Exercice 1 - Simplex implementation

Gleisson de Assis

April 17, 2018

1 Introduction

This work solves the following problem:

$$
\begin{array}{ll}
\text{minimize} & z = 2x_1 - 10x_2 + x_3 + 4x_4 \\
\text{subject to} & 3x_1 + 6x_2 3x_4 \leq 100 \\
& 10_4 + x_2 + 6x_3 \geq 50 \\
& -3x_1 + x_2 + 6x_3 \geq 30 \\
& x \geq 0
\end{array}
$$

1.1 How it works

This problem is solved by running simplex2 implementation without specifying a feasible base. The function simplex2 calls simplex1 in order to find a feasible base, after that using the found base the function is called again solving problem.

1.2 Results

The optimum result for this problem is:

$$
\begin{array}{l}
z = -119.0278 \\
x = [0, 14.1667, 2.6389, 5.0000]
\end{array}
$$

1.3 Input parameters

The simplxe2 implementation was configured as follows:

```
1  %test problem
2  %https://paginas.fe.up.pt/~mac/ensino/docs/MD20082009/
       MIEA_MD_TPC3TiagoPinto.pdf
3
4  c = [2 -10 1 4 0 0 0];
5
6  A = [ 3  6  0  3  1  0  0
7        4  0  0  10 0 -1  0
8       -3  1  6  0  0  0 -1];
```

```
 9
10  b = [100
11        50
12        30];
13
14  [x, z, Ir, it, tipo] = simplex2(A, b, c)
```

2 Source Code

simplex1.m

```
 1  function [x, z, I, it, tipo] = simplex1(A, b, c, I)
 2        [~,n] = size(A);
 3
 4        J = 1:n;
 5        J(I) = [];
 6
 7        it = 0;
 8
 9        while true
10            it = it + 1;
11
12            Ai = A(:, I) ^ -1;
13            Aj = A(:, J);
14            ci = c(:, I);
15            cj = c(:, J);
16
17            pi = ci * Ai;
18            ccj = pi * Aj - cj;
19
20            z = pi * b;
21
22            x = n:1;
23            x(I) = Ai * b;
24            x(J) = 0;
25            x = x.';
26
27            [t,k] = max(ccj);
28
29            if ccj == 0
30                tipo = 1;
31                break;
32            elseif t <= 0
33                tipo = 0;
```

```
34                    break ;
35              else
36                    xi =   Ai ∗ b;
37                    Ak = Ai ∗ A( : ,  J(k) ) ;
38
39                    byk = xi ./Ak;
40
41                    byk( byk <= 0) = inf ;
42                    [ t , r ] = min( byk ) ;
43
44                    if  t == inf
45                         tipo = −1;
46                         break ;
47                    end
48
49                    [ J(k) ,  I( r ) ] = deal ( I( r ) ,  J(k) ) ;
50              end
51        end
52  end
```

simplex2.m

```
1  function  [ x ,  z ,  I ,  it ,  tipo ] = simplex2 (A,  b ,  c )
2       [m, n] =  size (A) ;
3       I = 1:n+m;
4       ct = zeros (1 ,  n+m) ;
5
6       I (1:n) = [ ] ;
7       ct ( I ) = 1;
8
9       B = horzcat (A,  eye (m) ) ;
10
11       [ x ,  z_f1 ,  I_f1 ,  it_f1 ,  ˜] = simplex1 (B,  b ,  ct ,  I ) ;
12
13       if  z_f1 == 0
14            [ x ,  z ,  I ,  it_f2 ,  tipo ] = simplex1 (A,  b ,  c ,  I_f1 ) ;
15
16            it = it_f2 + it_f1 ;
17       end
18  end
```