

Octave per i Controlli Automatici

Gian Antonio Susto

December 11, 2016

Octave rappresenta un'alternativa open-source all'utilizzo di MATLAB: Octave utilizza gli stessi comandi di MATLAB e diverse funzioni afferenti al pacchetto per i controlli automatici sono implementate. La versione piú aggiornata di Octave può essere scaricata dal repository:

<https://www.gnu.org/software/octave/download.html>

Vengono elencati di seguito i comandi fondamentali per i controlli automatici

- `pkg load control` - comando necessario per caricare il pacchetto 'Control'. La lista di tutte le funzioni implementate in questo pacchetto é disponibile al link

<https://octave.sourceforge.io/control/overview.html>

- `tf(<num>, <den>)` - necessario per costruire una funzione di trasferimento a tempo continuo; `tf(<num>, <den>)` ha come argomenti (<num>, <den>) che sono rispettivamente il numeratore e il denominatore della f.d.t. da definire. Per generare il numeratore (e analogamente per il denominatore) si costruisce un vettore di lunghezza pari al grado del denominatore in questo modo: sia il numeratore

$$N(s) = \alpha_m s^m + \alpha_{m-1} s^{m-1} + \dots + \alpha_1 s + \alpha_0,$$

il vettore corrispondente sará `num = [α_m α_{m-1} ... α_1 α_0]`.

Va prestata molta attenzione in caso di coefficienti α_i pari a zero ricordandosi di inserire anche i coefficienti nulli nel vettore: ad esempio $N(s) = s^3 + 1$ avrà come vettore associato `num = [1 0 0 1]`.

Con il comando `G = tf(<num>, <den>)`, la f.d.t. viene salvata nella variabile `G`.

- `conv(<vec1>, <vec2>)` - molto spesso le f.d.t. che si vogliono rappresentare sono in forma fattorizzata (di Bode o di Evans); utilizzando il comando `conv(<vec1>, <vec2>)` é possibile ottenere il vettore rappresentante il prodotto fra due polinomi (rappresentati da <vec1> e <vec2>) evitando un calcolo diretto. Esempio: sia

$$\begin{aligned} N(s) &= (s^2 + 13s + 1)(s - 10) \\ &= s^3 + 3s - 129s - 10, \end{aligned}$$

tale rappresentazione può essere ottenuta utilizzando:

`conv([1 13 1], [1 -10]) = [1 3 -129 -10]`.

- `bode(<fdt>)` - consente di tracciare i diagrammi di Bode di modulo e fase per la f.d.t. <fdt>.
- `margin(<fdt>)` - consente il calcolo del margine di fase e di guadagno per la f.d.t. <fdt>.

Con il comando `[Gamma, Phi, omegaGamma, omegaPhi] = margin(<fdt>)` vengono salvate le seguenti quantità:

- in `Gamma` viene salvato il valore del margine di guadagno;
- in `Phi` viene salvato il valore del margine di fase;

- in `omegaGamma` viene salvata la pulsazione dove viene calcolato il margine di guadagno (la pulsazione dove il diagramma della fase é pari a -180);
- in `omegaPhi` viene salvata la pulsazione dove viene calcolato il margine di fase (la pulsazione dove il diagramma del modulo é pari a 0 dB).
- `nyquist(<fdt>)` - consente di tracciare il diagramma di Nyquist per la f.d.t. `<fdt>`.
- `rlocus(<fdt>)` - consente di tracciare il luogo delle radici per la f.d.t. `<fdt>`.