

Laboratorio 1

Optimización I (525351)

Brayan Sandoval León

Problema 1. Encontrar en el vector costrelativo el máximo valor para elegir cual variable entra a la base.

Solución 1. El siguiente código muestra la solución de esta parte del problema

```
1 costrelativo = z - cN';  
2 cr = max(costrelativo);  
3 cpvt = find(costrelativo == cr,1,'last');
```

En este problema se busca encontrar $\max_{1 \leq j \leq n} \{z_j - c_j\}$ lo cual para este caso seria encontrar el $\max(\text{costrelativo})$.

- **Línea 2.** En esta parte buscamos el elemento mas grande del vector `costrelativo`
- **Línea 3.** Se define la variable `cpvt` nos dice la posición del ultimo elemento mas grande del vector `costrelativo`, esto se puede realizar gracias a la función `find()`. Así se obtiene la columna del pivote.

Problema 2. Encontrar en la columna correspondiente a esa variable, el valor mínimo que permite elegir que variable sale de la base.

Solución 2. El siguiente código muestra la solución de esta parte del problema

```
1 h = [];  
2 for i = 1:length(BN(:,1))  
3     if BN(i,cpvt) == 0 | BN(i,cpvt) < 0  
4         h(i) = inf;  
5     else  
6         h(i) = xb(i)/(BN(i,cpvt));  
7     end  
8 end  
9 fp = min(h);  
10 fpvt = find(h == fp,1,'last');
```

Como en el problema 1 pudimos encontrar la columna del pivote ahora solo nos queda buscar la fila en donde esta el pivote, cual se obtiene de calcular $\min_{1 \leq i \leq m} \left\{ \frac{x_i}{\alpha_{ij_0}} : \alpha_{ij_0} > 0 \right\}$

- **Línea 1.** Se crea un vector columna en donde se encuentra los elementos que pertenecen a la columna del pivote.
- **Línea 2 hasta 8.** Se crea un ciclo `for` donde se rellena un vector `h` en el cual cada elemento corresponde a $\frac{x_i}{\alpha_{ij_0}}$, y donde además se cumple que $\alpha_{ij_0} > 0$, si esto no se cumple se reemplaza por un `inf`.
- **Línea 9.** Se crea la variable `fp` la cual alberga al elemento mas pequeño de `h`.

- **Línea 10.** Nuevamente utilizamos la función `find()` pero en este caso encontrar la posición donde se encuentra el elemento mas pequeño de `h`.

Problema 3. Usar los arreglos `indexN`, `indexB` y cambiar elementos entre ellos para así cambiar en un nuevo paso (reutilizando las líneas que van desde la 14 a la 20) una nueva matriz `N` y `B`.

Solución 3. El siguiente código muestra la solución de esta parte del problema

```

1 Auxindex = indexN;
2 indexN(cpvt) = indexB(fpvt);
3 indexB(fpvt) = Auxindex(cpvt);
4 B = [];
5 cN = [];
6 cB = [];
7 for i = 1:length(indexN)
8     N(:,i) = A(:,indexN(i));
9     cN = [ cN; c(indexN(i))];
10 end
11 for i = 1:length(indexB)
12     B(:,i) = A(:,indexB(i));
13     cB = [cB ; c(indexB(i))];
14 end

```

- **Línea 1 hasta 3.** Se crea primero una variable auxiliar para poder hacer los cambios de posición entre dos vectores, en este caso se cambia la posición del vector `indexB` con `indexN` con respecto a la posición donde se encuentra el pivote.
- **Línea 7 hasta 14.** Se reutiliza el código entregado en el laboratorio.

Problema 4. Arrojar si la región factible es o no acotada o si no existe solución.

Solución 4. El siguiente código muestra la solución de esta parte del problema

```

1 for l = 1:length(costorelativo)
2     if (costorelativo(l) < 0 | costorelativo(l) == 0)
3         crt(l) = 1;
4     else
5         crt(l) = 0;
6     end
7 end
8
9 if sum(crt) == length(costorelativo)
10     Sol = 'La solucion es optima';
11 else
12     Sol = 'La solucion no es acotada';
13 end

```

- **Línea 1 hasta 7.** Se crea un ciclo `for` en el cual se rellena un vector en el cual se pone un 1 si el elemento i -ésimo es menor o igual a cero, caso contrario se pone un 0. Esto con el fin de poder determinar si $z_j - c_j \leq 0 \forall j \in [1, \dots, n]$, o si $j \in [1, \dots, n] : z_j - c_j > 0$.
- **Línea 9 hasta 13.** Se crea un `if-else` en donde se pide que la suma de todos los elementos de `crt` sean iguales a el largo del vector `costorelativo`, si se cumple esto se tendrá que la solución es óptima, caso contrario la solución es acotada.

Problema 5. Finalmente y no menos importante, verificar en cada paso que se cumpla el criterio de detención, y en caso de no cumplirse, repetir el algoritmo.

Solución 5. El siguiente código muestra la solución de esta parte del problema

```

1 while max(costorelativo)>=0
2     xb=B\b;
3     xn = zeros(length(N(:,1)),1);
4     BN = inv(B)*N;
5     z = (cB')*BN;
6     costorelativo = z - cN';
7
8     cr = max(costorelativo);
9     cpvt = find(costorelativo == cr,1,'last');
10    h = [];
11    for i = 1:length(BN(:,1))
12        if BN(i,cpvt) == 0 | BN(i,cpvt) < 0
13            h(i) = inf;
14        else
15            h(i) = xb(i)/(BN(i,cpvt));
16        end
17    end
18    fp = min(h);
19    fpvt = find(h == fp,1,'last');
20    Auxindex = indexN;
21    indexN(cpvt) = indexB(fpvt);
22    indexB(fpvt) = Auxindex(cpvt);
23    B = [];
24    cN = [];
25    cB = [];
26    for i = 1:length(indexN)
27        N(:,i) = A(:,indexN(i));
28        cN = [ cN;c(indexN(i))];
29    end
30    for i = 1:length(indexB)
31        B(:,i) = A(:,indexB(i));
32        cB = [cB ;c( indexB(i))];
33    end
34    BN = inv(B)*N;
35    A = inv(B)*A;
36    b = inv(B)*b;
37    z = cB'*BN;
38    costorelativo = z - cN';
39 end

```

- **Línea 1 hasta 39.** Se crea un ciclo `while` en el cual se buscan las nuevas matrices y vectores `B,N,BN`, `costorelativo` y `b`, también se obtiene la columna y la fila del pivote de cada iteración. este ciclo `while` termina cuando el elemento mas grande del vector `costorelativo` es menor a cero, lo cual implicaría que todos los elementos del vector son negativos y por tanto se cumpliría con el criterio de detención.

Problema 6. Se tiene el siguiente problema de minimización,

$$\begin{aligned} \text{mín} \quad & -x_1 - x_2 \\ \text{s.a.} \quad & 2x_1 + x_2 \leq 29 \\ & x_1 + 2x_2 \leq 25 \\ & 3x_1 + 4x_2 \leq 24 \\ & x_1, x_2 \geq 0 \end{aligned}$$

Solución 6. Utilizando el código realizado en el laboratorio, se obtiene,

```

1 A = [2 1;1 2;3 4];
2 Aux = A;
3 b = [29; 25; 34];
4 c = [-1;-1];
5 I = eye(length(A(:,1)));
6 A = [A I];
7 c = [c;zeros(length(A(:,1)),1)];
8 index = 1:length(c);
9 indexN = index(1:(length(Aux(1,:))));
10 indexB = index((length(indexN)+1):end);
11 B = [];
12 N = [];
13 cN = [];
14 cB = [];
15 for i=1: length( indexN )
16     N = [N A(:, indexN(i))];
17     cN = [ cN;c(indexN(i))];
18 end
19 for i=1: length( indexB )
20     B = [B A(:, indexB(i))];
21     cB = [cB ;c( indexB(i))];
22 end
23 xb=B\b;
24 xn = zeros(length(N(:,1)),1);
25 BN = inv(B)*N;
26 z = (cB')*BN;
27 costorelativo = z - cN';
28
29 for l = 1:length(costorelativo)
30     if (costorelativo(l) < 0 | costorelativo(l) == 0)
31         crt(l) = 1;
32     else
33         crt(l) = 0;
34     end
35 end
36
37 if sum(crt) == length(costorelativo)
38     Sol = 'La solucion es optima';
39 else
40     Sol = 'La solucion no es acotada';
41 end
42
43 while max(costorelativo)>=0
44     xb=B\b;
45     xn = zeros(length(N(:,1)),1);
46     BN = inv(B)*N;
47     z = (cB')*BN;
48     costorelativo = z - cN';
49
50     cr = max(costorelativo);

```

```

51 cpvt = find(costorelativo == cr,1,'last');
52 h = [];
53 for i = 1:length(BN(:,1))
54     if BN(i,cpvt) == 0 | BN(i,cpvt) < 0
55         h(i) = inf;
56     else
57         h(i) = xb(i)/(BN(i,cpvt));
58     end
59 end
60 fp = min(h);
61 fpvt = find(h == fp,1,'last');
62 Auxindex = indexN;
63 indexN(cpvt) = indexB(fpvt);
64 indexB(fpvt) = Auxindex(cpvt);
65 B = [];
66 cN = [];
67 cB = [];
68 for i = 1:length(indexN)
69     N(:,i) = A(:,indexN(i));
70     cN = [cN; c(indexN(i))];
71 end
72 for i = 1:length(indexB)
73     B(:,i) = A(:,indexB(i));
74     cB = [cB; c(indexB(i))];
75 end
76 BN = inv(B)*N;
77 A = inv(B)*A;
78 b = inv(B)*b;
79 z = cB'*BN;
80 costorelativo = z - cN';
81 end
82 Z = b'*cB;

```

Donde la solución que nos arroja es $\max z = -11,333$ con las variables $x_1 = 0, x_2 = -1$, donde x_2 es básica y x_1 es no básica

Problema 7. Se tiene el siguiente problema de minimización,

$$\begin{aligned}
 \text{mín} \quad & -2x_1 + 3x_2 - x_3 \\
 \text{s.a.} \quad & x_1 + 3x_2 + x_3 \leq 12 \\
 & -x_1 + x_2 + x_3 \leq 6 \\
 & -x_1 + 3x_2 \leq 9 \\
 & x_1, x_2, x_3 \geq 0
 \end{aligned}$$

Solución 7. Utilizando el código realizado en el laboratorio, se obtiene,

```

1 A = [1 3 1; -1 1 2; -1 3 0];
2 Aux = A;
3 b = [12; 6; 9];
4 c = [-2; 3; -1];
5 I = eye(length(A(:,1)));
6 A = [A I];
7 c = [c; zeros(length(A(:,1))-1,1)];
8 index = 1:length(c);
9 indexN = index(1:(length(Aux(1,:)))));
10 indexB = index((length(indexN)+1):end);
11 B = [];
12 N = [];
13 cN = [];

```

```

14 cB = [];
15 for i=1: length( indexN )
16     N = [N A(:, indexN(i))];
17     cN =[ cN;c(indexN(i))];
18 end
19 for i=1: length( indexB )
20     B =[B A(:, indexB(i))];
21     cB =[cB ;c( indexB(i))];
22 end
23 xb=B\b;
24 xn = zeros(length(N(:,1)) ,1);
25 BN = inv(B)*N;
26 z = (cB')*BN;
27 costorelativo = z - cN';
28 while max(costorelativo)>=0
29     xb=B\b;
30     xn = zeros(length(N(:,1)) ,1);
31     BN = inv(B)*N;
32     z = (cB')*BN;
33     costorelativo = z - cN';
34
35     cr = max(costorelativo);
36     cpvt = find(costorelativo == cr,1,'last');
37     h = [];
38     for i = 1:length(BN(:,1))
39         if BN(i,cpvt) == 0 | BN(i,cpvt) < 0
40             h(i) = inf;
41         else
42             h(i) = xb(i)/(BN(i,cpvt));
43         end
44     end
45     fp = min(h);
46     fpvt = find(h == fp,1,'last');
47     Auxindex = indexN;
48     indexN(cpvt) = indexB(fpvt);
49     indexB(fpvt) = Auxindex(cpvt);
50     B = [];
51     cN = [];
52     cB = [];
53     for i = 1:length(indexN)
54         N(:,i) = A(:,indexN(i));
55         cN =[ cN;c(indexN(i))];
56     end
57     for i = 1:length(indexB)
58         B(:,i) = A(:,indexB(i));
59         cB =[cB ;c( indexB(i))];
60     end
61     BN = inv(B)*N;
62     A = inv(B)*A;
63     b = inv(B)*b;
64     z = cB'*BN;
65     costorelativo = z - cN';
66 end
67 Z = b'*cB;

```

Donde la solución que nos arroja es $\max z = -24$ con las variables $x_1 = -2, x_2 = 0, x_3 = 0$ donde x_1 es básica y x_2 con x_3 son no básicas.

Problema 8. Se tiene el siguiente problema de minimización,

$$\begin{aligned} \text{mín} \quad & -2x_1 - 3x_2 - 5x_3 - 4x_4 \\ \text{s.a.} \quad & x_1 + x_2 + 2x_3 \leq 4 \\ & x_2 + x_3 + x_4 \leq 3 \\ & 2x_1 + x_2 \leq 4 \\ & x_1 + x_2 \leq 2 \\ & x_3 + x_4 \leq 2 \\ & 3x_3 + 2x_4 \leq 5 \\ & x_1, x_2, x_3, x_4 \geq 0 \end{aligned}$$

Solución 8. Utilizando el código realizado en el laboratorio, se obtiene,

```

1 A = [1 1 2 0;0 1 1 1;2 1 0 0;1 1 0 0;0 0 1 1; 0 0 3 2];
2 Aux = A;
3 b = [4;3;4;2;2;5];
4 c = [-2;-3;-5;-4];
5 I = eye(length(A(:,1)));
6 A = [A I];
7 c = [c;zeros(length(A(:,1)),1)];
8 index = 1:length(c);
9 indexN = index(1:(length(Aux(1,:))));
10 indexB = index((length(indexN)+1):end);
11 B = [];
12 N = [];
13 cN = [];
14 cB = [];
15 for i=1: length( indexN )
16     N = [N A(:, indexN(i))];
17     cN = [ cN;c(indexN(i))];
18 end
19 for i=1: length( indexB )
20     B = [B A(:, indexB(i))];
21     cB = [cB ;c( indexB(i))];
22 end
23 xb=B\b;
24 xn = zeros(length(N(:,1)),1);
25 BN = inv(B)*N;
26 z = (cB')*BN;
27 costorelativo = z - cN';
28
29 for l = 1:length(costorelativo)
30     if (costorelativo(l) < 0 | costorelativo(l) == 0)
31         crt(l) = 1;
32     else
33         crt(l) = 0;
34     end
35 end
36
37 if sum(crt) == length(costorelativo)
38     Sol = 'La solucion es optima';
39 else
40     Sol = 'La solucion no es acotada';
41 end
42
43 while max(costorelativo)>=0
44     xb=B\b;
45     xn = zeros(length(N(:,1)),1);

```

```

46 BN = inv(B)*N;
47 z = (cB')*BN;
48 costorelativo = z - cN';
49
50 cr = max(costorelativo);
51 cpvt = find(costorelativo == cr,1,'last');
52 h = [];
53 for i = 1:length(BN(:,1))
54     if BN(i,cpvt) == 0 | BN(i,cpvt) < 0
55         h(i) = inf;
56     else
57         h(i) = xb(i)/(BN(i,cpvt));
58     end
59 end
60 fp = min(h);
61 fpvt = find(h == fp,1,'last');
62 Auxindex = indexN;
63 indexN(cpvt) = indexB(fpvt);
64 indexB(fpvt) = Auxindex(cpvt);
65 B = [];
66 cN = [];
67 cB = [];
68 for i = 1:length(indexN)
69     N(:,i) = A(:,indexN(i));
70     cN = [ cN; c(indexN(i))];
71 end
72 for i = 1:length(indexB)
73     B(:,i) = A(:,indexB(i));
74     cB = [cB ; c( indexB(i))];
75 end
76 BN = inv(B)*N;
77 A = inv(B)*A;
78 b = inv(B)*b;
79 z = cB'*BN;
80 costorelativo = z - cN';
81 end
82 Z = b'*cB;

```

Donde la solución que nos arroja es $\max z = -15$ con las variables $x_1 = 1,5, x_2 = 1, x_3 = 1, x_4 = 1$, donde todas las variables son básicas.