SME0305 - 2016

Roberto Ausas/Gustavo Buscaglia

Lista 3 - Redes hidráulicas em Octave

Criação das redes em Octave

1. Criar um arquivo com as conectividades aresta-nó da rede hidráulica utilizada na aula introdutoria e fazer um programa de Octave que a carregue, determine o seu número de arestas `nc` e o número de nós `nv`. Em um arquivo separado incluir as conductâncias de cada cano e também carregar-lo.

2. Considerar a função fornecida

   `function [nv,nc,conec,C,coords]= RedeHidraBairro(n,m,CH,CV)`

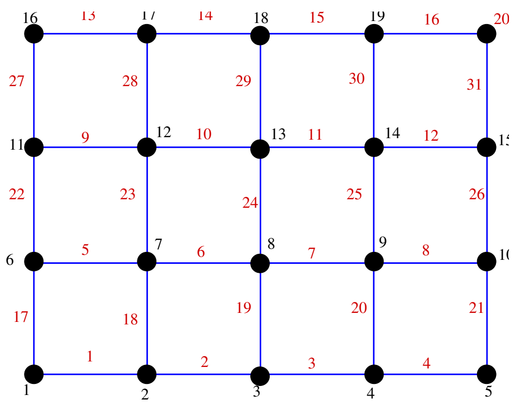
   que devolve uma rede hidráulica com `nv` (número de nós), `nc` (número de canos ou arestas), `conec` (matriz de conectividades aresta-nó da rede), `C` (vetor com os valores das conductâncias de cada aresta) e `coords` (coordenadas x-y de cada nó). As redes são do tipo mostrado na figura, sendo `n` nós na horizontal por `m` na vertical (i.e. `nv = n * m`), e sendo os valores da conductância dos canos horizontais de valor `CH` e dos verticais de valor `CV`. Criar uma rede hidráulica usando p.e. `n= 5, m = 4, CH = 2, CV = 3`.

   (**Comentário**: Você apenas precisar usar a função. Não precisa-se preocupar com os detalhes de sua implementação).

3. Suponha conhecido o vetor `C(1:nc)` de conductâncias de um circuito arbitrário. Programe uma função:

   `function Cnew = RandomFail(nc,C,a,Centup)`

   que calcule um novo vetor de valores de conductâncias `Cnew` estocástico, considerando que cada cano do circuito tem uma probabilidade `a` de se entupir. Então, cada cano `i` pode, **ou não falhar** (em cujo caso sua conductância fica como está, `Cnew(i)=C(i)`), **ou falhar** (em cujo caso seu valor passa para o valor `Centup`, constante, i.e., `Cnew(i)=Centup`).



Montagem das matrizes em Octave

4. Completar a função que faz a montagem da matriz para uma rede hidráulica:

```
function A =  Assembly(nv, nc, conec, C)
  A = zeros(  );
  for i=1:nc
    p = conec(i,    ;
    q =             ;
    A(p,p) =                  ;
                              ;
                              ;
                              ;
  end
end
```

   A continuação, implementar uma forma vetorizada, em que a matriz de 2×2 de cada cano seja somada de uma só vez na matriz global.

5. Fazer um programa de Octave para montar as matrizes das redes usadas nos exercícios 1 e 2 .

Resolução do sistema linear completo

6. Completar a função que faz as mudanças da matriz de uma rede hidráulica para eliminar a singularidade do sistema e também a mudança do vetor de lado direito.

```
function [Atilde b] = BuildSystem(nv, A,
                         QB, nB, Pref, natm)
  Atilde = A;
  b = zeros( ,1);
  Atilde(   ,:) =       ;
  Atilde(natm,) =      ;
  b(   ) =          ;
  b(   ) = ;
end
```

7. A partir dos códigos desenvolvidos nos exercícios previos, adicionar a função anterior e montar o sistema linear de equações definitivo. Em cada caso, escolher o nó `natm` que será conectado à atmosfera e o nó `nB` que será conectado à bomba. Utilizar o valor de pressão de referencia `Pref = 0` e injectar uma vazão `QB = 10` (m^3/s) por exemplo.

   Finalmente, resolver o sistema de equações para achar o vetor de pressões usando a barra de Octave/Matlab: `p = Atilde \ b`

Post-processo da solução

8. Na rede do exercício 2

   (a) Usando a função disponibilizada `PlotaRede`, plotar as pressões e vazões obtidas.

   `> PlotaRede(coords, conec, p, C, flag, xmin,xmax,ymin,ymax,'-ro')`

   (a variável `flag` se for igual a `0` imprime as pressões nos nós, e se for igual a `1` imprime as vazões nas arestas)

   (b) Plotar as linhas de pressão constante usando as instruções

   `> paux=reshape(p,n,m);`
   `> contourf(paux',15);`

   (c) Calcular a conductância equivalente da rede entre os pontos `nB` e `natm`.

(d) Calcular a vazão pelos canos da rede:

```
> vazoes = K * D * p
```

em que `K` é a matriz diagonal com as conductâncias dos canos definida por

$$\texttt{K(i,j)} = \begin{cases} \texttt{C(i)} & \texttt{se i=j} \\ \\ \texttt{0} & \texttt{no resto} \end{cases}$$

e `D` é a matriz definida por:

$$\texttt{D(k,j)} = \begin{cases} 1 & \texttt{se j = conec(k,1)} \\ \\ -1 & \texttt{se j = conec(k,2)} \\ \\ 0 & \text{no resto} \end{cases}$$

(e) Calcular a potência consumida pela bomba via cálculo das perdas na rede:

```
> potencia = p' * (D' * K * D) * p
```