Identificarea sistemelor

Ingineria sistemelor, anul 3
Universitatea Tehnică din Cluj-Napoca

Lucian Buşoniu

Partea V

Metoda ARX

Clasificare

Reamintim taxonomia modelelor din Partea I:

După numărul de parametri:

1. Modele parametrice: au formă fixă (formulă matematică), număr cunoscut şi de obicei mic de parametri
2. Modele neparametrice: nu pot fi descrise cu un număr fix, mic de parametri
   Adesea reprezentate prin grafice sau tabele

După cunoştinţele disponibile în avans ("culoare"):

1. Modele din principii de bază, cutie albă: complet cunoscute în avans
2. Modele cutie neagră: complet necunoscute în avans
3. Modele cutie gri: parţial cunoscute

Metoda ARX produce modele *parametrice*, de tip polinomial.

De ce ARX?

- Metodă pentru orice ordin, implementabilă programatic, cu garanţii – ca şi analiza de corelaţie
- Spre deosebire de analiza de corelaţie, furnizează un model *compact*, numărul de parametri fiind proporţional cu ordinul sistemului

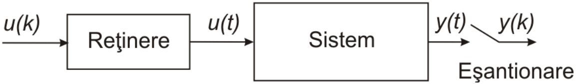
Conţinut

Vom rămâne în cazul cu intrări şi ieşiri scalare, cu excepţia anexei opţionale.
ARX neliniar este pentru proiect, nu va fi necesar pentru laboratoare.

Conţinut

Reamintim: Timp discret

Rămânem în cazul de timp discret:



Structura modelului ARX

În structura ARX, ieşirea $y(k)$ la pasul curent este calculată din intrările şi ieşirile la paşi precedenţi:

$$\begin{aligned} y(k) + a_1 y(k-1) + a_2 y(k-2) + \ldots + a_{na} y(k-na) \\ = b_1 u(k-1) + b_2 u(k-2) + \ldots + b_{nb} u(k-nb) + e(k) \end{aligned}$$

echivalent cu

$$\begin{aligned} y(k) = -a_1 y(k-1) - a_2 y(k-2) - \ldots - a_{na} y(k-na) \\ b_1 u(k-1) + b_2 u(k-2) + \ldots + b_{nb} u(k-nb) + e(k) \end{aligned}$$

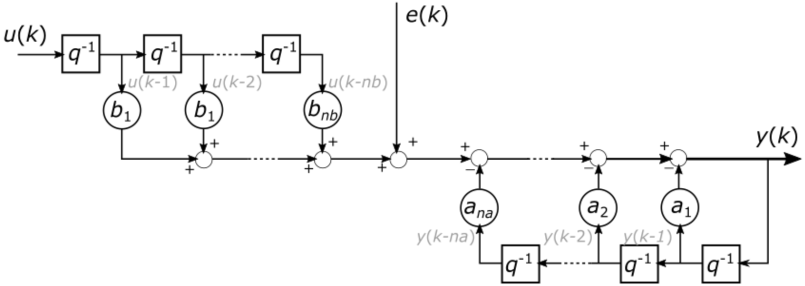
$e(k)$ este zgomotul la pasul k.

Parametrii modelului: $a_1, a_2, \ldots, a_{n_a}$ şi $b_1, b_2, \ldots, b_{nb}$.

Nume: Model AutoRegresiv ($y(k)$ depinde de valorile y precedente) cu intrare eXogenă (dependenţă de u)

Forma grafică

$$y(k) + a_1 y(k-1) + a_2 y(k-2) + \ldots + a_{na} y(k-na)$$
$$= b_1 u(k-1) + b_2 u(k-2) + \ldots + b_{nb} u(k-nb) + e(k)$$



Reprezentare polinomială

Operatorul de deplasare q^{-1}:

$$q^{-1}z(k) = z(k-1)$$

unde $z(k)$ este orice semnal în timp discret.

Folosind q^{-1}, avem:

$$\begin{aligned} & y(k) + a_1 y(k-1) + a_2 y(k-2) + \ldots + a_{na} y(k-na) \\ & \quad = (1 + a_1 q^{-1} + a_2 q^{-2} + \ldots + a_{na} q^{-na}) y(k) =: A(q^{-1}) y(k) \end{aligned}$$

şi:

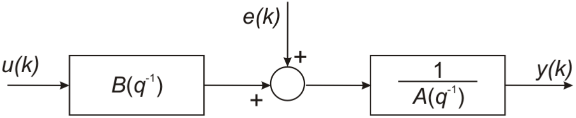
$$\begin{aligned} & b_1 u(k-1) + b_2 u(k-2) + \ldots + b_{nb} u(k-nb) \\ & \quad = (b_1 q^{-1} + b_2 q^{-2} + \ldots + b_{nb} q^{-nb}) u(k) =: B(q^{-1}) u(k) \end{aligned}$$

Modelul ARX în formă polinomială

Ca urmare, modelul ARX se poate scrie compact:

$$\boxed{A(q^{-1})y(k) = B(q^{-1})u(k) + e(k)}$$

Reprezentare grafică:



fiindcă:

$$y(k) = \frac{1}{A(q^{-1})}[B(q^{-1})u(k) + e(k)]$$

Observaţii: (1) Modelul ARX este general, poate descrie orice relaţie liniară între intrări şi ieşiri. Zgomotul participă însă în model într-o manieră restrictivă, şi mai târziu vom descrie modele care generalizează acest element. (2) Fără zgomot, am avea o funcţie standard de transfer în timp discret.

Model în forma de regresie liniară

Revenind la reprezentarea explicită recursivă:

$$
\begin{aligned}
y(k) = & -a_1 y(k-1) - a_2 y(k-2) - \ldots - a_{na} y(k-na) \\
& b_1 u(k-1) + b_2 u(k-2) + \ldots + b_{nb} u(k-nb) + e(k) \\
= & [-y(k-1), \ldots, -y(k-na), u(k-1), \ldots, u(k-nb)] \\
& \cdot [a_1, \ldots, a_{na}, b_1, \ldots, b_{nb}]^\top + e(k) \\
=: & \varphi^\top(k)\theta + e(k)
\end{aligned}
$$

ARX se supune aşadar formei standard de model din regresia liniară!

Vector de regresori şi parameteri

Vector de regresori: $\varphi(k) \in \mathbb{R}^{na+nb}$, ieşirile şi intrările precedente.

Vector de parametri: $\theta \in \mathbb{R}^{na+nb}$, coeficienţii polinoamelor.

$$\varphi(k) = \begin{bmatrix} -y(k-1) \\ \vdots \\ -y(k-na) \\ u(k-1) \\ \vdots \\ u(k-nb) \end{bmatrix} \qquad \theta = \begin{bmatrix} a_1 \\ \vdots \\ a_{na} \\ b_1 \\ \vdots \\ b_{nb} \end{bmatrix}$$

Conţinut

Problema de identificare

Considerăm un set de date $u(k), y(k)$, $k = 1, \dots, N$, din care trebuie aflaţi parametrii θ ai modelului.

Pentru orice k:

$$y(k) = \varphi^\top(k)\theta + \varepsilon(k)$$

unde $\varepsilon(k)$ este interpretat acum ca o eroare în ecuaţie (de unde şi notaţia schimbată).

Obiectiv: minimizarea erorii medii pătratice:

$$V(\theta) = \frac{1}{N}\sum_{k=1}^{N}\varepsilon(k)^2$$

Observaţie: Când $k \leq na, nb$, valori ale u şi y la momente de timp negative sau zero sunt necesare pentru construirea φ. Aceste valori pot fi luate 0 (presupunând condiţii iniţiale nule).

Sistem liniar de ecuaţii

$$y(1) = \begin{bmatrix} -y(0) & \cdots & -y(1-na) & u(0) & \cdots & u(1-nb) \end{bmatrix} \theta$$
$$y(2) = \begin{bmatrix} -y(1) & \cdots & -y(2-na) & u(1) & \cdots & u(2-nb) \end{bmatrix} \theta$$
$$\cdots$$
$$y(N) = \begin{bmatrix} -y(N-1) & \cdots & -y(N-na) & u(N-1) & \cdots & u(N-nb) \end{bmatrix} \theta$$

Forma matriceală:

$$\boxed{\begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(N) \end{bmatrix} = \begin{bmatrix} -y(0) & \cdots & -y(1-na) & u(0) & \cdots & u(1-nb) \\ -y(1) & \cdots & -y(2-na) & u(1) & \cdots & u(2-nb) \\ \vdots & \vdots & \vdots & \vdots & & \\ -y(N-1) & \cdots & -y(N-na) & u(N-1) & \cdots & u(N-nb) \end{bmatrix} \cdot \theta}$$

$$Y = \Phi\theta$$

cu notaţiile $Y \in \mathbb{R}^N$ şi $\Phi \in \mathbb{R}^{N \times (na+nb)}$.

Soluţia ARX

Din regresia liniară, parametrii care minimizează $\frac{1}{2}\sum_{k=1}^{N} \varepsilon(k)^2$ sunt:

$$\boxed{\widehat{\theta} = (\Phi^\top \Phi)^{-1} \Phi^\top Y}$$

Cum noua funcţie obiectiv $V(\theta) = \frac{1}{N}\sum_{k=1}^{N} \varepsilon(k)^2$ este proporţională cu criteriul de mai sus, aceeaşi soluţie minimizează şi $V(\theta)$.

Soluţia pentru seturi mari de date

În cazul în care numărul N de date este foarte mare, forma de mai sus este impractică pentru identificare. O formă mai bună este cea alternativă menţionată în partea de regresie liniară:

$$\Phi^\top \Phi = \sum_{k=1}^{N} \varphi(k)\varphi^\top(k), \quad \Phi^\top Y = \sum_{k=1}^{N} \varphi(k) y(k)$$

$$\Rightarrow \widehat{\theta} = \left[\sum_{k=1}^{N} \varphi(k)\varphi^\top(k)\right]^{-1} \left[\sum_{k=1}^{N} \varphi(k) y(k)\right]$$

Soluţia pentru seturi mari de date (continuare)

O problemă rămasă: suma celor N termeni poate fi mare, ducând la probleme numerice: (matrice de numere foarte mari)$^{-1}\cdot$ vector de numere foarte mari.

Soluţie: Normalizarea valorilor prin împărţirea fiecărui element cu N. În ecuaţii, N se simplifică deci nu are nici un efect asupra dezvoltării analitice, dar în practică această împărţire menţine numerele la valori rezonabile.

$$\widehat{\theta} = \left[\frac{1}{N}\sum_{k=1}^{N}\varphi(k)\varphi^\top(k)\right]^{-1}\left[\frac{1}{N}\sum_{k=1}^{N}\varphi(k)y(k)\right]$$

Cum rămâne cu împărţirea sumei la N? Se poate efectua recursiv, fără a implica niciodată numere mari - detalii mai târziu.

Utilizarea modelului

Predicţie cu un pas înainte: Secvenţa de ieşiri reale este cunoscută, deci toate valorile precedente sunt disponibile şi pot fi înlocuite în formulă, pe lângă coeficienţii extraşi din θ:

$$\begin{aligned}\hat{y}(k) = &- a_1 y(k-1) - a_2 y(k-2) - \ldots - a_{na} y(k-na) \\ & b_1 u(k-1) + b_2 u(k-2) + \ldots + b_{nb} u(k-nb)\end{aligned}$$

Semnalele la momente negative şi zero de timp pot fi luate 0.

Exemplu: În ziua $k-1$, predicţia meteo pentru ziua k.

Simulare: Ieşirile reale sunt necunoscute, trebuiesc înlocuite cu ieşirile simulate la iteraţii anterioare: $y(k-i)$ înlocuit cu $\hat{y}(k-i)$:

$$\begin{aligned}\hat{y}(k) = &- a_1 \hat{y}(k-1) - a_2 \hat{y}(k-2) - \ldots - a_{na} \hat{y}(k-na) \\ & b_1 u(k-1) + b_2 u(k-2) + \ldots + b_{nb} u(k-nb)\end{aligned}$$

(ieşirile simulate la paşi negativi şi zero pot fi şi ele luate 0.)

Exemplu: Simularea răspunsului unui avion la comenzi ale pilotului într-o situaţie de urgenţă, care ar fi periculoasă pentru sistemul real.

Notă despre utilizarea modelelor

Mai multe tipuri de modele se pot utiliza în predicţie sau simulare, nu doar ARX.

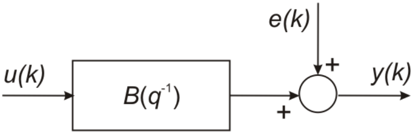
FIR este un caz particular de ARX

Alegând $A = 1$ ($na = 0$) în ARX, obţinem:

$$y(k) = B(q^{-1})u(k) + e(k) = \sum_{j=1}^{nb} b_j u(k-j) + e(k)$$
$$= \sum_{j=0}^{M-1} h(j)u(k-j) + e(k)$$

modelul FIR din analiza de corelaţie!

Mai detaliat, luăm $nb = M - 1$ şi $b_j = h(j)$. De notat că $h(0)$, răspunsul la impuls la momentul 0, se presupune egal cu 0 – adică sistemul nu răspunde instantaneu la schimbări ale intrării.



Diferenţa fundamentală între ARX şi FIR

$$\begin{aligned} \text{ARX:} \quad & A(q^{-1})y(k) = B(q^{-1})u(k) + e(k) \\ \text{FIR:} \quad & y(k) = B(q^{-1})u(k) + e(k) \end{aligned}$$

Cum ARX include o relaţie recursivă între ieşirea curentă şi cele precedente, este suficient să luăm ordinele *na* şi *nb* egale cu ordinul sistemului dinamic.

Modelul FIR are nevoie de un ordin *nb* (lungime M) suficient de mare pentru a modela întregul regim tranzitoriu al răspunsului la impuls (în mod ideal recuperăm modelul corect doar dacă $M \to \infty$).

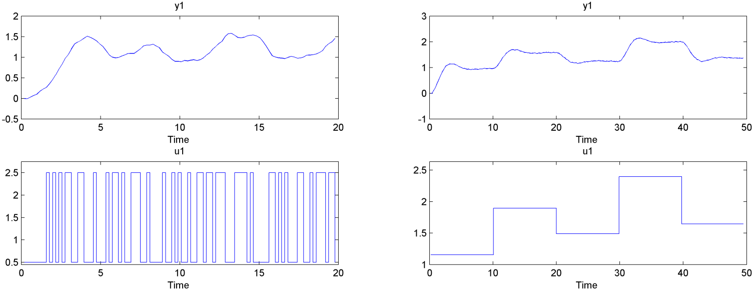
$\Rightarrow$ mai mulţi parametri $\Rightarrow$ mai multe date necesare pentru identificare.

Conţinut

Date experimentale

Se dau două seturi de date, unul pentru identificare, celălalt pentru validare.

`plot(id);` şi `plot(val);`



Observaţii: Intrarea de identificare: *semnal pseudo-aleator binar*. Intrarea de validare: o secvenţă de semnale treaptă.

Identificarea unui model ARX

```
model = arx(id, [na, nb, nk]);
```

Argumente funcţie:

1. Datele de identificare.
2. Vector conţinând ordinele A şi B, şi *întârzierea* nk.

Structura diferită de cea teoretică: include o întârziere minimă nk între intrări şi ieşiri, utilă pentru a modela sistemele cu timp mort.

$$
\begin{aligned}
y(k) + a_1 y(k-1) + a_2 y(k-2) + \ldots + a_{na} y(k-na) \\
= b_1 u(k-nk) + b_2 u(k-nk-1) + \ldots + b_{nb} u(k-nk-nb+1) + e(k)
\end{aligned}
$$

$$A(q^{-1})y(k) = B(q^{-1})u(k-nk) + e(k), \text{ unde:}$$

$$A(q^{-1}) = (1 + a_1 q^{-1} + a_2 q^{-2} + \ldots + a_{na} q^{-na})$$

$$B(q^{-1}) = (b_1 + b_2 q^{-1} + b_{nb} q^{-nb+1})$$

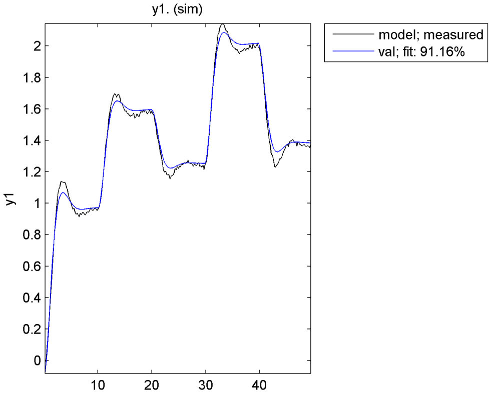
Structura teoretică se obţine luând $nk = 1$. Pentru $nk > 1$, noua structură se poate transforma în cea teoretică alegând un polinom B de ordinul $nk + nb - 1$, cu primii $nk - 1$ coeficienţi nuli:

$$B_{\text{theor}}(q^{-1}) = 0q^{-1} + \ldots 0q^{-nk+1} + b_1 q^{-nk} + \ldots + b_{nb} q^{-nk-nb+1}$$

Validarea modelului

Presupunând că sistemul este de ordinul 2, *în forma ARX*, şi fără întârzieri, alegem $na = 2, nb = 2, nk = 1$. Validare:

```
compare(model, val);
```



Rezultatele nu sunt bune.

Selecţia structurii

Alternativă: încercăm mai multe structuri diferite şi o alegem pe cea mai bună.

```
Na = 1:15;
Nb = 1:15;
Nk = 1:5;
NN = struc(Na, Nb, Nk);
V = arxstruc(id, val, NN);
```

- `struc` generează toate combinaţiile de ordine în `Na, Nb, Nk`.
- `arxstruc` identifică pentru fiecare combinaţie un model ARX (pe datele din primul argument), îl simulează (pe datele din al doilea argument), şi returnează toate valorile MSE pe prima linie din `V` (vezi `help arxstruc` pentru formatul variabilei `V`).

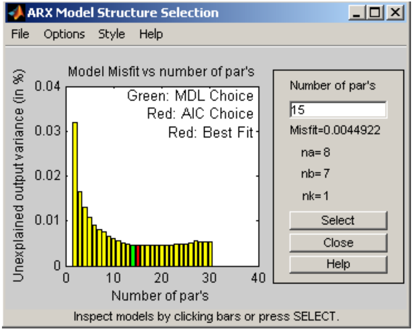
Selecţia structurii (continuare)

Pentru a alege structura cu valoarea MSE minimă:

```
N = selstruc(V, 0);
```

Pentru datele noastre, `N`$= [8, 7, 1]$.

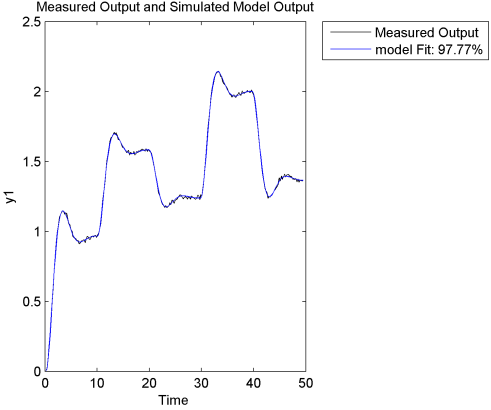
Alternativ, selecţie grafică: `N = selstruc(V, 'plot');` Click pe bara corespunzătoare modelului optim (roşu), apoi “Select”, “Close”.



(Mai târziu vom discuta selecţia între modele alternative cu alte criterii decât MSE.)

Validarea celui mai bun model ARX

```
model = arx(id, N); compare(model, val);
```



Rezultatele sunt mai bune. Sistemele de ordinul 8 sunt însă destul de rare în practică, probabil se întâmplă ceva mai complicat... vom relua ideea în cursurile următoare.

Conţinut

Rezultat

Ipoteze

1. Există un vector corect de parametri θ_0 pentru care:

$$y(k) = \varphi^\top(k)\theta_0 + v(k)$$

   unde $v(k)$ este un proces stohastic staţionar independent de $u(k)$.
2. $\mathrm{E}\left\{\varphi(k)\varphi^\top(k)\right\}$ este o matrice inversabilă.
3. $\mathrm{E}\left\{\varphi(k)v(k)\right\} = 0$.

Teoremă

Identificarea ARX este consistentă: parametrii estimaţi $\widehat{\theta}$ converg la cei corecţi θ_0, la limită când numărul de date creşte la infinit $N \to \infty$.

Discuţie ipoteze

1. Ipoteza 1 este echivalentă cu existenţa unor polinoame corecte $A_0(q^{-1})$, $B_0(q^{-1})$ pentru care:

$$A_0(q^{-1})y(k) = B_0(q^{-1})u(k) + v(k)$$

Pentru a motiva Ipoteza 2, reamintim

$$\widehat{\theta} = \left[\frac{1}{N}\sum_{k=1}^{N} \varphi(k)\varphi^\top(k)\right]^{-1} \left[\frac{1}{N}\sum_{k=1}^{N} \varphi(k)y(k)\right]$$

Când $N \to \infty$, $\frac{1}{N}\sum_{k=1}^{N} \varphi(k)\varphi^\top(k) \to \mathrm{E}\left\{\varphi(k)\varphi^\top(k)\right\}$.

2. $\mathrm{E}\left\{\varphi(k)\varphi^\top(k)\right\}$ este inversabilă dacă datele sunt “suficient de informative” (de ex. $u(k)$ nu trebuie să fie un feedback simplu de la $y(k)$; vezi Söderström & Stoica pentru discuţii adiţionale).
3. $\mathrm{E}\{\varphi(k)v(k)\} = 0$ de ex. dacă $v(k)$ este zgomot alb. Mai târziu, vom rediscuta Ipoteza 3 şi rolul condiţiei $\mathrm{E}\{\varphi(k)v(k)\} = 0$.

Rezumat

- Structura de model ARX şi reprezentarea cu polinoame în q^{-1}
- Forma de regresie liniară cu regresori φ şi parametri θ
- Soluţia problemei de regresie liniară. Rescriere pentru seturi mari de date
- Utilizarea modelului pentru predicţie şi simulare
- Relaţia cu FIR

- Exemplu Matlab.
- Garanţie de acurateţe simplificată.

Conţinut

Structura ARX neliniară

Reamintim ARX standard:

$$\begin{aligned} y(k) = & - a_1 y(k-1) - a_2 y(k-2) - \ldots - a_{na} y(k-na) \\ & b_1 u(k-1) + b_2 u(k-2) + \ldots + b_{nb} u(k-nb) + e(k) \end{aligned}$$

Dependenţă liniară de ieşirile precedente $y(k-1), \ldots, y(k-na)$ şi intrările precedente $u(k-1), \ldots, u(k-nb)$.

ARX neliniar (NARX) generalizează la orice dependenţă neliniară:

$$\begin{aligned} y(k) = & g\big(y(k-1), y(k-2), \ldots, y(k-na), \\ & u(k-1), u(k-2), \ldots, u(k-nb); \theta\big) + e(k) \end{aligned}$$

Funcţia g este parametrizată de $\theta \in \mathbb{R}^n$, şi aceşti parametri pot fi aleşi pentru a recupera datele de identificare, identificând astfel un sistem dinamic neliniar.

NARX polinomial

În cazul nostru particular, g este un polinom de gradul m în ieşirile şi intrările precedente:

$$\begin{aligned} y(k) &= p(y(k-1), \dots, y(k-na), u(k-1), \dots, u(k-nb)) + e(k) \\ &=: p(d(k)) + e(k) \end{aligned}$$

unde $d(k) = [y(k-1), \dots, y(k-na), u(k-1), \dots, u(k-nb)]^\top$ este vectorul de semnale precedente.

De ex., pentru ordinele $na = nb = 1$ (de unde $d(k) = [y(k-1), u(k-1)]^\top$) şi gradul $m = 1$, modelul este:

$$y(k) = ay(k-1) + bu(k-1) + c + e(k)$$

şi dacă impunem în plus $c = 0$, recuperăm modelul ARX liniar

NARX polinomial (continuare)

Pentru aceleaşi $na = nb = 1$ şi gradul $m = 2$:

$$\begin{aligned} y(k) =& ay(k-1) + bu(k-1) + cy(k-1)^2 \\ &+ du(k-1)^2 + wu(k-1)y(k-1) + z + e(k) \end{aligned}$$

Observaţii:

- A nu se confunda cu forma polinomială liniară $A(q^{-1})y(k) = B(q^{-1})u(k) + e(k)$
- Parametrii sunt acum coeficienţii tuturor polinoamelor, de ex. $\theta = [a, b, c, d, w, z]^\top$
- Regresia liniară funcţionează ca de obicei, găsind parametrii care minimizează valoarea MSE-ului!
- y şi u la momente zero şi negative pot fi luate 0, presupunând că sistemul este în condiţii iniţiale nule

Reamintim predicţie versus simulare

Predicţie cu un pas înainte: Ieşirea reală a sistemului este cunoscută, deci vectorul de semnale întârziate $d(k)$ este disponibil:

$$
\begin{aligned}
d(k) &= [y(k-1), \dots, y(k-na), u(k-1), \dots, u(k-nb)]^\top \\
\widehat{y}(k) &= g(d(k); \widehat{\theta})
\end{aligned}
$$

Simulare: Ieşirea reală este necunoscută, folosim ieşirile simulate anterior pentru a construi o *aproximare* a lui $d(k)$:

$$
\begin{aligned}
\widehat{d}(k) &= [\widehat{y}(k-1), \dots, \widehat{y}(k-na), u(k-1), \dots, u(k-nb)]^\top \\
\widehat{y}(k) &= g(\widehat{d}(k); \widehat{\theta})
\end{aligned}
$$

Anexă: Intrări şi ieşiri multiple

Sistem MIMO

Până acum am considerat $y(k) \in \mathbb{R}$, $u(k) \in \mathbb{R}$,
sisteme *Single-Input, Single-Output (SISO)*

Multe sisteme sunt *Multiple-Input, Multiple-Output (MIMO)*.
De ex., avion. Intrări: putere motoare, eleron, elevator, cârmă.
Ieşiri: viteză, deviaţii unghiulare în jurul celor trei axe.

ARX MIMO

Mai departe considerăm $y(k), e(k) \in \mathbb{R}^{ny}$, $u(k) \in \mathbb{R}^{nu}$.
Model ARX MIMO:

$$\begin{aligned} A(q^{-1})y(k) &= B(q^{-1})u(k) + e(k) \\ A(q^{-1}) &= I + A_1 q^{-1} + \ldots + A_{na} q^{-na} \\ B(q^{-1}) &= B_1 q^{-1} + \ldots + B_{nb} q^{-nb} \end{aligned}$$

unde I este matricea identitate $ny \times ny$, $A_1, \ldots, A_{na} \in \mathbb{R}^{ny \times ny}$, $B_1, \ldots, B_{nb} \in \mathbb{R}^{ny \times nu}$.

Exemplu concret

Luăm $na = 1$, $nb = 2$, $ny = 2$, $nu = 3$. Atunci:

$$
\begin{aligned}
A(q^{-1})y(k) &= B(q^{-1})u(k) + e(k) \\
A(q^{-1}) &= I + A_1 q^{-1} \\
&= I + \begin{bmatrix} a_1^{11} & a_1^{12} \\ a_1^{21} & a_1^{22} \end{bmatrix} q^{-1} \\
B(q^{-1}) &= B_1 q^{-1} + B_2 q^{-2} \\
&= \begin{bmatrix} b_1^{11} & b_1^{12} & b_1^{13} \\ b_1^{21} & b_1^{22} & b_1^{23} \end{bmatrix} q^{-1} + \begin{bmatrix} b_2^{11} & b_2^{12} & b_2^{13} \\ b_2^{21} & b_2^{22} & b_2^{23} \end{bmatrix} q^{-2}
\end{aligned}
$$

Exemplu concret (continuare)

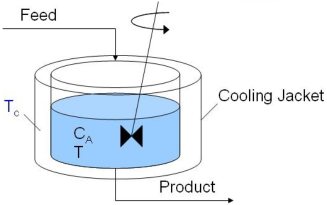
$$\left(\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} + \begin{bmatrix}a_1^{11} & a_1^{12}\\ a_1^{21} & a_1^{22}\end{bmatrix} q^{-1}\right)\begin{bmatrix}y_1(k)\\ y_2(k)\end{bmatrix}$$
$$= \left(\begin{bmatrix}b_1^{11} & b_1^{12} & b_1^{13}\\ b_1^{21} & b_1^{22} & b_1^{23}\end{bmatrix} q^{-1} + \begin{bmatrix}b_2^{11} & b_2^{12} & b_2^{13}\\ b_2^{21} & b_2^{22} & b_2^{23}\end{bmatrix} q^{-2}\right)\begin{bmatrix}u_1(k)\\ u_2(k)\\ u_3(k)\end{bmatrix} + \begin{bmatrix}e_1(k)\\ e_2(k)\end{bmatrix}$$

Relaţie explicită:

$$\begin{aligned}
y_1(k) &+ a_1^{11} y_1(k-1) + a_1^{12} y_2(k-1)\\
&= b_1^{11} u_1(k-1) + b_1^{12} u_2(k-1) + b_1^{13} u_3(k-1)\\
&+ b_2^{11} u_1(k-2) + b_2^{12} u_2(k-2) + b_2^{13} u_3(k-2) + e_1(k)\\
y_2(k) &+ a_1^{21} y_1(k-1) + a_1^{22} y_2(k-1)\\
&= b_1^{21} u_1(k-1) + b_1^{22} u_2(k-1) + b_1^{23} u_3(k-1)\\
&+ b_2^{21} u_1(k-2) + b_2^{22} u_2(k-2) + b_2^{23} u_3(k-2) + e_2(k)
\end{aligned}$$

Exemplu Matlab

Considerăm un reactor de tip *continuous stirred-tank reactor*, CSTR:



Credit imagine: mathworks.com

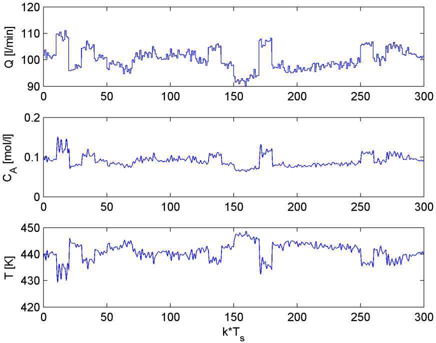
Intrare: debit Q al agentului de răcire

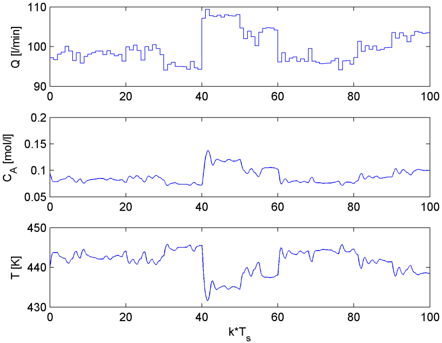
Ieşiri:

- Concentraţia C_A a substanţei A în amestec
- Temperatura T a amestecului

Matlab: Date experimentale

Stânga: identificare, Dreapta: validare





Matlab: ARX MIMO, diferit de teorie

$$A(q^{-1})y(k) = B(q^{-1})u(k) + e(k)$$

$$A(q^{-1}) = \begin{bmatrix} a^{11}(q^{-1}) & a^{12}(q^{-1}) & \dots & a^{1ny}(q^{-1}) \\ a^{21}(q^{-1}) & a^{22}(q^{-1}) & \dots & a^{2ny}(q^{-1}) \\ \dots & & & \\ a^{ny1}(q^{-1}) & a^{ny2}(q^{-1}) & \dots & a^{nyny}(q^{-1}) \end{bmatrix}$$

$$a^{ij}(q^{-1}) = \begin{cases} 1 & \text{dacă } i = j \\ 0 & \text{altfel} \end{cases} + a_1^{ij}q^{-1} + \ldots + a_{na_{ij}}^{ij}q^{-na_{ij}}$$

$$B = \begin{bmatrix} b^{11}(q^{-1}) & b^{12}(q^{-1}) & \dots & b^{1nu}(q^{-1}) \\ b^{21}(q^{-1}) & b^{22}(q^{-1}) & \dots & b^{2nu}(q^{-1}) \\ \dots & & & \\ b^{ny1}(q^{-1}) & b^{ny2}(q^{-1}) & \dots & b^{nynu}(q^{-1}) \end{bmatrix}$$

$$b^{ij}(q^{-1}) = b_1^{ij}q^{-nk_{ij}} + \ldots + b_{nb_{ij}}^{ij}q^{-nk_{ij}-nb_{ij}+1}$$

Matlab: Identificarea modelului

```
m = arx(id, [Na, Nb, Nk]);
```

Argumente funcţie:

1. Datele de identificare.
2. Matrici cu ordinele polinoamelor din *A*, *B*, şi *întârzieri nk*:

$$Na = \begin{bmatrix} na_{11} & \dots & na_{1ny} \\ \dots & & \\ na_{ny1} & \dots & na_{nyny} \end{bmatrix}$$

$$Nb = \begin{bmatrix} nb_{11} & \dots & nb_{1nu} \\ \dots & & \\ nb_{ny1} & \dots & nb_{nynu} \end{bmatrix}$$

$$Nk = \begin{bmatrix} nk_{11} & \dots & nk_{1nu} \\ \dots & & \\ nk_{ny1} & \dots & nk_{nynu} \end{bmatrix}$$

Matlab: Rezultate

Luăm $na = 2$, $nb = 2$, şi $nk = 1$ peste tot in elementele matricilor:

```
Na = [2 2; 2 2]; Nb = [2; 2]; Nk = [1; 1];
m = arx(id, [Na Nb Nk]);
compare(m, val);
```