Proyecto Investigación de Operaciones

Javier Corral Lizárraga 188190
Sebastian Serra 180760

20 de abril de 2023

1. Cortes de Gomory

1.1. problema de cortes de Gomory

1.1.1. problema a) de cortes de Gomory

$$
\begin{aligned}
\mathbf{max}\ z &= 5x_1 + 9x_2 \\
s.a. \quad -x_1 + 5x_2 &\leq 3 \\
5x_1 + 3x_2 &\leq 27 \\
x_1, x_2 &\geq 0 \\
x_1, x_2 &\in \mathbb{Z}
\end{aligned}
$$

1.1.2. solucion de cortes de Gomory problema a)

max $5x_1 + 9x_2 + 0S_1 + 0S_2$

s.a. $-x_1 + 5x_2 + S_1 = 3$

$5x_1 + 3x_2 + S_2 = 27$

$x_1, x_2, S_1, S_2 \geq 0$

It. 1		C_j	5	9	0	0	
B	C_B	X_B	x_1	x_2	S_1	S_2	$\frac{X_B}{x_2}$
S_1	0	3	-1	(5)	1	0	$0.6 \rightarrow$
S_2	0	27	5	3	0	1	9
$Z = 0$		Z_j	0	0	0	0	
		$Z_j - C_j$	-5	$-9 \uparrow$	0	0	

Entra x_2 Sale S_1 $\Rightarrow$ $R_1 = \frac{1}{5} R_1$ y $R_2 = R_2 - 3R_1$

It. 2		C_j	5	9	0	0	
B	C_B	X_B	x_1	x_2	S_1	S_2	$\frac{X_B}{x_1}$
x_2	9	$\frac{3}{5}$	$-\frac{1}{5}$	1	$\frac{1}{5}$	0	—
S_2	0	$\frac{126}{5}$	$(\frac{28}{5})$	0	$-\frac{3}{5}$	1	$4.5 \rightarrow$
$Z = \frac{27}{5}$		Z_j	$-\frac{9}{5}$	9	$\frac{9}{5}$	0	
		$Z_j - C_j$	$-\frac{34}{5} \uparrow$	0	$\frac{9}{5}$	0	

Entra x_1 Sale S_2

$\Rightarrow R_2 = \frac{5}{28} R_2$ y $R_1 = R_1 + \frac{1}{5} R_2$

It. 1		C_j	5	9	0	0
B	C_B	X_B	x_1	x_2	S_1	S_2
x_2	9	$\frac{3}{2}$	0	1	$\frac{5}{28}$	$\frac{1}{28}$
x_1	5	$\frac{9}{2}$	1	0	$-\frac{3}{28}$	$\frac{5}{28}$
$Z = 36$		Z_j	5	9	$\frac{15}{14}$	$\frac{17}{14}$
		$Z_j - C_j$	0	0	$\frac{15}{14}$	$\frac{17}{14}$

Primer Corte

$$\frac{3}{2} = 1x_2 + \frac{5}{28}S_1 + \frac{1}{28}S_2$$

$$\left(1+\frac{1}{2}\right) = (1+0)x_2 + \left(0+\frac{5}{28}\right)S_1 + \left(0+\frac{1}{28}\right)S_2$$

$$-\frac{1}{2} = Sg1 - \frac{5}{28}S_1 - \frac{1}{28}S_2 \longrightarrow \text{Corte } 1$$

It. 1		C_j	5	9	0	0	0
B	C_B	X_B	x_1	x_2	S_1	S_2	$Sg\,1$
x_2	9	$\frac{3}{2}$	0	1	$\frac{5}{28}$	$\frac{1}{28}$	0
x_1	5	$\frac{9}{2}$	1	0	$-\frac{3}{28}$	$\frac{5}{28}$	0
$Sg1$	0	$-\frac{1}{2}$	0	0	$\left(-\frac{5}{28}\right)$	$-\frac{1}{28}$	1
$Z = 36$		Z_j	5	9	$\frac{15}{14}$	$\frac{17}{14}$	0
		$Z_j - C_j$	0	0	$\frac{15}{14}$	$\frac{17}{14}$	0
		$\frac{Z_j - C_j}{Sg1,j}$	—	—	$-6 \uparrow$	-34	—

Entra S_1 Sale $Sg1$

$$\Rightarrow R_3 = -\frac{28}{5}R_3 \;,\; R_1 = R_1 - \frac{5}{28}R_3 \;\text{ y }\; R_2 = R_2 + \frac{3}{28}R_3$$

It. 1		C_j	5	9	0	0	0
B	C_B	X_B	x_1	x_2	S_1	S_2	$Sg\,1$
x_2	9	1	0	1	0	0	1
x_1	5	$\frac{24}{5}$	1	0	0	$\frac{1}{5}$	$-\frac{3}{5}$
S_1	0	$\frac{14}{5}$	0	0	1	$\frac{1}{5}$	$-\frac{28}{5}$
$z=33$		z_j	5	9	0	1	6
		z_j-C_j	0	0	0	1	6
		$\frac{z_j-C_j}{Sg1,j}$	—	—	—	—	—

Segundo Corte

$$\frac{24}{5} = x_1 + \frac{1}{5}S_2 - \frac{3}{5}Sg\,1$$

$$\left(4+\frac{4}{5}\right) = (1+0)x_1 + \left(0+\frac{1}{5}\right)S_2 + \left(-1+\frac{2}{5}\right)Sg\,1$$

$$-\frac{4}{5} = Sg\,2 - \frac{1}{5}S_2 - \frac{2}{5}Sg\,1 \longrightarrow \text{Corte 2}$$

It. 1		C_j	5	9	0	0	0	0
B	C_B	X_B	x_1	x_2	S_1	S_2	$Sg\,1$	$Sg\,2$
x_2	9	1	0	1	0	0	1	0
x_1	5	$\frac{24}{5}$	1	0	0	$\frac{1}{5}$	$-\frac{3}{5}$	0
S_1	0	$\frac{14}{5}$	0	0	1	$\frac{1}{5}$	$-\frac{28}{5}$	0
$Sg\,2$	0	$-\frac{4}{5}$	0	0	0	$\left(-\frac{1}{5}\right)$	$-\frac{2}{5}$	1
$z=33$		z_j	5	9	0	1	6	0
		z_j-C_j	0	0	0	1	6	0
		Ratio	—	—	—	$-5\uparrow$	-15	—

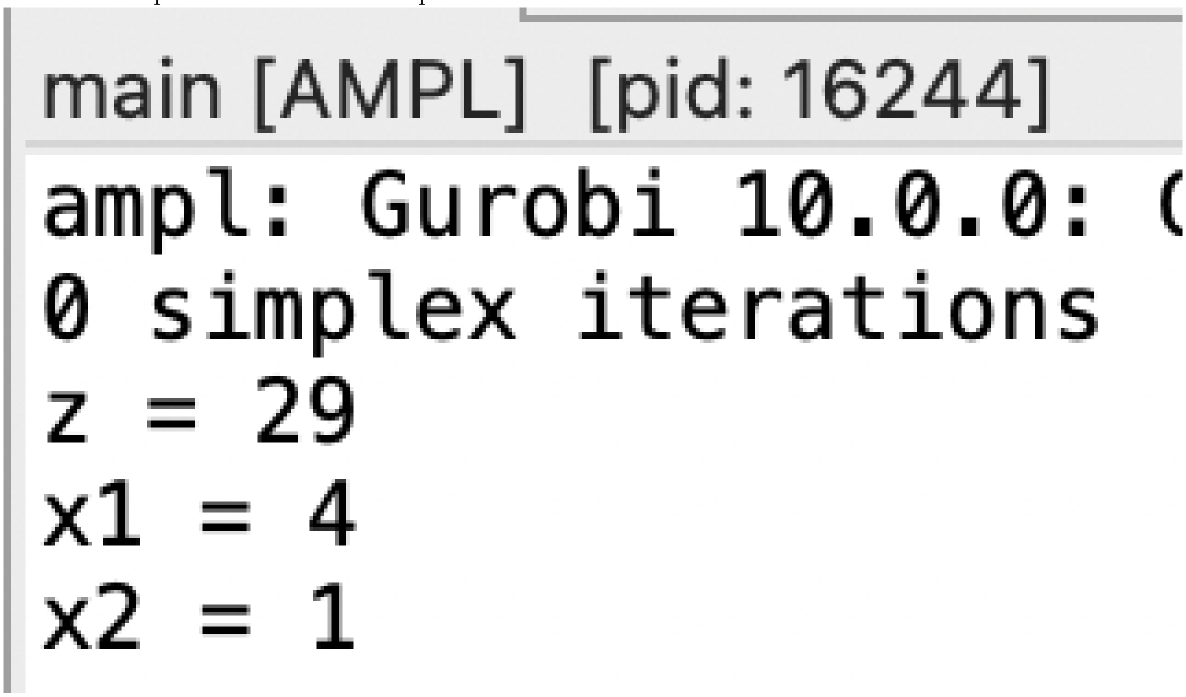
Entra S_2 Sale $S_g 2$ $\Rightarrow R_4 = -5 R_4$, $R_1 = R_1$, $R_2 = R_2 - \frac{1}{5} R_4$

y $R_3 = R_3 - \frac{1}{5} R_4$

It. 2		C_j	5	9	0	0	0	0
B	C_B	X_B	x_1	x_2	S_1	S_2	$S_g 1$	$S_g 2$
X_2	9	1	0	1	0	0	1	0
X_1	5	4	1	0	0	0	-1	1
S_1	0	2	0	0	1	0	-6	1
S_2	0	4	0	0	0	1	2	-5
$Z = 29$		Z_j	5	9	0	0	4	5
		$Z_j - C_j$	0	0	0	0	4	5
		Ratio	—	—	—	—	—	—

$Z_j - C_j \geq 0 \quad \forall j$ y $\bar{x} = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} 4 \\ 1 \end{pmatrix}$ con Max $Z = 29$

tambien comprobamos con AMPL que la solucion sea correcta



se adjunta el codigo aqui y en el zip en el archivo de nombre *gomorya.mod*

Listing 1: Flujo Maximo

```
var x1 >= 0 integer;
var x2 >= 0 integer;

maximize z:
5*x1 + 9*x2;

subject to c1: -x1 + 5*x2 <= 3;
subject to c2: 5*x1 + 3*x2 <= 27;

#model gomorya.mod;
option solver gurobi;
solve;
display z,x1,x2;
```

1.1.3. problema b) de cortes de Gomory

b)

$$\begin{array}{lr} \max & Z = 5x_1 + 4x_2 + 4x_3 + 2x_4 \\ \text{sujeto a} & x_1 + 3x_2 + 2x_3 + x_4 \leq 10 \\ & 5x_1 + x_2 + 3x_3 + 2x_4 \leq 15 \\ & x_1 + x_2 + x_3 + x_4 \leq 6 \\ & x_1, x_2, x_3, x_4 \geq 0\ ,\ x_1, x_2, x_3, x_4 \in \mathbb{Z}\,. \end{array}$$

1.1.4. solucion de cortes de Gomory problema b)

$$\max \quad z = 5x_1 + 4x_2 + 4x_3 + 2x_4 + 0S_1 + 0S_2 + 0S_3$$

s.a

$$x_1 + 3x_2 + 2x_3 + x_4 + S_1 = 10$$
$$5x_1 + x_2 + 3x_3 + 2x_4 + S_2 = 15$$
$$x_1 + x_2 + x_3 + x_4 + S_3 = 6$$
$$x_1, x_2, x_3, x_4, S_1, S_2, S_3 \geq 0$$

It. 1		C_j	5	4	4	2	0	0	0	
B	C_B	X_B	x_1	x_2	x_3	x_4	S_1	S_2	S_3	$\frac{X_B}{x_1}$
S_1	0	10	1	3	2	1	1	0	0	10
S_2	0	15	(5)	1	3	2	0	1	0	$\frac{15}{5} = 3 \rightarrow$
S_3	0	6	1	1	1	1	0	0	1	$\frac{6}{1} = 6$
$Z = 0$		Z_j	0	0	0	0	0	0	0	
		$Z_j - C_j$	$-5\uparrow$	-4	-4	-2	0	0	0	

Entra x_1 Sale S_2 $\Rightarrow$ $R_2 = \frac{1}{5} R_2$, $R_1 = R_1 - R_2$ y $R_3 = R_3 - R_2$

It. 2		C_j	5	4	4	2	0	0	0	
B	C_B	X_B	x_1	x_2	x_3	x_4	S_1	S_2	S_3	$\frac{X_B}{x_2}$
S_1	0	7	0	$\left(\frac{14}{5}\right)$	$\frac{7}{5}$	$\frac{3}{5}$	1	$-\frac{1}{5}$	0	$2.5 \rightarrow$
x_1	5	3	1	$\frac{1}{5}$	$\frac{3}{5}$	$\frac{2}{5}$	0	$\frac{1}{5}$	0	15
S_3	0	3	0	$\frac{4}{5}$	$\frac{2}{5}$	$\frac{3}{5}$	0	$-\frac{1}{5}$	1	3.75
$Z = 15$		Z_j	5	1	3	2	0	1	0	
		$Z_j - C_j$	0	$-3\uparrow$	-1	0	0	1	0	

Entra x_2 Sale S_1 $\Rightarrow$ $R_1 = \frac{5}{14} R_1$, $R_2 = R_2 - \frac{1}{5} R_1$ y $R_3 = R_3 - \frac{4}{5} R_1$

It. 3		C_j	5	4	4	2	0	0	0
B	C_B	X_B	x_1	x_2	x_3	x_4	s_1	s_2	s_3
x_2	4	$\frac{5}{2}$	0	1	$\frac{1}{2}$	$\frac{3}{14}$	$\frac{5}{14}$	$-\frac{1}{14}$	0
x_1	5	$\frac{5}{2}$	1	0	$\frac{1}{2}$	$\frac{5}{14}$	$-\frac{1}{14}$	$\frac{3}{14}$	0
s_3	0	1	0	0	0	$\frac{3}{7}$	$-\frac{2}{7}$	$-\frac{1}{7}$	1
$z = \frac{45}{2}$		z_j	5	4	$\frac{9}{2}$	$\frac{37}{14}$	$\frac{15}{14}$	$\frac{11}{14}$	0
		$z_j - c_j$	0	0	$\frac{1}{2}$	$\frac{9}{14}$	$\frac{15}{14}$	$\frac{11}{14}$	0

Como $z_j - c_j \geq 0 \quad \forall j$

llegamos a $\bar{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 5/2 \\ 5/2 \\ 0 \\ 0 \end{pmatrix}$ Max $z = \frac{45}{2}$, pero no son sol. enteras.

Hacemos un corte de Gomory

$$\frac{5}{2} = x_2 + \frac{1}{2}x_3 + \frac{3}{14}x_4 + \frac{5}{14}s_1 - \frac{1}{14}s_2$$

$$\left(2 + \frac{1}{2}\right) = (1+0)\,x_2 + \left(0 + \frac{1}{2}\right)x_3 + \left(0 + \frac{3}{14}\right)x_4 + \left(0 + \frac{5}{14}\right)s_1 + \left(-1 + \frac{13}{14}\right)s_2$$

El corte fraccionario será:

$$-\frac{1}{2} = s_{g1} - \frac{1}{2}x_3 - \frac{3}{14}x_4 - \frac{5}{14}s_1 - \frac{13}{14}s_2 \longrightarrow \text{(Corte 1)}$$

$\downarrow$

Nueva Tabla

It. 1		C_j	5	4	4	2	0	0	0	0
B	C_B	X_B	x_1	x_2	x_3	x_4	S_1	S_2	S_3	$Sg1$
x_2	4	$\frac{5}{2}$	0	1	$\frac{1}{2}$	$\frac{3}{14}$	$\frac{5}{14}$	$-\frac{1}{14}$	0	0
x_1	5	$\frac{5}{2}$	1	0	$\frac{1}{2}$	$\frac{5}{14}$	$-\frac{1}{14}$	$\frac{3}{14}$	0	0
S_3	0	1	0	0	0	$\frac{3}{7}$	$-\frac{2}{7}$	$-\frac{1}{7}$	1	0
$Sg1$	0	$-\frac{1}{2}$	0	0	$-\frac{1}{2}$	$-\frac{3}{14}$	$-\frac{5}{14}$	$\left(-\frac{13}{14}\right)$	0	1
$Z = \frac{45}{2}$		z_j	5	4	$\frac{9}{2}$	$\frac{37}{14}$	$\frac{15}{14}$	$\frac{11}{14}$	0	0
		$z_j - C_j$	0	0	$\frac{1}{2}$	$\frac{9}{14}$	$\frac{15}{14}$	$\frac{11}{14}$	0	0
	$\frac{z_j - C_j}{Sg1,j}$		–	–	-1	-3	-3	-0.8462	–	–

Entra S_2 y Sale $Sg1$ $\Rightarrow R_4 = -\frac{14}{13}R_4$, $R_1 = R_1 + \frac{1}{14}R_4$, $R_2 = R_2 - \frac{3}{14}R_4$ y $R_3 = R_3 + \frac{1}{7}R_4$

It. 2		C_j	5	4	4	2	0	0	0	0
B	C_B	X_B	x_1	x_2	x_3	x_4	S_1	S_2	S_3	$Sg1$
x_2	4	$\frac{33}{13}$	0	1	$\frac{7}{13}$	$\frac{3}{13}$	$\frac{5}{13}$	0	0	$-\frac{1}{13}$
x_1	5	$\frac{31}{13}$	1	0	$\frac{5}{13}$	$\frac{4}{13}$	$-\frac{2}{13}$	0	0	$\frac{3}{13}$
S_3	0	$\frac{14}{13}$	0	0	$\frac{1}{13}$	$\frac{6}{13}$	$-\frac{3}{13}$	0	1	$-\frac{2}{13}$
S_2	0	$\frac{7}{13}$	0	0	$\frac{7}{13}$	$\frac{3}{13}$	$\frac{5}{13}$	1	0	$-\frac{14}{13}$
$Z = \frac{287}{13}$		z_j	5	4	$\frac{53}{13}$	$\frac{72}{13}$	$\frac{10}{13}$	0	0	$\frac{11}{13}$
		$z_j - C_j$	0	0	$\frac{1}{13}$	$\frac{6}{13}$	$\frac{10}{13}$	0	0	$\frac{11}{13}$
	$\frac{z_j - C_j}{Sg1,j}$		–	–	–	–	–	–	–	–

Ahora con $\bar{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 31/13 \\ 33/13 \\ 0 \\ 0 \end{pmatrix}$ Max $z = \frac{287}{13}$

Construimos otro corte de Gomory

$$\frac{33}{13} = x_2 + \frac{7}{13} x_3 + \frac{3}{13} x_4 + \frac{5}{13} s_1 - \frac{1}{13} Sg1$$

$$\left(2 + \frac{7}{13}\right) = (1+0) x_2 + \left(0 + \frac{7}{13}\right) x_{13} + \left(0 + \frac{3}{13}\right) x_4 + \left(-1 + \frac{12}{13}\right) Sg1$$

$$-\frac{7}{13} = Sg2 - \frac{7}{13} x_3 + \frac{3}{13} x_4 - \frac{5}{13} s_1 - \frac{12}{17} Sg1 \longrightarrow \text{(Corte 2)}$$

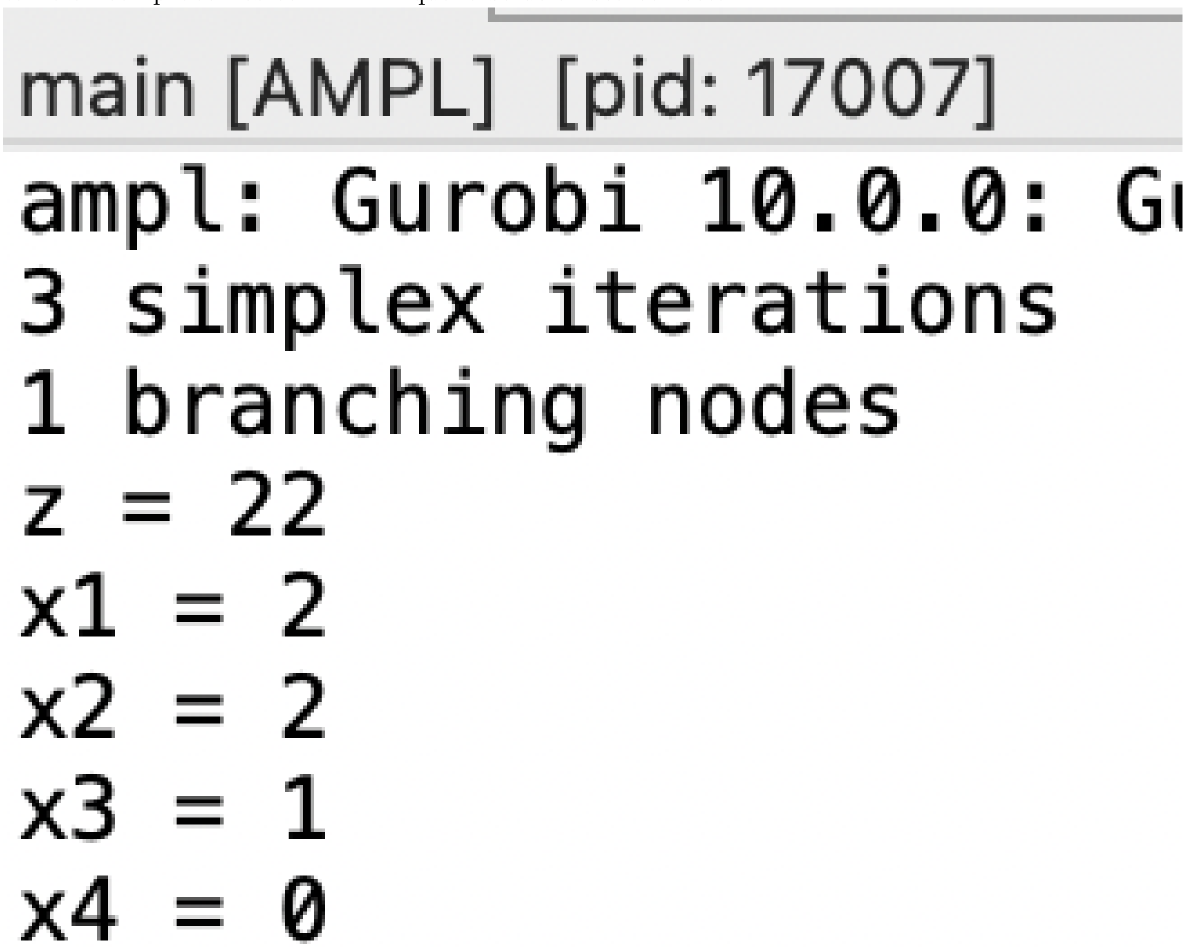
It. 1		C_j	5	4	4	2	0	0	0	0	0
B	C_B	X_B	x_1	x_2	x_3	x_4	s_1	s_2	s_3	$Sg1$	$Sg2$
x_2	4	$\frac{33}{13}$	0	1	$\frac{7}{13}$	$\frac{3}{13}$	$\frac{5}{13}$	0	0	$-\frac{1}{13}$	0
x_1	5	$\frac{31}{13}$	1	0	$\frac{5}{13}$	$\frac{4}{13}$	$-\frac{2}{13}$	0	0	$\frac{3}{13}$	0
s_3	0	$\frac{14}{13}$	0	0	$\frac{1}{13}$	$\frac{6}{13}$	$-\frac{3}{13}$	0	1	$-\frac{2}{13}$	0
$Sg1$	0	$\frac{7}{13}$	0	0	$\frac{7}{13}$	$\frac{3}{13}$	$\frac{5}{13}$	1	0	$-\frac{14}{13}$	0
$Sg2$	0	$-\frac{7}{13}$	0	0	$\left(-\frac{7}{13}\right)$	$-\frac{3}{13}$	$-\frac{5}{13}$	0	0	$-\frac{12}{13}$	1
$z = \frac{287}{13}$		z_j	5	4	$\frac{53}{13}$	$\frac{32}{13}$	$\frac{10}{13}$	0	0	$\frac{11}{13}$	0
		$z_j - C_j$	0	0	$\frac{1}{13}$	$\frac{6}{13}$	$\frac{10}{13}$	0	0	$\frac{11}{13}$	0
		$\frac{z_j - C_j}{Sg1,j}$	—	—	-6.1429 ↑	-2	-2	—	—	-0.9167	—

Entra x_3 Sale $Sg2$ $\Rightarrow R_5 = -\frac{13}{7} R_5$, $R_1 = R_1 - \frac{7}{13} R_5$, $R_2 = R_2 - \frac{5}{13} R_5$

It. 1		C_j	5	4	4	2	0	0	0	0	0
B	C_B	X_B	x_1	x_2	x_3	x_4	S_1	S_2	S_3	$Sg\,1$	$Sg\,2$
x_2	4	2	0	1	0	0	0	0	0	-1	1
x_1	5	2	1	0	0	$\frac{1}{7}$	$-\frac{3}{7}$	0	0	$-\frac{3}{7}$	$\frac{5}{7}$
S_3	0	1	0	0	0	$\frac{3}{7}$	$-\frac{2}{7}$	0	1	$-\frac{2}{7}$	$\frac{1}{7}$
S_2	0	0	0	0	0	0	0	1	0	-2	1
x_3	4	1	0	0	1	$\frac{3}{7}$	$\frac{5}{7}$	0	0	$\frac{12}{7}$	$-\frac{13}{7}$
$Z = 22$		z_j	5	4	4	$\frac{17}{7}$	$\frac{5}{7}$	0	0	$\frac{5}{7}$	$\frac{1}{7}$
		$z_j - C_j$	0	0	0	$\frac{3}{7}$	$\frac{5}{7}$	0	0	$\frac{5}{7}$	$\frac{1}{7}$
		$\frac{z_j - C_j}{Sg1_{,j}}$	—	—	—	—	—	—	—	—	—

Ahora sí $\bar{x} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \\ 1 \\ 0 \end{pmatrix}$ $\quad$ $\text{Max } Z = 22$ $\quad$ Con 2 cortes.

tambien comprobamos con AMPL que la solucion sea correcta



se adjunta el codigo aqui y en el zip con el nombre *gomoryb.mod*

Listing 2: Flujo Maximo

```
var x1 >= 0 integer;
var x2 >= 0 integer;
var x3 >= 0 integer;
var x4 >= 0 integer;

maximize z:
5*x1 + 4*x2 + 4*x3 + 2*x4;

subject to c1: x1 + 3*x2 + 2*x3 + x4 <= 10;
subject to c2: 5*x1 + x2 + 3*x3 + 2*x4 <= 15;
subject to c3: x1 + x2 + x3 + x4 <=  6;


#model gomoryb.mod;
option solver gurobi;
solve;
display z,x1,x2,x3,x4;
```

2. Programación Entera

2.1. problema 1 de programacion entera

2. **Programación Entera:**

- Una empresa de bienes raíces, Peterson & Johnson, analiza cinco proyectos de desarrollo posibles. La siguiente tabla muestra las ganancias a largo plazo estimadas (valor presente neto) que generaría cada proyecto y la inversión que se requiere para emprenderlo, en millones de dólares.

	Proyecto de desarrollo				
	1	**2**	**3**	**4**	**5**
Ganancia Estimada	1	1.8	1.6	0.8	1.4
Capital Requerido	6	12	10	4	8

Los propietarios de la empresa, Dave Peterson y Ron Johnson, reunieron $20 millones de capital de inversión para estos proyectos. Ellos quieren elegir la combinación de proyectos que maximice la ganancia total estimada a largo plazo (valor presente neto) sin invertir más de $20 millones.

a) Formule un modelo de Programación Entera Binaria para este problema.

b) Muestre el modelo en una hoja de cálculo de Excel.

c) Use la computadora para resolver este modelo.

2.2. solucion 1 de programacion entera

a)

$$\begin{aligned} \mathbf{max}\ z &= 1x_1 + 1{,}8x_2 + 1{,}6x_3 + 0{,}8x_4 + 1{,}4x_5 \\ s.a.\quad 6x_1 + 12x_2 + 10x_3 + 4x_4 + 8x_5 &\leq 20 \\ x_1, x_2, x_3, x_4, x_5 &\in \{0, 1\} \end{aligned}$$

donde x_i es 1 si se invierte en el proyecto i y 0 en caso contrario

b) y c)

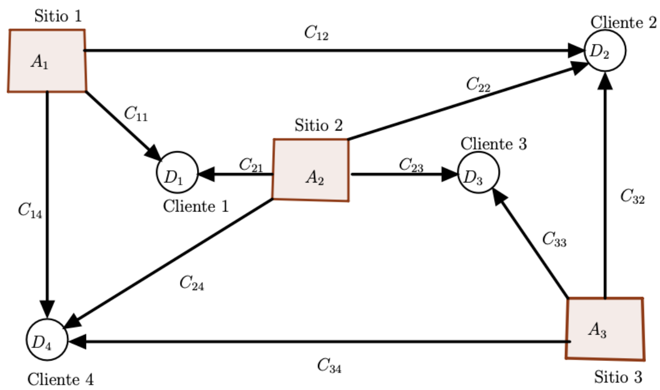
lo resolvemos con excel y da lo siguiente:

variables	valor	ganancia est	capital req
x1	1	1	6
x2	0	1.8	12
x3	1	1.6	10
x4	1	0.8	4
x5	0	1.4	8
funcion objetivo			
max	3.4		
sujeto a			
	20	<=	20

se adjunta el excel en el zip en el archivo con nombre *proen1.xlsx*

2.3. problema 2 de programacion entera

- Una empresa minorista planea expandir sus actividades en un área mediante la apertura de dos nuevos almacenes. Se están considerando tres posibles sitios, como se muestra en la siguiente figura.



Hay que abastecer a cuatro clientes cuyas demandas son D_1, D_2, D_3 y D_4. Suponga que dos sitios cualquiera pueden satisfacer todas las demandas, pero el sitio 1 solo puede satisfacer a los clientes 1, 2 y 4; el sitio 3 puede abastecer a los clientes 2, 3 y 4; mientras que el sitio 2 puede abastecer a todos los clientes. El costo unitario de transporte del sitio i al cliente j es Cij. Para cada sitio de almacén tenemos los siguientes datos:

Sitio	**Capacidad**	**Inversión de Capital Inicial $**	**Costo unitario de operación $**
1	A_1	K_1	P_1
2	A_2	K_2	P_2
3	A_3	K_3	P_3

El problema de optimización es elegir los sitios adecuados para los dos almacenes que minimicen el costo total de inversión, de operación y de transporte.

a) Introduzca variables binarias auxiliares para diseñar un modelo de PEB mixta de este problema.

b) Una vez planteado su modelo, introduzca los siguientes datos y resuelvalo con MATLAB o AMPL.

$$D_1 = 824, D_2 = 735, D_3 = 683, D_4 = 575,\ \ P_1 = 71.39, P_2 = 100.63, P_3 = 50.63$$

$$A_1 = 2538, A_2 = 3604, A_3 = 2533,\ \ K_1 = 187547, K_2 = 471471, K_3 = 167719$$

$$C_{12} = C_{34} = 86,\ C_{14} = C_{32} = 65,\ C_{24} = C_{22} = 72,\ C_{21} = 36, C_{23} = 40, C_{11} = 54, C_{33} = 57$$

2.4. solucion 2 de programacion entera

a)

sean:
$costo\ de\ inversion = B_1K_1 + B_2K_2 + B_3K_3$

$$costo\ de\ operacion = B_1P_1 + B_2P_2 + B_3P_3$$

$$\begin{aligned} costo\ de\ transporte = {} & B_1C_{11} + B_1C_{12} + B_1C_{14} + B_2C_{21} + B_2C_{22} + B_2C_{23} \\ & +B_2C_{24} + B_3C_{32} + B_3C_{33} + B_3C_{34} \end{aligned}$$

$$\begin{aligned} \textbf{min } z = costo\ de\ inversion + costo\ de\ operacion + costo\ de\ transporte \\ s.a. \quad B_1A_1 &<= D_1 + D_2 + D_4 \\ B_2A_2 &<= D_2 + D_3 + D_4 \\ B_3A_3 &<= D_1 + D_2 + D_3 + D_4 \\ B_1A_1 + B_2A_2 + B_3A_3 &>= D_1 + D_2 + D_3 + D_4 \\ B_1A_1 + B_2A_2 &>= D_1 \\ B_1A_1 + B_2A_2 + B_3A_3 &>= D_2 \\ B_2A_2 + B_3A_3 &>= D_3 \\ B_1A_1 + B_2A_2 + B_3A_3 &>= D_4 \\ B_1 + B_2 + B_3 &= 2 \\ B_1, B_2, B_3 &\in \{0,1\} \end{aligned}$$

donde B_i es 1 si se pondra un almacen en el sitio i y 0 en caso contrario

b)

Ahora lo introducimos a AMPL y nos da el siguiente resultado

```
ampl: presolve, constraint c1:
        all variables eliminated, but upper bound = -404 < 0
Infeasible constraints determined by presolve.
z = 355801
B1 = 1
B2 = 0
B3 = 1
```

Adjunto el codigo aqui y en el zip en el archivo de nombre *proen2b.mod*

Listing 3: Flujo Maximo

```
param D1 := 824;
param D2 := 735;
param D3 := 683;
param D4 := 575;
param P1 := 71.39;
param P2 := 100.63;
param P3 := 50.63;
param A1 := 2538;
param A2 := 3604;
param A3 := 2533;
param K1 := 187547;
param K2 := 471471;
param K3 := 167719;
param C12 := 86;
param C34 := 86;
param C14 := 65;
param C32 := 65;
param C24 := 72;
```

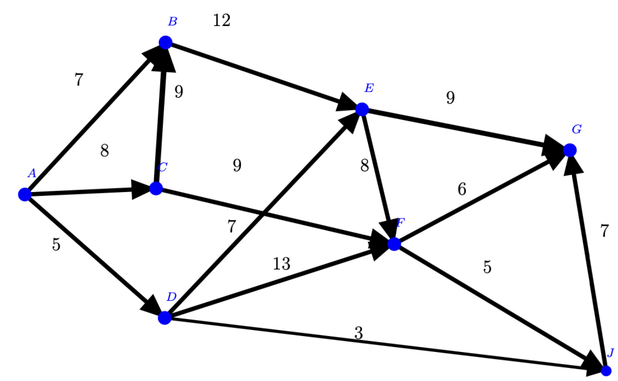
```
param C22 := 72;
param C21 := 36;
param C23 := 40;
param C11 := 54;
param C33 := 57;
var B1 binary;
var B2 binary;
var B3 binary;

minimize z:
B1*K1 + B2*K2 + B3*K3 + B1*P1 + B2*P2 + B3*P3 + B1*C11
+ B1*C12 + B1*C14 + B2*C21 + B2*C22 + B2*C23 + B2*C24
+ B3*C32 + B3*C33 + B3*C34;

# restriccion de capacidad
subject to c1: B1*A1 <= D1 + D2 + D4;
subject to c2: B2*A2 <= D2 + D3 + D4;
subject to c3: B3*A3 <= D1 + D2 + D3 + D4;
# restriccion demanda
subject to c4: B1*A1 + B2*A2 + B3*A3 >= D1 + D2 + D3 + D4;
subject to c5: B1*A1 + B2*A2 >= D1;
subject to c6: B1*A1 + B2*A2 + B3*A3 >= D2;
subject to c7: B2*A2 + B3*A3 >= D3;
subject to c8: B1*A1 + B2*A2 + B3*A3 >= D4;
# restriccion 2 almacenes
subject to c9: B1+B2+B3 = 2;

solve;

display z, B1, B2, B3;
```

3. Redes

3.1. problema a) de redes

3. **Redes:**

   a) Ocupe el algoritmo de Floyd que es más general que el algoritmo de Dijkstra para determinar la distancia más corta entre dos nodos cualesquiera de la siguiente red empezando en el nodo A y terminando en el J.



3.2. solucion a) de redes

Algo de Floyd para ejercicio 3. Redes a)

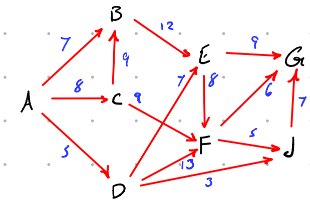
Matriz de distancias y secuencias

D_0

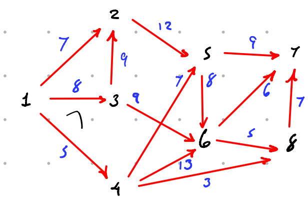
	1	2	3	4	5	6	7	8
1	–	7	8	5	∞	∞	∞	∞
2	∞	–	∞	∞	12	∞	∞	∞
3	∞	9	–	∞	∞	9	∞	∞
4	∞	∞	∞	–	7	13	∞	3
5	∞	∞	∞	∞	–	8	9	∞
6	∞	∞	∞	∞	∞	–	6	5
7	∞	∞	∞	∞	∞	∞	–	∞
8	∞	∞	∞	∞	∞	∞	7	–

S_0

	1	2	3	4	5	6	7	8
1	–	2	3	4	5	6	7	8
2	1	–	3	4	5	6	7	8
3	1	2	–	4	5	6	7	8
4	1	2	3	–	5	6	7	8
5	1	2	3	4	–	6	7	8
6	1	2	3	4	5	–	7	8
7	1	2	3	4	5	6	–	8
8	1	2	3	4	5	6	7	–



Le cambiaremos de nombre a los nodos



El mapeo es el siguiente

1 → A

2 → B

3 → C

4 → D

5 → E

6 → F

7 → G

8 → J

Iteración 1 ($k = 1$)

Las celdas que satisfacen:

$d_{i1} + d_{1j} < d_{ij}$ con $i, j \in \{2, 3, 4, 5, 6, 7, 8\} = \ell$

Son $\emptyset$ ya que $d_{1j} = \infty \ \forall j \in \ell$

D_1

	1	2	3	4	5	6	7	8
1	–	7	8	5	∞	∞	∞	∞
2	∞	–	∞	∞	12	∞	∞	∞
3	∞	9	–	∞	∞	9	∞	∞
4	∞	∞	∞	–	7	13	∞	3
5	∞	∞	∞	∞	–	8	9	∞
6	∞	∞	∞	∞	∞	–	6	5
7	∞	∞	∞	∞	∞	∞	–	∞
8	∞	∞	∞	∞	∞	∞	7	–

S_1

	1	2	3	4	5	6	7	8
1	–	2	3	4	5	6	7	8
2	1	–	3	4	5	6	7	8
3	1	2	–	4	5	6	7	8
4	1	2	3	–	5	6	7	8
5	1	2	3	4	–	6	7	8
6	1	2	3	4	5	–	7	8
7	1	2	3	4	5	6	–	8
8	1	2	3	4	5	6	7	–

Iteración 2 ($k = 2$)

Las celdas que satisfacen:

$d_{i2} + d_{2j} < d_{ij}$ con $i, j \in \{1, 3, 4, 5, 6, 7, 8\} = \ell$

Son d_{15} y d_{35} así que hacemos los reemplazos y modificamos S_2

D_2

	1	2	3	4	5	6	7	8
1	–	7	8	5	19	∞	∞	∞
2	∞	–	∞	∞	12	∞	∞	∞
3	∞	9	–	∞	21	9	∞	∞
4	∞	∞	∞	–	7	13	∞	3
5	∞	∞	∞	∞	–	8	9	∞
6	∞	∞	∞	∞	∞	–	6	5
7	∞	∞	∞	∞	∞	∞	–	∞
8	∞	∞	∞	∞	∞	∞	7	–

S_2

	1	2	3	4	5	6	7	8
1	–	2	3	4	2	6	7	8
2	1	–	3	4	5	6	7	8
3	1	2	–	4	2	6	7	8
4	1	2	3	–	5	6	7	8
5	1	2	3	4	–	6	7	8
6	1	2	3	4	5	–	7	8
7	1	2	3	4	5	6	–	8
8	1	2	3	4	5	6	7	–

Iteracion 3 $(k=3)$

Las celdas que satisfacen:

$d_{i3}+d_{3j}<d_{ij}$ con $i,j\in\{1,2,4,5,6,7,8\}=\ell$

Son d_{16}

D_3

	1	2	3	4	5	6	7	8
1	–	7	8	5	19	17	∞	∞
2	∞	–	∞	∞	12	∞	∞	∞
3	∞	9	–	∞	21	9	∞	∞
4	∞	∞	∞	–	7	13	∞	3
5	∞	∞	∞	∞	–	8	9	∞
6	∞	∞	∞	∞	∞	–	6	5
7	∞	∞	∞	∞	∞	∞	–	∞
8	∞	∞	∞	∞	∞	∞	7	–

S_3

	1	2	3	4	5	6	7	8
1	–	2	3	4	2	3	7	8
2	1	–	3	4	5	6	7	8
3	1	2	–	4	2	6	7	8
4	1	2	3	–	5	6	7	8
5	1	2	3	4	–	6	7	8
6	1	2	3	4	5	–	7	8
7	1	2	3	4	5	6	–	8
8	1	2	3	4	5	6	7	–

Iteracion 4 $(k=4)$

Las celdas que satisfacen:

$d_{i4}+d_{4j}<d_{ij}$ con $i,j\in\{1,2,3,5,6,7,8\}=\ell$

Son d_{15} y d_{18}

D_4

	1	2	3	4	5	6	7	8
1	–	7	8	5	12	17	∞	8
2	∞	–	∞	∞	12	∞	∞	∞
3	∞	9	–	∞	21	9	∞	∞
4	∞	∞	∞	–	7	13	∞	3
5	∞	∞	∞	∞	–	8	9	∞
6	∞	∞	∞	∞	∞	–	6	5
7	∞	∞	∞	∞	∞	∞	–	∞
8	∞	∞	∞	∞	∞	∞	7	–

S_4

	1	2	3	4	5	6	7	8
1	–	2	3	4	4	3	7	4
2	1	–	3	4	5	6	7	8
3	1	2	–	4	2	6	7	8
4	1	2	3	–	5	6	7	8
5	1	2	3	4	–	6	7	8
6	1	2	3	4	5	–	7	8
7	1	2	3	4	5	6	–	8
8	1	2	3	4	5	6	7	–

Iteracion 5 $(k=5)$

Las celdas que satisfacen:

$d_{i5}+d_{5j}<d_{ij}$ con $i,j\in\{1,2,3,4,6,7,8\}=\ell$

Son d_{17}, d_{27}, d_{37}, d_{47} y d_{27}

D_5

	1	2	3	4	5	6	7	8
1	–	7	8	5	12	17	21	8
2	∞	–	∞	∞	12	20	21	∞
3	∞	9	–	∞	21	9	30	∞
4	∞	∞	∞	–	7	13	16	3
5	∞	∞	∞	∞	–	8	9	∞
6	∞	∞	∞	∞	∞	–	6	5
7	∞	∞	∞	∞	∞	∞	–	∞
8	∞	∞	∞	∞	∞	∞	7	–

S_5

	1	2	3	4	5	6	7	8
1	–	2	3	4	4	3	5	4
2	1	–	3	4	5	5	5	8
3	1	2	–	4	2	6	5	8
4	1	2	3	–	5	6	5	8
5	1	2	3	4	–	6	7	8
6	1	2	3	4	5	–	7	8
7	1	2	3	4	5	6	–	8
8	1	2	3	4	5	6	7	–

Iteracion 6 ($k=6$)

Las celdas que satisfacen:

$$d_{i6}+d_{6j}<d_{ij} \quad \text{con } i,j\in\{1,2,3,4,5,7,8\}=\ell$$

Son d_{28}, d_{38}, d_{58} y d_{37}

D_6

	1	2	3	4	5	6	7	8
1	–	7	8	5	12	17	21	8
2	∞	–	∞	∞	12	20	21	25
3	∞	9	–	∞	21	9	15	14
4	∞	∞	∞	–	7	13	16	3
5	∞	∞	∞	∞	–	8	9	13
6	∞	∞	∞	∞	∞	–	6	5
7	∞	∞	∞	∞	∞	∞	–	∞
8	∞	∞	∞	∞	∞	∞	7	–

S_6

	1	2	3	4	5	6	7	8
1	–	2	3	4	4	3	5	4
2	1	–	3	4	5	5	5	6
3	1	2	–	4	2	6	6	6
4	1	2	3	–	5	6	5	8
5	1	2	3	4	–	6	7	6
6	1	2	3	4	5	–	7	8
7	1	2	3	4	5	6	–	8
8	1	2	3	4	5	6	7	–

Iteracion 7 ($k=7$)

Las celdas que satisfacen:

$$d_{i7}+d_{7j}<d_{ij} \quad \text{con } i,j\in\{1,2,3,4,5,6,8\}=\ell$$

Son $\emptyset$ porque $d_{7j}=\infty \ \forall j\in\ell$

D_7

	1	2	3	4	5	6	7	8
1	–	7	8	5	12	17	21	8
2	∞	–	∞	∞	12	20	21	25
3	∞	9	–	∞	21	9	15	14
4	∞	∞	∞	–	7	13	16	3
5	∞	∞	∞	∞	–	8	9	1
6	∞	∞	∞	∞	∞	–	6	5
7	∞	∞	∞	∞	∞	∞	–	∞
8	∞	∞	∞	∞	∞	∞	7	–

S_7

	1	2	3	4	5	6	7	8
1	–	2	3	4	4	3	5	4
2	1	–	3	4	5	5	5	6
3	1	2	–	4	2	6	6	6
4	1	2	3	–	5	6	5	8
5	1	2	3	4	–	6	7	6
6	1	2	3	4	5	–	7	8
7	1	2	3	4	5	6	–	8
8	1	2	3	4	5	6	7	–

Iteracion 8 ($k=8$)

Las celdas que satisfacen:

$$d_{i8}+d_{8j}<d_{ij} \quad \text{con } i,j\in\{1,2,3,4,5,6,7\}=\ell$$

Son d_{17}, d_{47} y d_{57}

D_8

	1	2	3	4	5	6	7	8
1	–	7	8	5	12	17	15	8
2	∞	–	∞	∞	12	20	21	25
3	∞	9	–	∞	21	9	15	14
4	∞	∞	∞	–	7	13	10	3
5	∞	∞	∞	∞	–	8	8	1
6	∞	∞	∞	∞	∞	–	6	5
7	∞	∞	∞	∞	∞	∞	–	∞
8	∞	∞	∞	∞	∞	∞	7	–

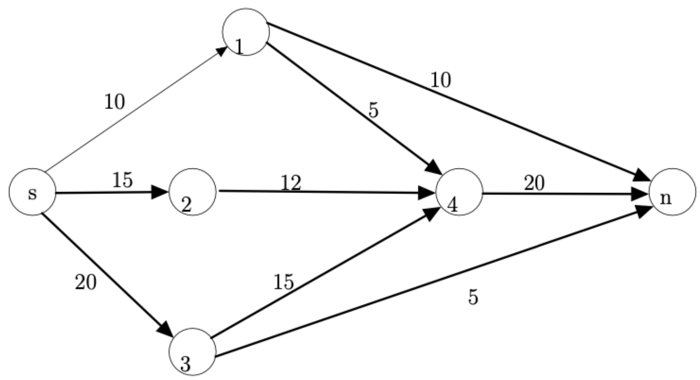
S_8

	1	2	3	4	5	6	7	8
1	–	2	3	4	4	3	8	4
2	1	–	3	4	5	5	5	6
3	1	2	–	4	2	6	6	6
4	1	2	3	–	5	6	8	8
5	1	2	3	4	–	6	8	6
6	1	2	3	4	5	–	7	8
7	1	2	3	4	5	6	–	8
8	1	2	3	4	5	6	7	–

∴ La ruta mas corta entre A y J (1 y 8) es $d_{18}=8$ y esta dada por 1 – 4 – 8
(A – D – J)

3.3. problema b) de redes

b) Considere la siguiente red dirigida donde el nodo fuente es s y el nodo destino es n y los números a través de los arcos denotan las capacidades de los flujos:



1) Encuentre el flujo máximo de s a n usando el algoritmo de Ford-Fulkenson. Los calculos hacer a mano.
2) Modele este problema como un problema de programación entera binario y use el software AMPL para resolver dicho modelo.
3) Programe un algoritmo (puede ser en Python o Matlab) para encontrar el flujo máximo de la red anterior (puede ocupar uno de los algoritmos vistos en la clase 14 o utilizar

otro), y compare su resultado con el obtenido en el inciso 1) o 2). Adjunte el script de su programa a la carpeta que entregará y guardelo como

`Algoritmoflujomaximo.m`

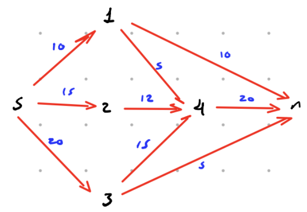
o

`Algoritmoflujomaximo.py`

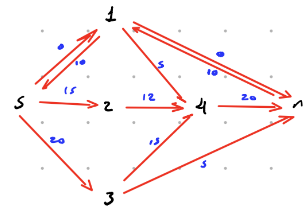
4) Determine el mínimo corte que corta el flujo de s a n y verifique que se cumple el teorema de mínimo corte y máximo flujo.

3.4. solucion b) de redes

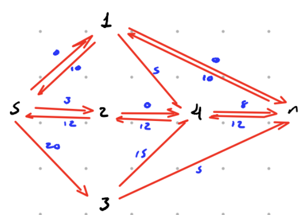
1.



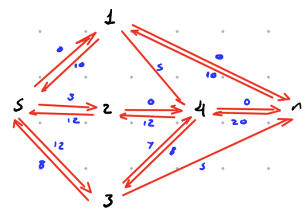
Red Residual 1: $S_1 = 10$



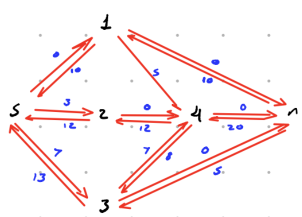
Red Residual 2: $S_2 = 12$



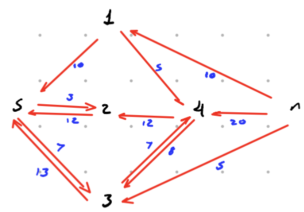
Red Residual 3: $S_3 = 8$



Red Residual 4: $S_4 = 5$



Red Conclusoria



con flujo maximo $= \sum_{i=1}^{4} S_i = 10+12+8+5 = 35$

2.

El modelado se veria de la siguiente forma:

$$
\begin{aligned}
\textbf{maximizar } Flujo &= x_{s1} + x_{s2} + x_{s3} \\
s.a. \quad x_{s1} &= x_{14} + x_{1n} \\
x_{s2} &= x_{24} \\
x_{s3} &= x_{34} + x_{3n} \\
x_{14} + x_{24} + x_{34} &= x_{4n} \\
x_{s1} &<= 10 \\
x_{s2} &<= 15 \\
x_{s3} &<= 20 \\
x_{14} &<= 5 \\
x_{1n} &<= 10 \\
x_{24} &<= 12 \\
x_{34} &<= 15 \\
x_{3n} &<= 5 \\
x_{4n} &<= 20 \\
x_{s1}, x_{s2}, x_{s3}, x_{14}, x_{1n}, x_{24}, x_{34}, x_{3n}, x_{4n} &\geq 0 \\
x_{s1}, x_{s2}, x_{s3}, x_{14}, x_{1n}, x_{24}, x_{34}, x_{3n}, x_{4n} &\in \mathbb{Z}
\end{aligned}
$$

se encuentra el archivo de AMPL con nombre *FlujoMaximo.mod* pero adjuntamos el codigo tambien aqui

Listing 4: Flujo Maximo

```
var xs1 >= 0 integer; var xs2 >= 0 integer; var xs3 >= 0 integer;
var x14 >= 0 integer; var x1n >= 0 integer;
var x24 >= 0 integer;
var x34 >= 0 integer; var x3n >= 0 integer;
var x4n >= 0 integer;

maximize Flujo:
xs1 + xs2 + xs3;

# restricciones de conservacion
subject to Nodo1: xs1 = x14 + x1n;
subject to Nodo2: xs2 = x24;
subject to Nodo3: xs3 = x34 + x3n;
subject to Nodo4: x14 + x24 + x34 = x4n;

# restricciones de capacidad
subject to Neqc0: xs1 <= 10;
subject to Neqc1: xs2 <= 15;
subject to Neqc2: xs3 <= 20;
subject to Neqc3: x14 <= 5;
subject to Neqc4: x1n <= 10;
subject to Neqc5: x24 <= 12;
subject to Neqc6: x34 <= 15;
subject to Neqc7: x3n <= 5;
subject to Neqc8: x4n <= 20;

#model FlujoMax.mod;
option solver gurobi;
solve;
```

```
display Flujo, xs1, xs2, xs3, x14, x1n, x24, x34, x3n, x4n;
```

```
AMPL
ampl: model '/Users/javier/
Gurobi 10.0.0:
0 simplex iterations
Flujo = 35
xs1 = 10
xs2 = 5
xs3 = 20
x14 = 0
x1n = 10
x24 = 5
x34 = 15
x3n = 5
x4n = 20
```

vemos en la imagen como tambien nos da 35 con este metodo

3.

El algoritmo fue realizado en python y el archivo con nombre *Algoritmoflujomaximo.py* esta incluido en el zip del proyecto pero tambien se incluye la transcripcion del codigo a continuación:

Listing 5: Flujo Maximo

```python
import collections

def ford_fulkerson(graph, source, sink):
    # Create residual graph with initial capacities
    residual_graph = collections.defaultdict(dict)
    for (u, v), capacity in graph.items():
        residual_graph[u][v] = capacity
        residual_graph[v][u] = 0

    max_flow = 0
```

```
    while True:
        # Run BFS to find augmenting path
        parent = {}
        queue = collections.deque()
        queue.append(source)
        parent[source] = None
        while queue:
            u = queue.popleft()
            if u == sink:
                break
            for v in residual_graph[u]:
                if v not in parent and residual_graph[u][v] > 0:
                    parent[v] = u
                    queue.append(v)

        # If no augmenting path found, return max flow
        if sink not in parent:
            return max_flow

        # Determine bottleneck capacity of the augmenting path
        path_capacity = float("inf")
        v = sink
        while v != source:
            u = parent[v]
            path_capacity = min(path_capacity, residual_graph[u][v])
            v = u

        # Update residual graph capacities along the augmenting path
        v = sink
        while v != source:
            u = parent[v]
            residual_graph[u][v] -= path_capacity
            residual_graph[v][u] += path_capacity
            v = u

        # Add flow through the augmenting path to max flow
        max_flow += path_capacity

# Proyecto Ford-Fulkenson Redes b)

graph = {
    ('s', 1): 10,
    ('s', 2): 15,
    ('s', 3): 20,
    (1, 4): 5,
    (1, 'n'): 10,
    (2, 4): 12,
    (3, 4): 15,
    (3, 'n'): 5,
    (4, 'n'): 20,
}

max_flow = ford_fulkerson(graph, 's', 'n')
print("Maximum flow:", max_flow)
```

```
max_flow = ford_fulkerson(graph, 's', 'n')
print("Maximum flow:", max_flow)
```

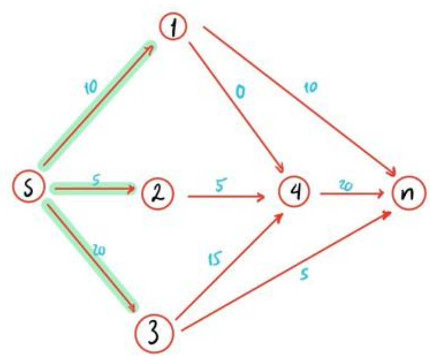
```
Maximum flow: 35
```

Vemos que el algoritmo computacional nos da el mismo resultado que los incisos 1) y 2)

4.

[Teorema de mínimo corte y máximo flujo]

Sea $G = (V, E)$ una red dirigida con una capacidad no negativa c_e para cada arco $e \in E$. Sea f una función de flujo que satisface $0 \leq f_e \leq c_e$ para todo arco $e \in E$. Sea s el vértice origen y n el vértice destino en G. Sea S un conjunto de vértices tal que $s \in S$ y $n \notin S$. Entonces, el valor del flujo máximo $|f|$ en G es igual al valor mínimo del corte $C = (S, V \setminus S)$ que separa s de t en G.



Flujo máximo = 35

$\Rightarrow$ Corte Mínimo ha de ser 35

y sea C el conjunto de cortes mínimos en la red, entonces

Dada la red final notamos que $C = \{s1, s2, s3\}$ tiene un peso de corte mínimo $\therefore$

$CM = 10 + 5 + 20 = 35 =$ flujo máximo

Cumpliendo así con el teorema antes descrito