Identificarea sistemelor – Laborator 10
Metoda ARX recursivă

Organizare

Recitiți partea de logistică din laboratorul 2, aceleași reguli se vor aplica și pentru acest laborator. Singurele lucruri care se schimbă sunt assignment-ul pe Teams, care pentru acest laborator este "Lab 10 (ARX recursiv)", și desigur numărul laboratorului în numele fișierului.

Descrierea laboratorului

Vom implementa în acest laborator varianta recursivă a metodei ARX, vezi cursul *Identificarea recursivă*. Vom interacționa online cu motorul de curent continuu, vezi secțiunea „Online mode" din ghid.

Cerințele pentru laborator sunt următoarele:

- Pentru început, vom aplica o secvență de zerouri, un semnal treaptă (care împreună cu ieșirea corespunzătoare va constitui setul de date de validare), urmat de o altă secvență de zerouri pentru a readuce motorul în condiții nule în pregătirea experimentului de identificare online. Perioada de eșantionare este 0.01 s (10 ms). Semnalul treaptă are amplitudinea de 0.3 și durează aproximativ 70 de eșantioane.
- Pentru a pregăti identificarea online, creați (fără a-l aplica încă!) un semnal de intrare de tip SPAB cu o lungime de aproximativ 200 de eșantioane și cu valori între -0.8 și 0.8. Pentru a genera semnalul SPAB, folosiți fie `idinput`, fie codul dvs. de la laboratoarele anterioare, dar în acest al doilea caz folosiți suficienți biți pentru ca semnalul să nu se repete.
- Implementați metoda ARX recursivă, care aplică pas cu pas motorului intrarea creată mai sus, și apoi actualizează online modelul ARX; vezi pseudocodul de mai jos, care conține mai multe detalii decât la curs. Codul trebuie să producă la ieșire o matrice $\Theta \in \mathbb{R}^{(na+nb)\times N}$ conținând pe fiecare coloană k vectorul de parametri $\theta(k)$: întâi coeficienții $a_1, \dots, a_{na}$ ai polinomului A, și apoi coeficienții $b_1, \dots, b_{nb}$ din B. Vectorul inițial de parametri $\theta(0)$ poate fi luat zero, iar matricea inversă inițială $P^{-1}(0) = 1000 I_{na+nb}$. Puteți încerca un model cu ordinele na si nb egale cu 2, modelul va merge mai bine chiar dacă sistemul este de ordinul 1.
- Comparați *pe datele de validare* calitatea a două modele: unul cu parametrii finali găsiți după procesarea întregului set de date; și altul după 5% din date. Care model este mai bun, și de ce? Indiciu: Puteți folosi `idpoly(A, B, [], [], [], 0, Ts)` urmat de `compare` pe un obiect de tip `iddata` (pe care va trebui să-l creați dvs). Nu uitați că vectorii de coeficienți din polinoame trebuie să fie de tip linie și să conțină coeficienții constanți (puterea 0 a argumentului q^{-1}), care trebuie să fie 1 în A, și 0 în B; matricea Θ de parametri *nu* conține acești coeficienți constanți.

inițializează $\hat{\theta}(0)$, un vector coloană $na + nb$
inițializează $P^{-1}(0)$, o matrice $(na + nb) \times (na + nb)$
for fiecare pas $k = 1, 2, \dots, N$ **do**
    aplică $u(k)$ sistemului, și citește $y(k)$ de la sistem
    formează vectorul de regresori ARX:
        $\varphi(k) = [-y(k-1), \cdots, -y(k-na), u(k-1), \cdots, u(k-nb)]^\top$
    găsește eroarea de predicție $\varepsilon(k) = y(k) - \varphi^\top(k)\hat{\theta}(k-1)$ (un scalar)
    actualizează inversa: $P^{-1}(k) = P^{-1}(k-1) - \frac{P^{-1}(k-1)\varphi(k)\varphi^\top(k)P^{-1}(k-1)}{1+\varphi^\top(k)P^{-1}(k-1)\varphi(k)}$
    calculează ponderile: $W(k) = P^{-1}(k)\varphi(k)$ (un vector coloană $(na + nb)$)
    actualizează parametrii: $\hat{\theta}(k) = \hat{\theta}(k-1) + W(k)\varepsilon(k)$
    important: așteaptă să se scurgă timpul rămas din perioada de eșantionare
end for