Anàlisi de circuits amb Octave

Circuits resistius

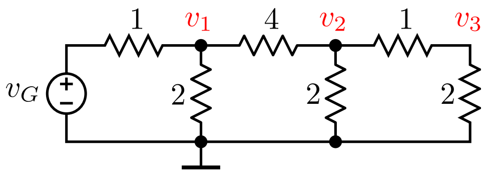
Pere Palà

Setembre de 2011

En aquest document expliquem com podem ajudar-nos amb Octave per analitzar circuits resistius. Octave *no* és un programa d'anàlisi de circuits, però es pot fer servir per als càlculs que cal dur a terme. Octave o Matlab, la versió no lliure en què s'inspira, són **les** eines matemàtiques que fan servir els enginyers electrònics i/o de telecomunicació.

1 Plantejament del problema

Considerem un circuit elemental, com el que es representa a continuació:



Tenim tres tensions nodals desconegudes, v_1, v_2 i v_3. Per tant, plantegem 3 KCLs i obtenim el sistema d'equacions

$$\begin{bmatrix} \frac{1}{1}+\frac{1}{2}+\frac{1}{4} & -\frac{1}{4} & 0 \\ -\frac{1}{4} & \frac{1}{4}+\frac{1}{2}+\frac{1}{1} & -\frac{1}{1} \\ 0 & -\frac{1}{1} & \frac{1}{1}+\frac{1}{2} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} v_g \\ 0 \\ 0 \end{bmatrix}, \tag{1}$$

que podem expressar matricialment de forma compacta com

$$\mathbf{T}\mathbf{v} = \mathbf{w} \tag{2}$$

Per resoldre aquest sistema d'equacions, podem ajudar-nos de `octave`, que és l'objectiu d'aquest document.

2 Primers passos amb Octave

En primer lloc, cridem `octave`. Està als repositoris de linux però també hi ha versió per windows.

```
xxx@yyy-u:~$ octave
GNU Octave, version 3.0.5
Copyright (C) 2008 John W. Eaton and others.
This is free software; see the source code for copying conditions.
There is ABSOLUTELY NO WARRANTY; not even for MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE.  For details, type ‘warranty’.
...
octave:1>
```

En primer lloc, creem la matriu **T**, especificant cadascun del seus elements:

```
octave:1> T=[1+1/2+1/4 -1/4 0; -1/4 1/4+1/2+1 -1; 0 -1 1+1/2]
T =
   1.75000  -0.25000   0.00000
  -0.25000   1.75000  -1.00000
   0.00000  -1.00000   1.50000
```

Observeu que les files es separen amb espais en blanc (també es podria fer amb coma “,“) i les columnes amb punt i coma “;”.

A continuació entrem el vector de dades **w** (el terme de la dreta de l'equació (1)):

```
octave:2> w=[1; 0; 0]
w =
   1
   0
   0
```

Observareu que hem posat 1 en comptes de v_g. En un sistema lineal la sortida és proporcional a l'entrada. Si coneixem la sortida per una entrada 1, la sortida per una entrada 2 serà el doble i la sortida per una entrada v_g serà v_g vegades més gran (o petita).

Ara, per resoldre el sistema d'equacions, és a dir obtenir el vector **v**, (aquí comença la màgia de l'Octave) amb una simple instrucció n'hi ha prou:

```
octave:3> v=T\w
v =
   0.590909
   0.136364
   0.090909
```

Aquesta expressió és equivalent a haver calculat explícitament la inversa de la matriu **T** i multiplicar-la pel vector **w**, com es fa matemàticament. Això s'escriuria així:

```
octave:3> v=inv(T)*w
v =
   0.590909
   0.136364
   0.090909
```

No obstant, per problemes de més dimensió és més eficaç la primera notació, que només demana resoldre el sistema d'equacions i no necessàriament calcular la matriu inversa.

Ara tenim totes les tensions nodals al vector **v**. Si volem accedir a una de les variables, per exemple la tensió de sortida al node 3, v_3:

```
octave:4> v(3)
ans =  0.090909
```

Aquest valor és la sortida per $v_g = 1$. En general, la sortida serà $v_3 = 0.90900v_g$.

El resultat ens queda a la variable *ans*, que podem aprofitar per qualsevol altra cosa. I amb això, ja podem calcular qualsevol cosa. Per exemple, el corrent que circula per la resistència de 4Ω val:

```
octave:5> i_4=(v(1)-v(2))/4
i_4 =   0.11364
```

Aquest valor, novament, és la sortida per $v_g = 1$. Si volem l'expressió general, sols cal tenir en compte la proporcionalitat amb v_g, de forma que $i_4 = 0.11364 v_g$.

Qualsevol altra variable es calcularia de forma inmediata a partir de les variables generadores calculades.