

Identificarea sistemelor – Laborator 6

Metoda ARX

Organizare

- Acest laborator se rezolvă independent de către fiecare student. Doar dacă există mai mulți studenți decât calculatoare, studenții se pot grupa câte doi la un calculator.
- Soluția constă din cod Matlab. Dezvoltați acest cod într-un singur script Matlab. Codul va fi verificat și rulat de către profesor, în timpul laboratorului, și prezența la laborator va fi validată doar dacă aveți o soluție funcțională și originală. Prezențele la toate laboratoarele trebuie validate pentru eligibilitate la examen. Cum cel mult două laboratoare se pot recupera la sfârșitul semestrului, acumularea a trei sau mai multe laboratoare lipsă la orice moment de timp duce automat la pierderea finală a eligibilității.
- Discutarea ideilor între studenți este permisă și chiar de dorit, dar copierea sau schimbul direct de cod este interzis. Încălcarea acestei reguli va duce la invalidarea soluției.

Descrierea laboratorului

La acest laborator vom identifica modele ARX (autoregresive cu intrare exogenă) folosind regresia liniară, vezi materialul de curs, Partea V: *Metoda ARX*.

Fiecărui student i se alocă de către profesor un index pentru setul de date. Apoi, studentul descarcă fișierul Matlab ce formează baza laboratorului de pe pagina cursului:

http://busoniu.net/teaching/sysid2018/index_ro.html

Fișierul conține datele de identificare în variabila `id`, și separat datele de validare în variabila `val`. Ambele variabile sunt obiecte de tip `iddata` din toolbox-ul Matlab de identificare a sistemelor, vezi `doc iddata`. Se știe de la operatorul sistemului că sistemul nu are timp mort (întârziere).

Cerințe:

- Implementați explicit identificarea ARX folosind regresia liniară, utilizând descrierea din curs. Reamintim că regresorii sunt $y(k-1), \dots, y(k-na), u(k-1), \dots, u(k-nb)$. Codul dvs. trebuie să funcționeze pentru orice valori na și nb . Implementați apoi și simularea modelului calculat cu intrările de validare. Țineți cont că pentru simulare, nu se cunosc ieșirile reale ale sistemului, deci se pot folosi doar ieșirile anterioare ale modelului însuși; mai exact $y(k-i)$ în formula modelului trebuie înlocuit cu valoarea sa simulată anterior $\hat{y}(k-i)$, pentru $i = 1, \dots, na$. Indiciu: semnalele la momente negative sau zero de timp pot fi luate 0.
- Reprezentați grafic și examinați datele furnizate.
- Încercați să intuiți o structură a sistemului din forma răspunsurilor din datele de validare. Alegeți corespunzător ordinele na și nb ale modelului ARX, și identificați un model folosind întâi codul dvs., și apoi funcția Matlab `arx` (cu $nk = 1$ fiindcă știm deja că sistemul nu are întârzieri). Comparați cele două rezultate.
- Creați apoi un set de structuri de model și folosiți funcțiile `***struc` pentru a găsi structura care recuperează cel mai bine datele de validare. Reprezentați grafic comparația modelului cu ieșirile de validare și comparați rezultatul cu cel obținut la punctul anterior. Acest pas folosește funcții Matlab, nu codul dvs.

- Gândiți-vă la rezultatele obținute. Are sistemul structura pe care ați intuit-o inițial? Dacă ordinele identificate automat sunt mai mari decât cele indicate de răspunsul la treaptă al sistemului, puteți identifica un motiv? (reamintindu-vă de discuția de la curs)?
- Opțional, dacă mai aveți timp, re-identificați un model cu cel mai bun ordin găsit cu `selstruct`, dar de data aceasta folosind codul dezvoltat de dvs., și verificați că obțineți rezultate similare.

Funcții relevante din toolbox-ul de identificare a sistemelor: `arx`, `struc`, `arxstruc`, `selstruc`, `plot`, `compare`. Când o funcție din toolbox-ul de identificare are același nume cu o funcție din alt toolbox – de exemplu `compare`, care suprascrie implementarea din toolbox-ul de MPC – scrieți de ex. `doc ident/compare` pentru a obține documentația variantei din `ident`. Vezi și `doc ident` pentru documentația completă a toolbox-ului.