0} \times (2) : (y_1 + y_2) x_1 + (4y_1 + y_2) x_2 \le 8y_1 + 4y_2$

- $y_1 = \frac{1}{3}$, $y_2 = \frac{2}{3}$, $x_1 + 2x_2 \le 1\frac{1}{3}$ ⇒ gives CG cut $x_1 + 2x_2 \le 5$
- $y_1 = \frac{2}{3}$, $y_2 = \frac{1}{3}$, $x_1 + 3x_2 \le \frac{20}{3}$ ⇒ gives CG cut $x_1 + 3x_2 \le 6$

( ) Gomory cuts (special case of CG procedure)

(IP) : max $c^T x$ s.t. $Ax=b$, $x \geq 0$, x integer

suppose upon solving LP-relaxation, we get opt sol x^*; x^* is fractional

consider x_i-row of final simplex tableau

$$x_i + \sum_{j \in N} \bar{a}_{ij} x_j = \bar{b}_i \quad (*)$$

$(*)$ is obtained from taking a linear combination of $Ax=b$

$\Rightarrow$ step 1 of the CG procedure.

Run CG procedure starting with $(*)$

2) $x_i + \sum_{j \in N} \lfloor \bar{a}_{ij} \rfloor x_j \leq \bar{b}_i$ satisfied by all feasible x for LP-relaxation

3) $x_i + \sum_{j \in N} \lfloor \bar{a}_{ij} \rfloor x_j \leq \lfloor \bar{b}_i \rfloor$ valid for (IP)

Rewrite $(***)$: consider $(*) - (***)$: $\underbrace{\sum (\bar{a}_{ij} - \lfloor \bar{a}_{ij} \rfloor) x_j \geq \bar{b}_i - \lfloor \bar{b}_i \rfloor}_{(q)}$

(q) Gomory cut generated from x_i-row

(q) is violated by x^* RHS of (q) >0

LHS of (q) $=0$ at x^* since all x_j^* for $j \in N = 0$

Example

Solving LP-relaxation (LP_0) of (IP) gives final simplex tableau

$$\left.\begin{aligned} z + 7/3 s_1 + 2/3 s_2 &= 64/3 \\ x_1 - 1/3 s_1 + 4/3 s_2 &= 8/3 \\ x_2 + 1/3 s_1 - 1/3 s_2 &= 4/3 \\ -2/3 s_1 - 1/3 s_2 + s_3 &= -2/3 \end{aligned}\right\} (T_0')$$

s_1, s_2 - slack var for (1) (2)

Gomory cut from x_1-row : $2/3 s_1 + 1/3 s_2 \geq 2/3$

Add this to tableau, run dual simplex on LP :

s_3 leaves, s_2 enters. pivot on (s_3, s_2) gives

$$\left.\begin{aligned} z + s_1 + 2 s_3 &= 20 \\ x_1 - 3 s_1 + 4 s_3 &= 0 \\ x_2 + s_1 - s_3 &= 2 \\ 2 s_1 + s_2 - s_3 &= 2 \end{aligned}\right\} (T_1)$$

final simplex tableau for new LP as all RHS >0

opt soln to new LP is integral $x^* = (0,2,0,2,0)^T$

Lecture 11A

<u>Strength of Formulations</u>

<u>Definition</u> : let (IP$_1$), (IP$_2$) be 2 integer programs with same set X of feasible solutions.

Let (LP$_1$), (LP$_2$) be LP relaxations of (IP$_1$)(IP$_2$) having feasible region F_1, F_2

$X = \{x \in F_1 : x \text{ integer}\} = \{x \in F_2 : x \text{ integer}\}$

Say that (IP$_2$) is stronger than (IP$_1$) if $F_2 \subseteq F_1$

<u>Observations</u>

1) suppose (IP$_2$) is obtained from (IP$_1$) by an inequality $\alpha^T x \le \beta$ violated for (IP$_1$)

$F_2 = F_1 \cap \{x : \alpha^T x \le \beta\} \subseteq F_1$

By definition, (IP$_2$) is stronger than (IP$_1$)

2) If (IP$_2$) is stronger than (IP$_1$), then for any objective

$\max c^T x$, $OPT_{IP_1} = OPT_{IP_2} \le \underbrace{\max c^T x \text{ s.t. } x \in F_2}_{OPT_{LP_2}} \le \underbrace{\max c^T x \text{ s.t. } x \in F_1}_{OPT_{LP_1}}$

3) recall a set $S \subseteq \mathbb{R}^n$ is called convex if $\forall x, y \in S$, $\forall \lambda \in [0,1]$ the point $\lambda x + (1-\lambda) y$ is also in S

 convex  not convex

The <u>convex hull</u> of a set $X \subseteq \mathbb{R}^n$ denoted $CH(X)$ is the smallest convex set containing X.

Fact: The feasible region of an LP is a convex set.

Suppose we have an IP formulation, with feasible region X, where LP relaxation (LP) is such that F = feasible region of LP $= CH(X)$

Then if (IP') is any other integer program ; also with feasible region X, we have that (IP) is stronger than (IP')

<u>Why?</u> let F' is convex, $F' \supseteq X \Rightarrow F' \supseteq CH(X)$

$\Rightarrow$ (IP) is stronger than (IP')

$\Rightarrow$ (IP) is the strongest formulation with feasible region X.

Example Facility-Location Problem

$F = \{1, \dots, n\}$: candidate facility locations, each facility i has a facility opening cost $f_i > 0$

$C = \{1, \dots, m\}$: clients. Each client j has to be assigned to an open facili[ty]; assigning client j to facility i incurs cost c_{ij}

Goal Minimize total cost

IP Formulations

Binary variables $y_i = \begin{cases} 1 & \text{if facility } i \text{ is open} \\ 0 & \text{o/w} \end{cases}$

$x_{ij} = \begin{cases} 1 & \text{if client } j \text{ assigned to facility } i \\ 0 & \text{o/w} \end{cases}$

Objective $\min \sum_{i \in F} f_i y_i + \sum_{j \in C} \sum_{i \in F} c_{ij} x_{ij}$

constraints $\sum_{i \in F} x_{ij} = 1 \quad \forall j \in C \quad (1)$ all vars ≥ 0 (5)

$x_{ij}, y_i \in \{0, 1\} \quad \forall i \in F, j \in C \quad (2)$

encode: client can only be assigned to an open facility 2 ways:

if $(x_{ij} = 1)$ then $(y_i = 1)$ $\forall i \in F, j \in C$

$x_{ij} \leq y_i \quad \forall i \in F, j \in C \quad (3)$

if $(\sum_{j \in C} x_{ij} > 0)$ then $(y_i = 1)$ $\forall i \in F$

Big M : $\sum_{j \in C} x_{ij} \leq M y_i \quad \forall i \in F$ (4) M: largest possible value LHS

(IP_1) : IP with ①②③⑤ $\qquad (LP_1)$: LP with ①③⑤

(IP_2) : IP with ①②④⑤ $\qquad \underbrace{(LP_2) : \text{LP with ①④⑤}}_{\text{LP relaxations}}$

if (x, y) is feasible for (LP_1) then (x, y) is also feasible in (LP_2)

$\Rightarrow (IP_1)$ is stronger than (IP_2)

Lecture 11B

<u>Network flows</u>

example : transportation problem

n outlets, each outlet i can supply S_i units of a product

m customers, customer j can handle a max demand of d_j

each outlets i can supply to some set C_i of customers.

what is the max number of units that can be transported from outlet to cust

<u>Notation / Terms</u>

A directed graph is a pair $G = (V, E)$ where

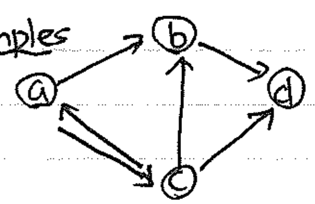
V: Vertex / node set

E: set of directed edges

each edge in E is of the form $u \to v$

i.e. starts at a node $u \in V$, ends at another node $v \in V$

<u>Examples</u>



$V = \{a, b, c, d\}$

$E = \{\underbrace{(a,b)}_{a \to b}, (b,d), (c,d), (a,c), (c,b), (c,a)\}$

Examples of flows = product / commodity / traffic

Moving through a network:

i) Transportation network (highway network = directed graph)

ii) electrical network (circuit = directed graph)

edges = wires, vertices = circuit deck (battery, resister)

A <u>flow network</u> consists of:

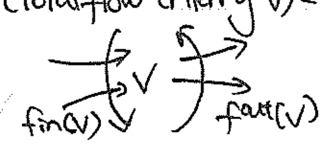
- a directed graph $G = (V, E)$
- capacity $c_e \geq 0$ for every $e \in E$
- source $s \in V$, sink $t \in V$ (assume s has no incoming edges) t has no outgoing edges

An s-t flow is an assignment of value $f_e \geq 0$ to the edges of G satisfy :

1) $0 \leq f_e \leq c_e$ (capacity constraint)

2) for every node $v \neq s, t$ (total flow entering v) = (total flow leaving u)

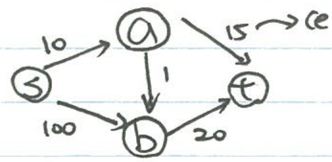
i.e $\underbrace{\sum_{e \text{ enters } v} f_e}_{f^{in}(v)} = \underbrace{\sum_{e \text{ leaves } v} f_e}_{f^{out}(v)}$



Conservation constraints

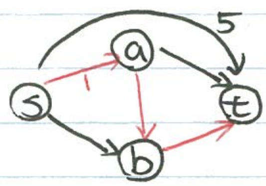
The value $v(f)$ of a flow (i.e. s-t flow) is defined as $v(f) = f^{out}(s) = \sum_{e \text{ leaves}} f$

Example

flow network: 

— send along path

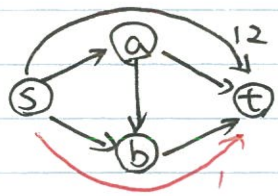
— send along path

a) 

$f_{sa} = 5+1 = 6$ $\quad f_{at} = 5$

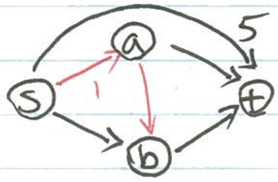
$f_{ab} = 1$, $f_{sb} = 0$, $f_{bt} = 1$

f: flow of value $v(f) = 6$

b) 

not a flow because

$f_{sa} = 12 > 10 = c_{sa}$

c) 

$f_{sa} = 6$, $f_{ab} = 1$ ...

not a flow because

$f^{in}(b) = 1 > 0 = f^{out}(b)$

maximum flow problem

given flow network $\begin{cases} G=(V,E) \\ \text{s.t. } \{e\}_{\in E} \end{cases}$

Find an s-t flow f that max $v(f)$

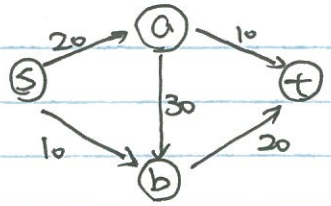
every s-t flow f has:

- $v(f) \le 110$, since $v(f) = f_{sa} + f_{sb} \le c_{sa} + c_{sb} \ge 110$
- $v(f) \le 30$, f must exit $\{s,b\}$ and in doing so use up to ≥ 1 unit of some outgoing edge to $\{s,b\}$ total capacity of such edges $= 30$
- $v(f) \le 35$. $v(f) =$ total flow entering t, $\le c_{at} + c_{bt} = 35$

Lecture 12A

Ford Fulkerson Algorithm

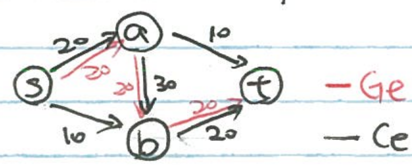
Running network



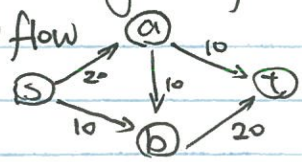
start with feasible flow
$f_e = 0, \ \forall e \in E$

Defn: an s-t path in $G=(V,E)$ is a sequence
$v_0 = s, v_1, v_2, \dots, v_k, v_{k+1} = t$ where $(v_i, v_{i+1}) \in E \ \forall i = 0, \dots, k$

Idea: push flow doing st. paths.
Here, best s-t path to push flow on is $s \to a \to b \to t$
on which we can push 20 units. This gives flow:



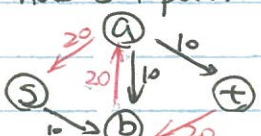
new flow



to make progress
- push "forward" 10 units on $(s,b) \Rightarrow \uparrow f_{sb}$ by 10
- push "backward" 10 units on $(a,b) \Rightarrow \downarrow f_{ab}$ by 10
- push "forward" 10 units on $(a,t) \Rightarrow \uparrow f_{at}$ by 10

residual graph given a flow network $(G=(V,E)$ s.t. $\{c_e\}_{e\in E})$ and s-t flow f,
the residual graph of G w.r.t. f, denoted G_f, is as follows
- G_f has vertex set V
- for every edge $e=(a,b)$ of G with $f_e < c_e$, include edge e (forward edge) in G_f $\overline{w}$ capacity $c_e - f_e$
- for every edge of G with $f_e > 0$, include f_e (opposite) edge (b,a) $\overline{w}$ capacity f_e

example G_f has s-t path $s \to b \to a \to t$



call an s-t path in $G_f \Rightarrow$ augmenting path
bottleneck$(p, f) = 10$

A: A is such that $s \in A, t \notin A$, A has no outgoing edges
A: $\{v, v$ is reachable flow in $G_f\}$

Given a s-t path p in G_f, define bottleneck $(p,f) = \min_{e \in P}$ (residual capacity of e)

Augment (f, p) – let x = bottleneck (p,f)

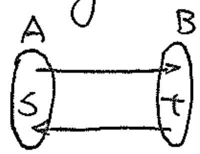
- For each edge $e = (u,v) \in P$
- if (u,v) is a forward edge, set $f_e \leftarrow f_e + x$
- if (u,v) is a backward edge, set $f_{e'} \leftarrow f_{e'} - x$
  $e' = (u,v)$ is in G
- return f.

Claim 1: if $f' \leftarrow$ augment (f,p) then
- f' is an s-t flow
- $v(f') = v(f) +$ bottleneck (p,f)

<u>Ford-Fulkerson Algorithm</u> given $G = (V,E)$, s.t. $\{c_e\}$ $e \in E$

1. start with $f_e \leftarrow 0$ $\forall e \in E$
2. compute residual graph G_f and f
3. While – if s-t path is G_f, pick an s-t path p in G_f
   – update $f \leftarrow$ augment (f,p), update G_f
   – return f

<u>Definition</u> An s-t arc of $G = (V,E)$ is a pattern, (A,B) of V s.t. $s \in A$, $t \in B$. The capacity of an st arc (A,B) denoted by $c(A,B)$ is defined by $c(A,B) = \sum' c_e$



$e = (u,v)$ $u \in A$, $v \in B$, total capacity $= c(A,B)$

a min s-t cut is an s-t arc with min capacity

<u>Thm 2</u>

Let c_e be an integer. $\forall e \in E$, then FF algorithms terminates in a finite # of iterations and returns an integral flow (i.e. f_e's are int)

<u>Thm 3</u> let f^* be flow returned by FF algorithm.

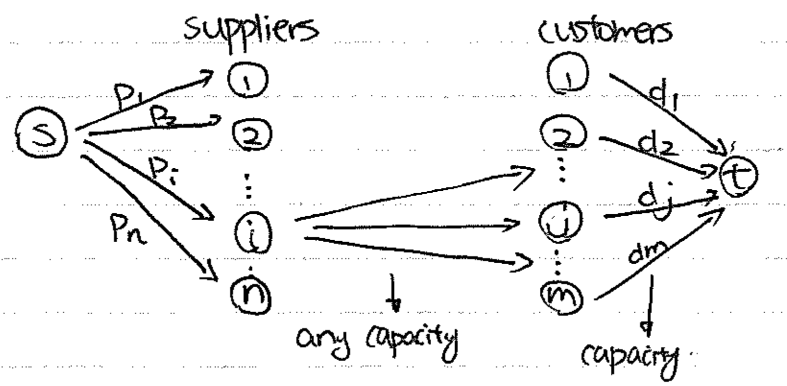
Let $A = \{s : \exists\, s \to v$ path in $G_{f^*}\}$. i.e. A nodes reachable from s in G_{f^*}

Then $(A, B = V \setminus A)$ is an s-t and $v(f^*) = c(A,B) \to f^*$ is max st flow

(A,B) is a max st. cut

Applications recall transportation Problem

n suppliers, supplier i can supply p_i units (of some product)
m customers, customer j can handle at most d_j (i.e. max demand)
Each supplier i can supply to a set C_i of customers
What is the max # of units that can be transported from suppliers to customers?



Create edges $i \to j$ $\forall j \in C_i$
$s \to i$ $\forall$ supplier i
$j \to t$ $\forall$ customer j

We want to impose the correspondence:

x_{ij} units sent from supplier i to customer j = sending x_{ij} units of flow along the s-t path $s \to i \to j \to t$

Given correspondence: flow on $s \to i$ = total # of units supplied by i
flow on $j \to t$ = total # units sent to customer j

In this flow network, all capacities are integers
$\Rightarrow$ FF algorithm terminates with an integer max-flow

Using our correspondence, this yields a solution to our problem (since value of flow = total # units transported)

Lecture 12B

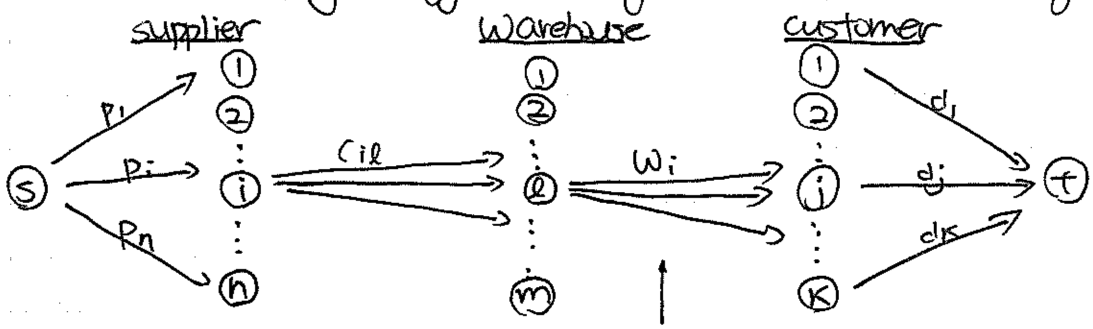
Question: In addition to suppliers, customers, there are k warehouses

supplier i may supply to a set W_i of warehouses

Each warehouse l may supply to a set of C_i of customers.

A warehouse l may store at most C_{il} unit supported by supplier i

Is there a shipping strategy s.t. every customer j receives exactly d_j units.



$W_{lj} \geq \min \sum_{i \in W_i} C_{il} d_j$

1 unit transported from supplier i to warehouse l to customer $j \in C_l$

$=$ 1 of flow along $s \to i \to l \to j \to t$ path

Under this correspondence:

- Flow on (s, i) : total units supplied by i
- Flow on (j, t) : total units sent to customer j
- Flow on (i, l): total units sent from supplier i to warehouse l
- Flow on (l, j): total units sent from warehouse l to customer j.

In terms of flows, we seek an integer value s-t flow s.t. flow on $(j,t) = d_j$ $\forall j$

such a flow must be a max s-t flow (since $v(f) \leq \sum_{j=1} d_j$ for any flow)

If we have an s-t flow of value $\sum_{j=1} d_j$ such flow must be max s-t flow

Is there a integer max s-t flow of value $\sum_{j=1} d_j$?

Flow network with integer capacities $\Rightarrow$ can use FF algorithm to find an integer max s-t flow f^*, if $v(f^*) = \sum d_j$, can find desired shipping strategy from our correspondence.

Otherwise, no such shipping strategy exists.

What if warehouse l had c_l total capacity for starting items (across all i)

<u>Theorem 1</u>

If all capacities are integers, the FF algorithm terminates with integer flow

<u>Observations</u>

if all c_is and f_is are integers, then bottleneck (P, f) is an integer ≥ 1 and $f' \leftarrow$ augment to (f, p) is an integer value.

(<u>bottleneck (P,f)</u> = min some f_e value >0 and int, and some $c_e - f_e$ value >0 in integer, $>0 \Rightarrow \geq 1$

$$f'_e \begin{cases} f_e \\ f_e + \text{bottleneck}(P,f) \\ f_e - \text{bottleneck}(P,f) \end{cases} \Rightarrow f'_e \text{ integer.}$$

<u>proof of theorem 1</u>

Let $C = \sum^{\text{leaving}} c_e$, we have $v(f) \leq C$ for any s-t flow f.

By observation, $\uparrow v(f) \geq 1$ in every iteration, f is integer valued

$\Rightarrow$ terminates in $\leq C$ iterations with an integer flow

Notation: let $A \subseteq V$, $f^{out}(A) = \sum^{\text{leaving}} f_e$

$f^{in}(A) = \sum^{\text{entering}} f_e$

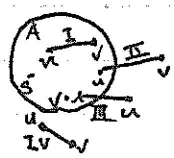
<u>Claim 2</u> let f be a s-t flow and (A, B) be an s-t cut ($s \in A, t \in B, A \cup B = V$)

Then $v(f) = f^{out}(A) - f^{in}(A)$, Hence, $v(f) \leq c(A,B) := \sum_{\text{leaving}} c_e$

<u>proof</u> we have $v(f) = f^{out}(s) = f^{out}(s) - \overbrace{f^{in}(s)}^{=0 \text{ since } s \text{ has no incoming edges}}$

$0 = f^{out}(v) - f^{in}(v) \quad \forall v \neq s, t$

(1) + (2) summed over all $v \in A$, $v \neq s$



total contribution of edges e to RHS of (*) is:

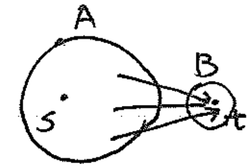
$$\begin{cases} 0, & e \text{ of type I} \\ f_e, & e \text{ of type II} \\ -f_e, & e \text{ of type III} \\ 0, & e \text{ of type IV} \end{cases}$$

So (*) $= v(f) = f^{out}(A) - f^{in}(A)$

Corollary 3

(i) $v(f) \le c(A,B)$

(ii) $v(f) = f^{in}(t)$ [take $(A,B) = (V\setminus\{t\}, \{t\})$]



$f^{out}(A) = f^{in}(t)$, $f^{in}(A) = 0$ since there are no edges leaving t

Lemma 4 Let f be an s-t flow, (A,B) s-t cut such that in G_f, A only has incoming edges. Then,

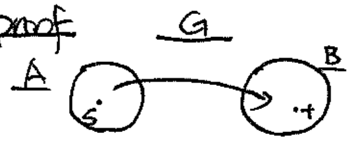
1) if $e = (u,v)$ is an edge of G, $u \in A$, $v \notin A$, $f_e = c_e$

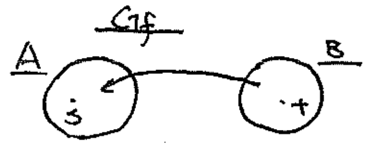
2) if $e = (u,v)$ is an edge of G, $u \notin A$, $v \in A$, $f_e = 0$

3) so $f^{out}(A) = c(A,B)$, $f^{in}(A) = 0$

$\Rightarrow v(f) = c(A,B) \Rightarrow f$ in a max s-t flow, (A,B) is a min s-t cut

proof





3) follows from 1), 2) and claim 2

1) Since e does not appear (as forward edge) in G_f, we have $f_e = c_e$

2) since e does not create backward edge (u,v) in G_f, $f_e = 0$

Theorem 5 Let $f^* \leftarrow$ flow returned by ff algorithm. $A^* = \{u : \exists\, v \text{ in } G_p^{f^*}\}$

Then $(A^*, B^* = U\setminus A^*)$ is an s-t cut, $v(f^*) = c(A^*, B^*) \Rightarrow f^*$ max st flow, (A^*,B^*) = min c

proof $t \notin A^*$ since FF algorithm terminated $\Rightarrow (A^*, B^*)$ is an s-t cut.

$\exists$ some path P, so $P + (u,v)$ gives an $s \leadsto v$ path

$v \in A^* \rightarrow$ contradiction since (u,v) leaves A^*

so A^* has only incoming edges $\Rightarrow$ by lemma 4, f^*: max s-t flow (A^*,B^*) min cu

In fact, we have shown, if f is s-t, G_f has no s-t path, then f is max flo

Theorem : max flow min cut theorem

In any flow network, value of max s-t flow = capacity of min s-t cut

proof max s-t flow problem can be phased as an LP.

The LP is feasible, not unbounded. $\Rightarrow$ LP has optimal sol $\Rightarrow$ max flow exist

$\Rightarrow$ no s-t path in $G_f \Rightarrow$ maxflow exists