

Identificarea sistemelor

Ingineria sistemelor, anul 3
Universitatea Tehnică din Cluj-Napoca

Lucian Buşoniu



Partea IV

Analiza de corelație

Conținut

- 1 Metoda analizei de corelație
 - Derivare analitică
 - Un algoritm practic. Modelul FIR
- 2 Garanție de acuratețe (simplificată)
- 3 Exemplu Matlab

Motivație 1

De ce alte metode pe lângă de analiza în domeniul timp?

Analiza în domeniul timp a răspunsurilor la treaptă și impuls:

- Se poate aplica doar pentru câteva valori ale ordinului sistemului
- Trebuie de obicei efectuată (semi-)manual
- Produce un model imprecis, euristic al sistemului

Metodele de identificare pe care le vom discuta mai departe:

- Funcționează pentru orice ordin al sistemului
- Furnizează algoritmi automați, complet implementabili
- Garantează acuratețea soluției (în anumite condiții tehnice)

Motivație 2

De ce analiza de corelație?

- Cea mai apropiată de analiza în domeniul timp (modelul este răspunsul la impuls)
- Model “cu adevărat” neparametric
- O metodă “simplă” din rândul tehnicilor generale de identificare

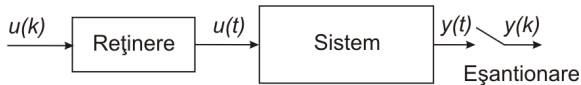
Clasificare

Reamintim clasificarea modelelor din Partea I:

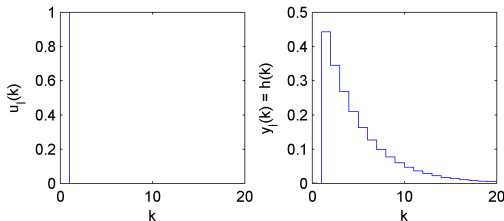
- ① Modele mentale sau verbale
- ② *Grafice și tabele (neparametrice)*
- ③ Modele matematice, cu două subtipuri:
 - Modele analitice, din principii de bază
 - Modele din identificarea sistemelor

Analiza de corelație este o metodă cu adevărat neparametrică;
produce un *model sub formă de răspuns la impuls*.

Reamintim: model în timp discret



Răspunsul discret la impuls



Semnal impuls unitar în timp discret:

$$u_I(k) = \begin{cases} 1 & k = 0 \\ 0 & k > 0 \end{cases}$$

(nu are aria 1, fiind așadar diferit de realizarea în timp discret a impulsului continuu!)

Răspuns discret la impuls:

$$y_I(k) = h(k), \quad k \geq 0$$

$h(k), k \geq 0$ se numește și **funcția pondere** a sistemului.

Convoluție

Răspunsul (fără perturbații) la o intrare arbitrară $u(k)$ este *convoluția* intrării cu răspunsul discret la impuls:

$$y(k) = \sum_{j=0}^{\infty} h(j)u(k-j)$$

Intuiție: Luăm semnalul $\tilde{u}_j(k)$ egal $u(j)$ at $k = j$, și 0 în rest; $\tilde{u}_j(k)$ este o versiune deplasată și scalată a impulsului unitar:

$$\tilde{u}_j(k) = u(j)u_1(k-j)$$

Răspunsul la $\tilde{u}_j(k)$ este așadar o versiune deplasată și scalată a răspunsului la impuls:

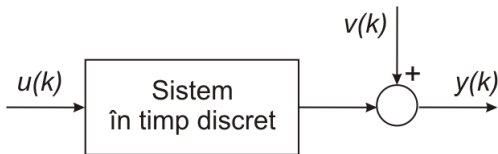
$$\tilde{y}_j(k) = u(j)h(k-j)$$

Dar $u(k)$ = superpoziția mai multor semnale $\tilde{u}_j$, și datorită liniarității:

$$y(k) = \sum_{j=0}^k \tilde{y}_j(k) = \sum_{j=0}^k u(j)h(k-j) = \sum_{j=0}^k h(j)u(k-j) = \sum_{j=0}^{\infty} h(j)u(k-j)$$

unde am presupus condiții inițiale zero, i.e. $u(j) = 0 \forall j < 0$.

Model de tip răspuns la impuls



$$y(k) = \sum_{j=0}^{\infty} h(j)u(k-j) + v(k)$$

Include pe lângă modelul ideal și o componentă de perturbație $v(k)$.

Ipoteze

Ipoteze

- 1 Intrarea $u(k)$ este un proces stohastic staționar.
- 2 Intrarea $u(k)$ și perturbația $v(k)$ sunt independente.

Reamintim:

- Independența variabilelor aleatoare.
- Proces stohastic staționar: aceeași medie la orice moment de timp, covarianța depinde doar de diferența între pașii de timp.

Funcții de covarianță

Funcțiile de covarianță de definesc astfel:

$$r_{yu}(\tau) = E \{y(k + \tau)u(k)\}$$

$$r_u(\tau) = E \{u(k + \tau)u(k)\}$$

Observație: Aceste cantități sunt covarianțele adevărate doar dacă intrarea și ieșirea sunt de medie zero. Dacă această condiție nu este satisfăcută, atunci mediile nonzero trebuie eliminate din semnale înainte de a aplica algoritmul.

Relația între covarianțe și răspunsul la impuls

Dacă nu ar exista perturbații, atunci:

$$\begin{aligned} r_{yu}(\tau) &= E \{ y(k + \tau) u(k) \} \\ &= E \left\{ \left[\sum_{j=0}^{\infty} h(j) u(k + \tau - j) \right] u(k) \right\} \\ &= \sum_{j=0}^{\infty} h(j) E \{ u(k + \tau - j) u(k) \} = \sum_{j=0}^{\infty} h(j) r_u(\tau - j) \end{aligned}$$

Erorile generate de perturbații sunt tratate implicit mai târziu, folosind regresia liniară.

Identificarea răspunsului la impuls

Scriem ecuația covarianțelor pentru toate valorile τ :

$$r_{yu}(0) = \sum_{j=0}^{\infty} h(j)r_u(-j) = h(0)r_u(0) + h(1)r_u(-1) + h(2)r_u(-2) + \dots$$

$$r_{yu}(1) = \sum_{j=0}^{\infty} h(j)r_u(1-j) = h(0)r_u(1) + h(1)r_u(0) + h(2)r_u(-1) + \dots$$

...

obținând (în principiu) un sistem infinit de ecuații liniare:

- Coeficienții sunt $r_u(\tau)$, $r_{yu}(\tau)$.
- Necunoscutele sunt $h(0)$, $h(1)$, $\dots$: soluția sistemului.

Urmează un algoritm practic, ce folosește un set finit de date.

Conținut

- 1 Metoda analizei de corelație
 - Derivare analitică
 - Un algoritm practic. Modelul FIR
- 2 Garanție de acuratețe (simplificată)
- 3 Exemplu Matlab

Obținerea covarianțelor din date

Se dau semnalele $u(k)$, $y(k)$, unde $k = 1, \dots, N$.

Pentru valori pozitive τ , avem:

$$r_u(\tau) = E \{u(k + \tau)u(k)\}$$

$$\approx \frac{1}{N - \tau} \sum_{k=1}^{N-\tau} u(k + \tau)u(k)$$

$$=: \hat{r}_u(\tau), \quad \forall \tau \geq 0$$

și $\hat{r}_u(-\tau) = \hat{r}_u(\tau)$ pentru $\tau < 0$, u fiind un proces staționar.

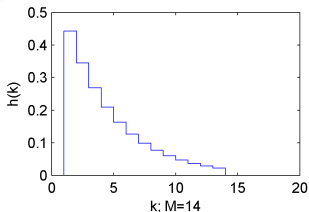
Pentru valori τ pozitive și negative:

$$r_{yu}(\tau) = E \{y(k + \tau)u(k)\}$$

$$\approx \frac{1}{N - |\tau|} \sum_{k=1-\min\{\tau,0\}}^{N-\max\{\tau,0\}} y(k + \tau)u(k)$$

$$=: \hat{r}_{yu}(\tau), \quad \forall \tau \geq 0$$

Modelul răspuns finit la impuls



Impunem condiția $h(k) = 0$ pentru $k \geq M$. Obținem modelul de tip **răspuns finit la impuls** (en. *finite impulse response*, FIR):

$$y(k) = \sum_{j=0}^{M-1} h(j)u(k-j) + v(k)$$

Equația covarianțelor este trunchiată în același fel:

$$r_{yu}(\tau) = \sum_{j=0}^{M-1} h(j)r_u(\tau-j)$$

De notat: M trebuie selectat pentru a avea $MT_s \gg$ constantele de timp dominante (sistemul să fie aproape în regim staționar)

Sistem liniar de ecuații

Folosind $\hat{r}_{yu}$, $\hat{r}_u$ estimate din date, scriem ecuațiile trunchiate pentru $\tau = 0, \dots, T-1$ (ținând cont că $\hat{r}_u(-\tau) = \hat{r}_u(\tau)$):

$$\begin{aligned}\hat{r}_{yu}(0) &= \sum_{j=0}^{M-1} h(j)\hat{r}_u(-j) \\ &= h(0)\hat{r}_u(0) + h(1)\hat{r}_u(1) + \dots + h(M-1)\hat{r}_u(M-1)\end{aligned}$$

$$\begin{aligned}\hat{r}_{yu}(1) &= \sum_{j=0}^{M-1} h(j)\hat{r}_u(1-j) \\ &= h(0)\hat{r}_u(1) + h(1)\hat{r}_u(0) + \dots + h(M-1)\hat{r}_u(M-2)\end{aligned}$$

...

$$\begin{aligned}\hat{r}_{yu}(T-1) &= \sum_{j=0}^{M-1} h(j)\hat{r}_u(T-1-j) \\ &= h(0)\hat{r}_u(T-1) + h(1)\hat{r}_u(T-2) + \dots + h(M-1)\hat{r}_u(T-M)\end{aligned}$$

– un sistem liniar de T ecuații cu M necunoscute $h(0), \dots, h(M-1)$.

Sistem liniar (continuaare)

În formă matriceală:

$$\begin{bmatrix} \hat{r}_{yu}(0) \\ \hat{r}_{yu}(1) \\ \vdots \\ \hat{r}_{yu}(T-1) \end{bmatrix} = \begin{bmatrix} \hat{r}_u(0) & \hat{r}_u(1) & \dots & \hat{r}_u(M-1) \\ \hat{r}_u(1) & \hat{r}_u(0) & \dots & \hat{r}_u(M-2) \\ \vdots & \vdots & & \vdots \\ \hat{r}_u(T-1) & \hat{r}_u(T-2) & \dots & \hat{r}_u(T-M) \end{bmatrix} \cdot \begin{bmatrix} h(0) \\ h(1) \\ \vdots \\ h(M-1) \end{bmatrix}$$

Selecția naivă $T = M$ ar furniza o soluție exactă a sistemului, dar datorită zgomotului și perturbațiilor această soluție ar fi supra-antrenată. Este așadar necesar să avem $T > M$ (preferabil, $T \gg M$).

Putem acum aplica metodologia de regresie liniară (vezi Partea 3) pentru a rezolva problema.

Utilizarea modelului FIR

După ce sistemul a fost rezolvat obținând ponderile estimate $\hat{h}$, prezicem ieșirea cu:

$$\hat{y}(k) = \sum_{j=0}^{M-1} \hat{h}(j)u(k-j)$$

Conținut

- 1 Metoda analizei de corelație
- 2 Garanție de acuratețe (simplificată)
- 3 Exemplu Matlab

Caz special: Intrare de tip zgomot alb

Ipoteză adițională

③ Intrarea $u(k)$ este zgomot alb de medie zero.

Atunci $r_u(\tau) = 0$ pentru orice $\tau \neq 0$ (zgomotul alb fiind necorelat), iar $r_{yu}(\tau) = \sum_{j=0}^{\infty} h(j)r_u(\tau - j)$ se reduce la:

$$r_{yu}(\tau) = h(\tau)r_u(0)$$

ducând la algoritmul foarte simplu:

$$h(\tau) = \frac{\hat{r}_{yu}(\tau)}{\hat{r}_u(0)}$$

Garanție simplificată

Teoremă

Pentru intrare de tip zgomot alb, valorile estimate $\hat{h}(\tau)$ converg la valorile reale $h(\tau)$ când numărul de eșantioane N tinde la infinit.

Observație: Acest tip de rezultat, în care soluția corectă este obținută la limita numărului infinit de date, se numește *consistență*.

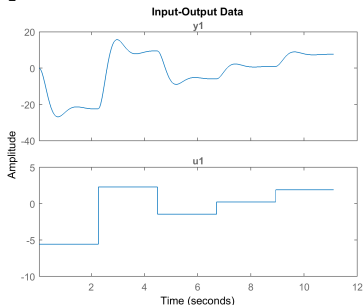
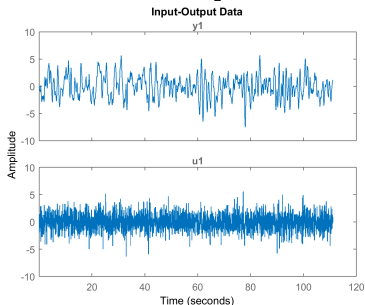
Conținut

- 1 Metoda analizei de corelație
- 2 Garanție de acuratețe (simplificată)
- 3 Exemplu Matlab

Date experimentale

Se dau următoarele seturi de date, separate pentru identificare și validare.

```
plot(id); and plot(val);
```



De notat că intrarea de identificare este zgomot alb, dar intrarea de validare nu este. Setul de identificare conține 1100 de eșantioane.

Aplicarea analizei de corelație

```
fir = cra(id, M, 0); sau fir = cra(id, M, 0, plotlevel);
```

Argumentele funcției:

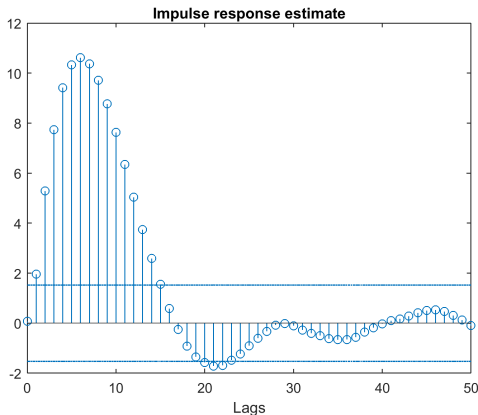
- 1 Datele de identificare.
- 2 Lungimea M a modelului de tip FIR, fixată aici la 45.
- 3 Al treilea argument egal cu 0 înseamnă ca nu se efectuează *albirea intrării*.

Tratarea intrărilor ne-ideale:

- Dacă intrarea nu are medie zero, setul de identificare trebuie trecut prin funcția `detrend` pentru a scădea valorile medii din semnale.
- Dacă intrarea nu este zgomot alb, al treilea argument trebuie lăsat egal cu valoarea implicită (nespecificându-l, sau fixându-l egal cu o matrice vidă), ceea ce va duce la albirea semnului de intrare.

Aplicarea analizei de corelație (continuare)

Implicit (sau când `plotlevel=1`) parametrii modelului FIR sunt reprezentați grafic împreună cu un interval de încredere de 99%.

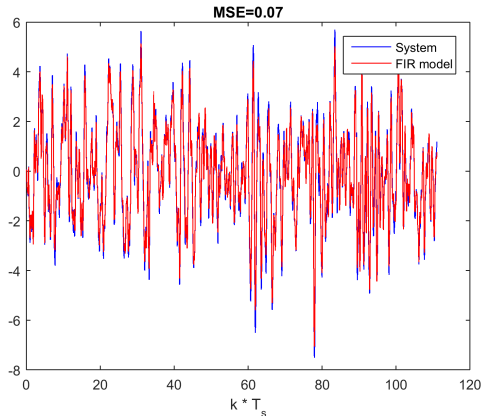


`plotlevel=2` reprezintă grafic de asemenea și funcțiile de covarianță.

Rezultate pe datele de identificare

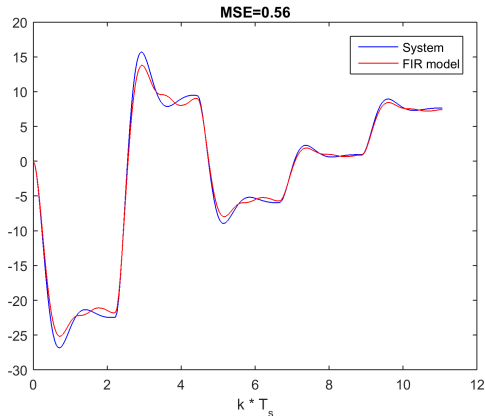
```
yhat = conv(fir, id.u); yhat = yhat(1:length(id.u));
```

Pentru a simula modelul FIR, trebuie efectuată *convoluția* între parametrii FIR și intrare. ieșirea simulată este mai lungă decât este necesar, și este așadar trunchiată la lungimea corectă.



Validarea modelului FIR

```
yhat = conv(fir, val.u); yhat = yhat(1:length(val.u));
```



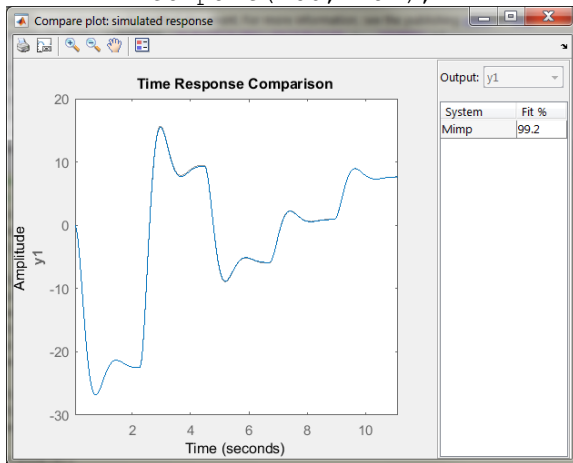
Rezultatele sunt rezonabile, dar nu excelente.

Alternativă: funcția impulseest

```
model = impulseest(id, M); or model = impulseest(id);
```

Folosește un algoritm mai avansat decât cel studiat la curs.

```
compare(mod, val);
```



Aproximarea este foarte bună.