Trabalho 2 SEP

Aluno: Victor Daniel Oliveira Gaete

Implementação em MATLAB, e aplicando ao sistema IEEE14 barras e IEEE33 barras.

IEEE 14 Barras

```
% [       NUM    TIPO    V        TETA     PG    QG      PD         QD       GSH      BSH ]
barras = [
            1   2   1.060 0.0  232.4 -16.9  0.0     0.0     0.0    0.0 0 0
            2   1   1.045 0.0  40.0  42.4  21.70  12.70    0.0    0.0 -40  50
            3   1   1.010 0.0  0.0  23.4  94.20  19.00    0.0    0.0 0    40
            4   0   1.000 0.0  0.0    0.0  47.80   -3.90   0.0    0.0 0    0
            5   0   1.000 0.0  0.0    0.0  7.60   1.600    0.0    0.0 0    0
            6   1   1.070 0.0  0.0    12.2  11.20  7.500    0.0    0.0 -6   24
            7   0   1.000 0.0  0.0    0.0   0.0    0.0     0.0    0.0 0    0
            8   1   1.090 0.0  0.0    17.4   0.0    0.0     0.0    0.0 -6   24
            9   0   1.000 0.0  0.0    0.0  29.50   16.60   0.0   19.0 0    0
            10  0   1.000 0.0  0.0    0.0   9.00   5.800   0.0    0.0 0    0
            11  0   1.000 0.0  0.0    0.0   3.50   1.800   0.0    0.0 0    0
            12  0   1.000 0.0  0.0    0.0   6.10   1.600   0.0    0.0 0    0
            13  0   1.000 0.0  0.0    0.0   13.50  5.800   0.0    0.0 0    0
            14  0   1.000 0.0  0.0    0.0   14.90  5.000   0.0    0.0 0    0 ];

%            [FR     TO         R          X      BSHtotal   Tap   ]
linhas = [
1    2    0.01938  0.05917    0.05280    0
1    5    0.05403  0.22304    0.04920    0
2    3    0.04699  0.19797    0.04380    0
2    4    0.05811  0.17632    0.03400    0
2    5    0.05695  0.17388    0.03460    0
3    4    0.06701  0.17103    0.01280    0
4    5    0.01335  0.04211    0          0
4    7    0.0      0.20912    0          0.978
4    9    0.0      0.55618    0          0.969
5    6    0.0      0.25202    0          0.932
6    11    0.09498  0.19890    0          0
6    12    0.12291  0.25581    0          0
6    13    0.06615  0.13027    0          0
7    8    0.0      0.17615     0          0
7    9    0.0      0.11001    0          0
9    10    0.03181  0.08450    0            0
9    14    0.12711  0.27038    0          0
10    11    0.08205  0.19207    0          0
12    13    0.22092  0.19988    0          0
13    14    0.17093  0.34802    0          0          ];


Ref= 1;
```

```
% ========================
% Leitura dos dados de entrada

[NumBarras,~] = size(barras);
[NumLinhas,~] = size(linhas);

% Linhas
% Init das variaveis
de = zeros(NumLinhas,1);
para = zeros(NumLinhas,1);
R = zeros(NumLinhas,1);
X = zeros(NumLinhas,1);
B = zeros(NumLinhas,1);
G = zeros(NumLinhas,1);
Tap = zeros(NumLinhas,1);

for i=1:NumLinhas
    de(i) = linhas(i,1);
    para(i) = linhas(i,2);
    R(i) = linhas(i,3);
    X(i) = linhas(i,4);
    B(i) = 1/X(i);
    G(i) = 1/(R(i) + 0.0001);
    Tap(i) = linhas(i,6);
end

% Desconsiderando taps
Tap = ones(NumLinhas,1);

% Barras
% Init das variaveis
Tipo = zeros(NumBarras,1);
V = zeros(NumBarras,1);
Teta = zeros(NumBarras,1);
Pg = zeros(NumBarras,1);
Qg = zeros(NumBarras,1);
Pd = zeros(NumBarras,1);
Qd = zeros(NumBarras,1);
Gsh = zeros(NumBarras,1);
Bsh = zeros(NumBarras,1);

for i=1:NumBarras
    Tipo(i) = barras(i,2);
    V(i) = barras(i,3);
    Teta(i) = barras(i,4);
    Pg(i) = barras(i,5);
    Qg(i) = barras(i,6);
    Pd(i) = barras(i,7);
    Qd(i) = barras(i,8);
    Gsh(i) = barras(i,9);
    Bsh(i) = barras(i,10);
end

% Potencias em PU
Pg = Pg/100;
Qg = Qg/100;
Pd = Pd/100;
Qd = Qd/100;



% ========================
% Matriz B_linha

B_linha = zeros(NumBarras,NumBarras);

for i=1:NumLinhas
    K = de(i);
    M = para(i);
    B_linha(K,K) = B_linha(K,K) + B(i)/(Tap(i)^2); % diagonal principal considerando tap
    B_linha(M,M) = B_linha(M,M) + B(i); % diagonal principal
    B_linha(K,M) = B_linha(K,M) - B(i)/Tap(i); % Fora da diagonal principal
    B_linha(M,K) = B_linha(M,K) - B(i)/Tap(i); % Fora da diagonal principal
end

% ========================
% Fluxo de potencia linearizado


B_REF = Ref;
B_linha(B_REF,B_REF) = 10^20; % Infinito
Teta=  B_linha\(-Pd+Pg); % inv
```

```
Warning: Matrix is close to singular or badly scaled. Results may be inaccurate. RCOND =  4.588513e-21.
```

```

% Perdas
erro_min = 1e-6;
erro = 1;
perdas = zeros(NumBarras,1);
Pperdas = zeros(NumLinhas,1);
Iteracoes = 0;

while erro > erro_min
    Teta_antigo = Teta;
    % Calculo de Pperdas = Gkm*(Teta(K)-Teta(M))^2
    for i=1:NumLinhas

        K = de(i);
        M = para(i);
        Pperdas(i) = G(i)*(Teta(K)-Teta(M))^2/100;
        perdas(K) = Pperdas(i)/2;
        perdas(M) = Pperdas(i)/2;
    end
    % Calculo de Teta
    Teta=  B_linha\(-Pd+Pg-perdas); % inv
    % Erro
    erro = max(abs(Teta-Teta_antigo));
    Iteracoes= Iteracoes+1;
end
```

```
Warning: Matrix is close to singular or badly scaled. Results may be inaccurate. RCOND =  4.588513e-21.
Warning: Matrix is close to singular or badly scaled. Results may be inaccurate. RCOND =  4.588513e-21.
Warning: Matrix is close to singular or badly scaled. Results may be inaccurate. RCOND =  4.588513e-21.
Warning: Matrix is close to singular or badly scaled. Results may be inaccurate. RCOND =  4.588513e-21.
Warning: Matrix is close to singular or badly scaled. Results may be inaccurate. RCOND =  4.588513e-21.
Warning: Matrix is close to singular or badly scaled. Results may be inaccurate. RCOND =  4.588513e-21.
```

```

Pkm = zeros(NumLinhas,1);

for i=1:NumLinhas
    K = de(i);
    M = para(i);
    Pkm(i) = (Teta(K)-Teta(M))*B(i)/Tap(i); % Pkm = Pk - Pm
end


% Variaveis de saida
disp('Fluxo de potencia linearizado - IEEE 14 Barras -  Pd = 100%')
```

```
Fluxo de potencia linearizado - IEEE 14 Barras -  Pd = 100%
```

```
disp('Theta: ')
```

```
Theta:
```

```
disp(Teta)
```

```
   -0.0000
   -0.1290
   -0.2949
   -0.2771
   -0.2480
   -0.3570
   -0.3432
   -0.3432
   -0.3731
   -0.3778
   -0.3710
   -0.3766
   -0.3797
   -0.3987
```

```
disp('Pkm: ')
```

```
Pkm:
```

```
disp(Pkm)
```

```
    2.1802
    1.1117
    0.8380
    0.8399
    0.6841
   -0.1041
   -0.6921
    0.3163
    0.1727
    0.4328
    0.0702
    0.0765
    0.1740
   -0.0000
    0.2716
    0.0548
    0.0945
   -0.0352
    0.0155
    0.0545
```

```
disp('Número de Iterações')
```

```
Número de Iterações
```

```
disp(Iteracoes)
```

```
     6
```

IEEE 33 Barras

```
% Dados:


% [        NUM     TIPO    V         TETA      PG     QG       PD         QD        GSH      BSH ]
barras = [
             1      0      1.0000  0.0000    0      0       100.0       60.0      0.0      0.0;
             2      0      1.0000  0.0000    0      0        90.0       40.0      0.0      0.0;
             3      0      1.0000  0.0000    0      0       120.0       80.0      0.0      0.0;
             4      0      1.0000  0.0000    0      0        60.0       30.0      0.0      0.0;
             5      0      1.0000  0.0000    0      0        60.0       20.0      0.0      0.0;
             6      0      1.0000  0.0000    0      0       200.0      100.0      0.0      0.0;
             7      0      1.0000  0.0000    0      0       200.0      100.0      0.0      0.0;
             8      0      1.0000  0.0000    0      0        60.0       20.0      0.0      0.0;
             9      0      1.0000  0.0000    0      0        60.0       20.0      0.0      0.0;
            10      0      1.0000  0.0000    0      0        45.0       30.0      0.0      0.0;
            11      0      1.0000  0.0000    0      0        60.0       35.0      0.0      0.0;
            12      0      1.0000  0.0000    0      0        60.0       35.0      0.0      0.0;
            13      0      1.0000  0.0000    0      0       120.0       80.0      0.0      0.0;
```

```
           14       0     1.0000  0.0000    0      0       60.0       10.0     0.0    0.0;
           15       0     1.0000  0.0000    0      0       60.0       20.0     0.0    0.0;
           16       0     1.0000  0.0000    0      0       60.0       20.0     0.0    0.0;
           17       0     1.0000  0.0000    0      0       90.0       40.0     0.0    0.0;
           18       0     1.0000  0.0000    0      0       90.0       40.0     0.0    0.0;
           19       0     1.0000  0.0000    0      0       90.0       40.0     0.0    0.0;
           20       0     1.0000  0.0000    0      0       90.0       40.0     0.0    0.0;
           21       0     1.0000  0.0000    0      0       90.0       40.0     0.0    0.0;
           22       0     1.0500  0.0000    0      0       90.0       50.0     0.0    0.0;
           23       0     1.0000  0.0000    0      0      420.0      200.0     0.0    0.0;
           24       0     1.0000  0.0000    0      0      420.0      200.0     0.0    0.0;
           25       0     1.0500  0.0000    0      0       60.0       25.0     0.0    0.0;
           26       0     1.0500  0.0000    0      0       60.0       25.0     0.0    0.0;
           27       0     1.0000  0.0000    0      0       60.0       20.0     0.0    0.0;
           28       0     1.0000  0.0000    0      0      120.0       70.0     0.0    0.0;
           29       0     1.0000  0.0000    0      0      200.0      600.0     0.0    0.0;
           30       0     1.0000  0.0000    0      0      150.0       70.0     0.0    0.0;
           31       0     1.0000  0.0000    0      0      210.0      100.0     0.0    0.0;
           32       0     1.0000  0.0000    0      0       60.0       40.0     0.0    0.0;
           33       2     1.0000  0.0000    0      0        0.0        0.0     0.0    0.0
];



%            [FR    TO         R          X      BSHtotal   Tap  Tapmin  Tapmax]
linhas    = [ 33      1      0.0922     0.0470       0        0     0       0;
               1      2      0.4930     0.2511       0        0     0       0;
               2      3      0.3660     0.1864       0        0     0       0;
               3      4      0.3811     0.1941       0        0     0       0;
               4      5      0.8190     0.7070       0        0     0       0;
               5      6      0.1872     0.6188       0        0     0       0;
               6      7      0.7114     0.2351       0        0     0       0;
               7      8      1.0300     0.7400       0        0     0       0;
               8      9      1.0440     0.7400       0        0     0       0;
               9      10     0.1966     0.0650       0        0     0       0;
              10      11     0.3744     0.1238       0        0     0       0;
              11      12     1.4680     1.1550       0        0     0       0;
              12      13     0.5416     0.7129       0        0     0       0;
              13      14     0.5910     0.5260       0        0     0       0;
              14      15     0.7463     0.5450       0        0     0       0;
              15      16     1.2890     1.7210       0        0     0       0;
              16      17     0.7320     0.5740       0        0     0       0;
               1      18     0.1640     0.1565       0        0     0       0;
              18      19     1.5042     1.3554       0        0     0       0;
              19      20     0.4095     0.4784       0        0     0       0;
              20      21     0.7089     0.9373       0        0     0       0;
               2      22     0.4512     0.3083       0        0     0       0;
              22      23     0.8980     0.7091       0        0     0       0;
              23      24     0.8960     0.7011       0        0     0       0;
               5      25     0.2030     0.1034       0        0     0       0;
               25      26      0.2842      0.1447       0        0      0       0;
               26      27      1.0590      0.9337       0        0      0       0;
               27      28      0.8042      0.7006       0        0      0       0;
               28      29      0.5075      0.2585       0        0      0       0;
               29      30      0.9744      0.9630       0        0      0       0;
               30      31      0.3105      0.3619       0        0      0       0;
               31      32      0.3410      0.5302       0        0      0       0;
              ];

Sb = 1e6;      % Potencia Base
Vb = 12.66e3;  % Tensão Base
Zb = Vb^2/Sb;
linhas(:,3:4) = linhas(:,3:4)/Zb;

barras(:,7:8) = barras(:,7:8)*1e3/Sb;

ref = 1;
% ========================
% Leitura dos dados de entrada

[~,~] = size(barras);

[~,~] = size(linhas);

[NumBarras,NumBCol] = size(barras);
[NumLinhas,NumLCol] = size(linhas);

% Linhas
% Init das variaveis
de = zeros(NumLinhas,1);
para = zeros(NumLinhas,1);
R = zeros(NumLinhas,1);
X = zeros(NumLinhas,1);
B = zeros(NumLinhas,1);
G = zeros(NumLinhas,1);
Tap = zeros(NumLinhas,1);

for i=1:NumLinhas
    de(i) = linhas(i,1);
    para(i) = linhas(i,2);
    R(i) = linhas(i,3);
    X(i) = linhas(i,4);
    B(i) = 1/X(i);
    G(i) = 1/(R(i) + 0.00001);
    Tap(i) = linhas(i,6);
end

% Desconsiderando taps
Tap = ones(NumLinhas,1);
```

```
% Barras
% Init das variaveis
Tipo = zeros(NumBarras,1);
V = zeros(NumBarras,1);
Teta = zeros(NumBarras,1);
Pg = zeros(NumBarras,1);
Qg = zeros(NumBarras,1);
Pd = zeros(NumBarras,1);
Qd = zeros(NumBarras,1);
Gsh = zeros(NumBarras,1);
Bsh = zeros(NumBarras,1);

for i=1:NumBarras
    Tipo(i) = barras(i,2);
    V(i) = barras(i,3);
    Teta(i) = barras(i,4);
    Pg(i) = barras(i,5);
    Qg(i) = barras(i,6);
    Pd(i) = barras(i,7);
    Qd(i) = barras(i,8);
    Gsh(i) = barras(i,9);
    Bsh(i) = barras(i,10);
end

% Potencias em PU
Pg = Pg/100;
Qg = Qg/100;
Pd = Pd/100;
Qd = Qd/100;



% ========================
% Matriz B_linha

B_linha = zeros(NumBarras,NumBarras);

for i=1:NumLinhas
    K = de(i);
    M = para(i);
    B_linha(K,K) = B_linha(K,K) + B(i)/(Tap(i)^2); % diagonal principal considerando tap
    B_linha(M,M) = B_linha(M,M) + B(i); % diagonal principal
    B_linha(K,M) = B_linha(K,M) - B(i)/Tap(i); % Fora da diagonal principal
    B_linha(M,K) = B_linha(M,K) - B(i)/Tap(i); % Fora da diagonal principal
end

% ========================
% Fluxo de potencia linearizado
```

```
B_REF = ref;
B_linha(B_REF,B_REF) = 10^20; % Infinito
Teta=  B_linha\(-Pd+Pg); % inv
```

```
Warning: Matrix is close to singular or badly scaled. Results may be inaccurate. RCOND =  2.169055e-20.
```

```

% Perdas
erro_min = 1e-6;
erro = 1;
perdas = zeros(NumBarras,1);
Pperdas = zeros(NumLinhas,1);
Iteracoes = 0;

while erro > erro_min
    Teta_antigo = Teta;
    % Calculo de Pperdas = Gkm*(Teta(K)-Teta(M))^2
    for i=1:NumLinhas

        K = de(i);
        M = para(i);
        Pperdas(i) = G(i)*(Teta(K)-Teta(M))^2/100;
        perdas(K) = Pperdas(i)/2;
        perdas(M) = Pperdas(i)/2;
    end
    % Calculo de Teta
    Teta=  B_linha\(-Pd+Pg-perdas); % inv
    % Erro
    erro = max(abs(Teta-Teta_antigo));
    Iteracoes= Iteracoes+1;
end
```

```
Warning: Matrix is close to singular or badly scaled. Results may be inaccurate. RCOND =  2.169055e-20.
```

```

Pkm = zeros(NumLinhas,1);

for i=1:NumLinhas
    K = de(i);
    M = para(i);
    Pkm(i) = (Teta(K)-Teta(M))*B(i)/Tap(i); % Pkm = Pk - Pm
end


% Variaveis de saida
disp('Fluxo de potencia linearizado - IEEE 33 Barras -  Pd = 100%')
```

```
Fluxo de potencia linearizado - IEEE 33 Barras -  Pd = 100%
```

```
disp('Theta: ')
```

```
Theta:
```

```
disp(Teta)
```

```
   1.0e-03 *

   -0.0000
   -0.0510
   -0.0770
   -0.1026
   -0.1933
   -0.2348
   -0.2476
   -0.2788
   -0.3072
   -0.3094
   -0.3133
   -0.3458
   -0.3631
   -0.3720
   -0.3791
   -0.3952
   -0.3984
   -0.0035
   -0.0263
   -0.0317
   -0.0370
   -0.0689
   -0.1060
   -0.1244
   -0.1992
   -0.2070
   -0.2536
   -0.2859
   -0.2959
   -0.3211
   -0.3272
   -0.3292
   -0.0000
```

```
disp('Pkm: ')
```

```
Pkm: 
```

```
disp(Pkm)
```

```
         0
    0.0326
    0.0224
    0.0212
    0.0206
    0.0108
    0.0088
    0.0068
    0.0062
    0.0056
    0.0051
    0.0045
    0.0039
    0.0027
    0.0021
    0.0015
    0.0009
    0.0036
    0.0027
    0.0018
    0.0009
    0.0093
    0.0084
    0.0042
    0.0092
    0.0086
    0.0080
    0.0074
    0.0062
    0.0042
    0.0027
    0.0006
```

```
disp('Número de Iterações')
```

```
Número de Iterações
```

```
disp(Iteracoes)
```

```
     1
```

Conclusão:

Foi observado que a convergência ocorreu rapidamente, principalmente para o sistema de 33 barras em que o resultado convergiu na preimeira iteração, demonstrando que as perdas são realmente pequenas. Considera-se o trabalho um sucesso, visto que a implementação foi capaz de convergir e demonstrar a perdas no sistema