Yi Ping Tseng
yt2690
giping.t@columbia.edu.

problem 1.

①

	p	point	d(s)
	-1	s	0
V_2 ✗	~~1~~	V_1	~~∞~~ ~~4~~ 0
s	~~1~~	V_2	~~∞~~ 2
V_2	~~1~~	V_2	~~∞~~ 4
V_3	-1	t.	~~∞~~. 0

the shortest path from s → t is

s → V_2 → V_3 → t where the weights is 0 ✱

② ⓐ the shortest path from V_1 to V_3 is unknown. ✱

the information is not enough, if there is a path with smaller weight from V_1 to V_2, then the path found from the table is contradicted, so we don't know the shortest path from V_1 to V_3 according to the table ✱

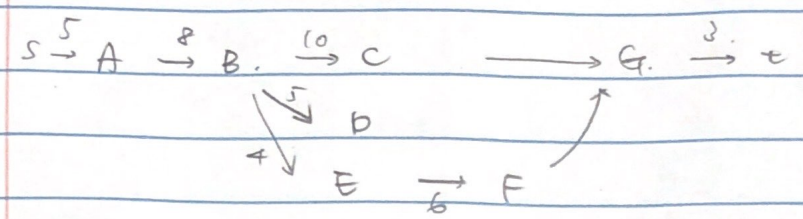
ⓑ the shortest path from V_2 to t is.

s → V_2 → V_2 → t W: 0.

s → V_2 W: 2

→ V_2 → V_3 → t W: -2 ✱

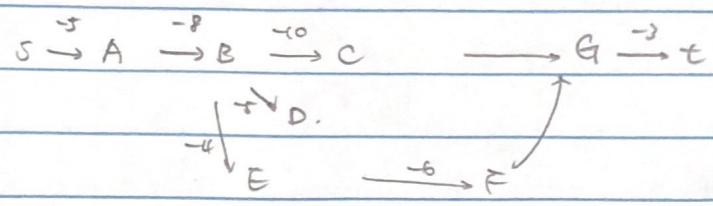
prob 2



to find the longest path :

$$\min \; -5T_{SA} - 8T_{AB} - 10T_{BC} - 5T_{BD} - 4T_{BE} - 6T_{EF} - 3T_{Gt}$$

→ Convert to shortest path problem



P	point	d(s).
~~→~~	S	0
S ~~→~~	A	~~∞~~ -5
A ~~→~~	B	~~∞~~ -13.
B ~~→~~	C	~~∞~~ -23.
B ~~→~~	D	~~∞~~ -18
B ~~→~~	E	~~∞~~ -17.
E ~~→~~	F	~~∞~~ -23.
C ~~→~~	G.	~~∞~~ -23.
G ~~→~~	t	~~∞~~ -26

the Critical path has two path

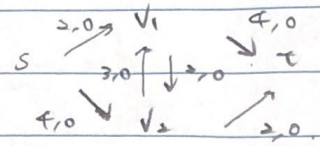
① S → A → B → C → G → t

② S → A → B → E → F → G → t

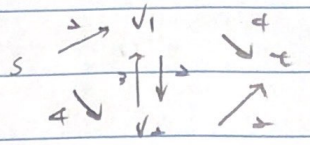
and weights are the same : -26 #

problem 3.

ori

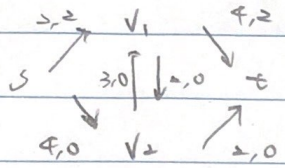


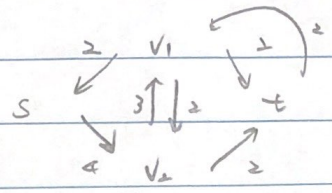
residual



path: $S \to V_1 \to t$

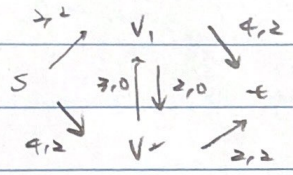
flow: $\min(2, 4) = 2$

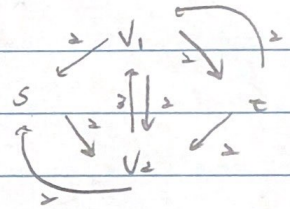




path: $S \to V_2 \to t$

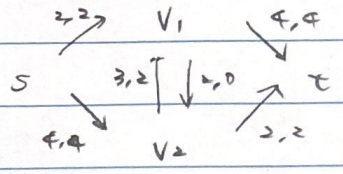
flow: $\min(4, 2) = 2$

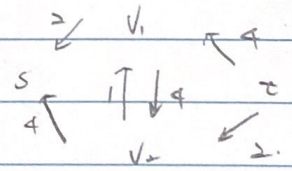




path: $S \to V_2 \to V_1 \to t$

flow: $\min(2, 3, 2) = 2$



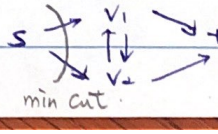


maximum flow: 6

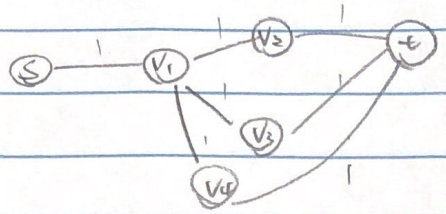
- $S \to V_1 \to t$ with 2
- $S \to V_2 \to t$ w/ 2
- $S \to V_2 \to V_1 \to t$ w/ 2

minimum cut: 6

- $S \to V_1$: w/ 2
- $S \to V_2$: w/ 4.



prob 4.



in this case, Although there are 3 maximum flow path, but there is only one minimum cut, which is $s \rightarrow v_1$

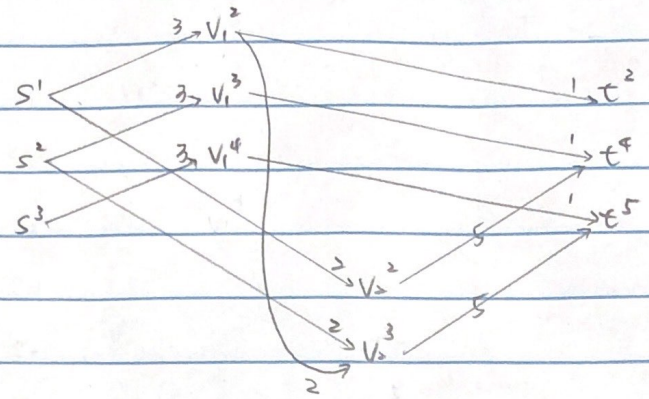
→ it violates the statement.

prob 5.

if the amount of increase doesn't have to be integer then the number of iteration is ∞.
Since we can always find an $\frac{\varepsilon}{2}$ improvement in the next iteration.

prob 6.

(a)



(b) t^3 : 1 (v_1 : 1 v_2 : 0)

t^4 : 3 (v_1 : 1 v_2 : 2)

t^5 : 5 (v_1 : 1 v_2 : 4).

maximum flow over time horizon $T=5$: 9

(c) flows are traversing arc (v_2, t) at time $t=3,4$

we should look at $(v_2^2, t^4) + (v_2^3, v^3) = 2+4 = 6$ #

prob 7.

M =

	A	B	C	D	E	F	G	outflow
A		40	30	-120	12	60	40	→ 62
B	-40		30	-70	15	-40	-12	→ -117
C	-30	-30		90	11	-30	60	→ 81
D	120	70	-90		40	-15	30	→ 145
E	-12	-15	-11	-40		-30	-30	→ -128
F	-60	40	30	15	30		70	→ 105
G	-40	12	-60	-70	30	-70		→ -148

X_{ij} := $ transfer from i bank to j bank

$$\min \quad 0.003 \sum_i \sum_j X_{ij}$$

s.t.

$$-\sum_j X_{aj} + \sum_j X_{ja} = -62 \quad \left(\text{outflow } \times(-1) + \text{inflow} = \text{outflow value} \right.$$

$$-\sum_j X_{bj} + \sum_j X_{jb} = 117$$

$$-\sum_j X_{cj} + \sum_j X_{jc} = -81$$

$$-\sum_j X_{dj} + \sum_j X_{jd} = -145$$

$$-\sum_j X_{ej} + \sum_j X_{je} = 128$$

$$-\sum_j X_{fj} + \sum_j X_{jf} = -105$$

$$-\sum_j X_{gj} + \sum_j X_{jg} = 148$$

$X_{ij} \geq 0$, $i = a, b, c, \ldots, g$; $j = a, b, \ldots, g$

obj value = 1.179 by gurobi

OPT HW4 problem 7

November 17, 2019

[ ]:
```
from gurobipy import *


# create a model
m = Model()

# create variables
ab = m.addVar(vtype=GRB.INTEGER, name="ab", lb=0)
ac = m.addVar(vtype=GRB.INTEGER, name="ac", lb=0)
da = m.addVar(vtype=GRB.INTEGER, name="da", lb=0)
ae = m.addVar(vtype=GRB.INTEGER, name="ae", lb=0)
af = m.addVar(vtype=GRB.INTEGER, name="af", lb=0)
ag = m.addVar(vtype=GRB.INTEGER, name="ag", lb=0)
bc = m.addVar(vtype=GRB.INTEGER, name="bc", lb=0)
db = m.addVar(vtype=GRB.INTEGER, name="db", lb=0)
be = m.addVar(vtype=GRB.INTEGER, name="be", lb=0)
fb = m.addVar(vtype=GRB.INTEGER, name="fb", lb=0)
gb = m.addVar(vtype=GRB.INTEGER, name="gb", lb=0)
cd = m.addVar(vtype=GRB.INTEGER, name="cd", lb=0)
ce = m.addVar(vtype=GRB.INTEGER, name="ce", lb=0)
fc = m.addVar(vtype=GRB.INTEGER, name="fc", lb=0)
cg = m.addVar(vtype=GRB.INTEGER, name="cg", lb=0)
de = m.addVar(vtype=GRB.INTEGER, name="de", lb=0)
fd = m.addVar(vtype=GRB.INTEGER, name="fd", lb=0)
dg = m.addVar(vtype=GRB.INTEGER, name="dg", lb=0)
fe = m.addVar(vtype=GRB.INTEGER, name="fe", lb=0)
ge = m.addVar(vtype=GRB.INTEGER, name="ge", lb=0)
fg = m.addVar(vtype=GRB.INTEGER, name="fg", lb=0)

ba = m.addVar(vtype=GRB.INTEGER, name="ba", lb=0)
ca = m.addVar(vtype=GRB.INTEGER, name="ca", lb=0)
ad = m.addVar(vtype=GRB.INTEGER, name="ad", lb=0)
ea = m.addVar(vtype=GRB.INTEGER, name="ea", lb=0)
fa = m.addVar(vtype=GRB.INTEGER, name="fa", lb=0)
ga = m.addVar(vtype=GRB.INTEGER, name="ga", lb=0)
cb = m.addVar(vtype=GRB.INTEGER, name="cb", lb=0)
bd = m.addVar(vtype=GRB.INTEGER, name="bd", lb=0)
```

```
eb = m.addVar(vtype=GRB.INTEGER, name="eb", lb=0)
bf = m.addVar(vtype=GRB.INTEGER, name="bf", lb=0)
bg = m.addVar(vtype=GRB.INTEGER, name="bg", lb=0)
dc = m.addVar(vtype=GRB.INTEGER, name="dc", lb=0)
ec = m.addVar(vtype=GRB.INTEGER, name="ec", lb=0)
cf = m.addVar(vtype=GRB.INTEGER, name="cf", lb=0)
gc = m.addVar(vtype=GRB.INTEGER, name="gc", lb=0)
ed = m.addVar(vtype=GRB.INTEGER, name="ed", lb=0)
df = m.addVar(vtype=GRB.INTEGER, name="df", lb=0)
gd = m.addVar(vtype=GRB.INTEGER, name="gd", lb=0)
ef = m.addVar(vtype=GRB.INTEGER, name="ef", lb=0)
eg = m.addVar(vtype=GRB.INTEGER, name="eg", lb=0)
gf = m.addVar(vtype=GRB.INTEGER, name="gf", lb=0)


# integrate new variables
m.update()

# set objective
m.setObjective(
    0.003*(ab + ac + da + ae + af + ag + bc + db + be + fb + gb + cd + ce + fc␣
 ↪+ cg + de + fd + dg + fe + ge + fg + ba + ca + ad + ea + fa + ga + cb + bd +␣
 ↪eb + bf + bg + dc + ec + cf + gc + ed + df + gd + ef + eg + gf),
    GRB.MINIMIZE
)

# add constraints
m.addConstr(-1*(ab + ac + ad + ae + af + ag) + ba + ca + da + ea + fa + ga ==␣
 ↪-1*(62))
m.addConstr(-1*(ba + bc + bd + be + bf + bg) + ab + cb + db + eb + fb + gb ==␣
 ↪-1*(-1*117))
m.addConstr(-1*(ca + cb + cd + ce + cf + cg) + ac + bc + dc + ec + fc + gc ==␣
 ↪-1*(81))
m.addConstr(-1*(da + db + dc + de + df + dg) + ad + bd + cd + ed + fd + gd ==␣
 ↪-1*(145))
m.addConstr(-1*(ea + eb + ec + ed + ef + eg) + ae + be + ce + de + fe + ge ==␣
 ↪-1*(-1*128))
m.addConstr(-1*(fa + fb + fc + fd + fe + fg) + af + bf + cf + df + ef + gf ==␣
 ↪-1*(105))
m.addConstr(-1*(ga + gb + gc + gd + ge + gf) + ag + bg + cg + dg + eg + fg ==␣
 ↪-1*(-1*148))

# optimize
m.optimize()
print("Model status: ", m.status)
```

```
# print out decision variables
for v in m.getVars():
    print(v.varName, v.x, "\n")

print("-"*15)
print("Obj Value: ", m.objVal)


'''
obj value: 1.179
ab 12.0
ae 47.0
ag 3.0
fb 105.0
ce 81.0
dg 145.0

ac 0.0
da 0.0
af 0.0
bc 0.0
db 0.0
be 0.0
gb 0.0
cd 0.0
fc 0.0
cg 0.0
de 0.0
fd 0.0
fe 0.0
ge 0.0
fg 0.0
ba 0.0
ca 0.0
ad 0.0
ea 0.0
fa 0.0
ga 0.0
cb 0.0
bd 0.0
eb 0.0
bf 0.0
bg 0.0
dc 0.0
ec 0.0
cf 0.0
gc 0.0
```

```
ed 0.0
df 0.0
gd 0.0
ef 0.0
eg 0.0
gf 0.0
'''
```