Identificarea sistemelor

Ingineria sistemelor, anul 3
Universitatea Tehnică din Cluj-Napoca

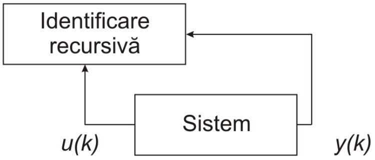
Lucian Buşoniu

Partea IX

Identificarea recursivă

Conţinut

Identificarea recursivă: Idee



Metodele recursive pot funcţiona *online*, în paralel cu sistemul.

La pasul k, calculează o nouă *estimare a parametrilor* $\widehat{\theta}(k)$, folosind estimarea anterioară $\widehat{\theta}(k-1)$ şi noile date disponibile $u(k)$, $y(k)$.

Observaţie: Pentru a le diferenţia de identificarea recursivă, vom numi metodele precedente, care folosesc întregul set de date deodată, metode *offline*.

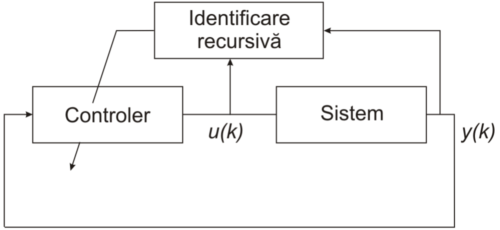
Motivare

Metodele recursive:

- Necesită mai puţină memorie, şi mai puţin timp de calcul *pentru fiecare actualizare*, decât algoritmul offline.

⇒ Mai uşor de aplicat în timp real.
(timpul total, după setul complet de date, poate fi mai mare)

- Cu modificările potrivite, pot trata sistemele variabile în timp.
- Dacă modelul este folosit pentru a proiecta un regulator, obţinem o schemă de *control adaptiv*:



Dezavantaj

Metodele recursive sunt în general o aproximare a metodelor offline
$\Rightarrow$ garanţiile de performanţă sunt mai greu de obţinut.

Exemplu motivant: Estimarea unui scalar

Model:

$$y(k) = b + e(k) = 1 \cdot b + e(k) = \varphi(k)\theta + e(k)$$

unde $\varphi(k) = 1 \forall k$, $\theta = b$.

Pentru măsurătorile până la (inclusiv) k:

$$\begin{aligned} y(1) &= \varphi(1)\theta = 1 \cdot b \\ &\dots \\ y(k) &= \varphi(k)\theta = 1 \cdot b \end{aligned}$$

După câteva calcule, soluţia este:

$$\widehat{\theta}(k) = \frac{1}{k}[y(1) + \ldots + y(k)]$$

(estimarea este media măsurătorilor, filtrând zgomotul).

Estimarea unui scalar: Formulare recursivă

Rescriem formula:

$$\begin{aligned}
\widehat{\theta}(k) &= \frac{1}{k}[y(1) + \ldots + y(k)] \\
&= \frac{1}{k}[(k-1)\frac{1}{k-1}(y(1) + \ldots + y(k-1)) + y(k)] \\
&= \frac{1}{k}[(k-1)\widehat{\theta}(k-1) + y(k)]
\end{aligned}$$

(deja o formulă recursivă, dar continuăm pentru a înţelege mai mult)

$$\begin{aligned}
&= \frac{1}{k}[k\widehat{\theta}(k-1) + y(k) - \widehat{\theta}(k-1)] \\
&= \widehat{\theta}(k-1) + \frac{1}{k}[y(k) - \widehat{\theta}(k-1)]
\end{aligned}$$

Estimarea unui scalar: Proprietăţi

$$\widehat{\theta}(k) = \widehat{\theta}(k-1) + \frac{1}{k}[y(k) - \widehat{\theta}(k-1)]$$

Metoda are multe din proprietăţile tehnicilor recursive generale:

- Formulă recursivă: noua estimare $\widehat{\theta}(k)$ calculată folosind estimarea anterioară $\widehat{\theta}(k-1)$ şi noua măsurătoare $y(k)$.
- $[y(k) - \widehat{\theta}(k-1)]$ este o *eroare de predicţie* $\varepsilon(k)$, fiindcă $\widehat{\theta}(k-1) = \widehat{b} = \widehat{y}(k)$, predicţia cu un pas înainte a ieşirii.
- Actualizarea aplică o corecţie proporţională cu eroarea $\varepsilon(k)$, ponderată de $\frac{1}{k}$:
  - Când eroarea $\varepsilon(k)$ este mare (sau mică), se efectuează o corecţie mare (sau mică).
  - Ponderea $\frac{1}{k}$ descreşte cu pasul k, ducând la corecţii mai mici pe măsură ce estimarea $\widehat{\theta}$ se îmbunătăţeşte.

Conţinut

Reamintim: Regresie liniară CMMP

Model:

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

Setul de date până la k duce la un sistem liniar de ecuaţii:

$$\begin{aligned} y(1) &= \varphi^\top(1)\theta \\ y(2) &= \varphi^\top(2)\theta \\ &\cdots \\ y(k) &= \varphi^\top(k)\theta \end{aligned}$$

După algebră şi analiză liniară, soluţia CMMP se poate scrie:

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

CMMP: Formula recursivă

Cu notaţia $P(k) = \sum_{j=1}^{k} \varphi(j)\varphi^\top(j)$:

$$\begin{aligned}
\widehat{\theta}(k) &= P^{-1}(k) \left[\sum_{j=1}^{k} \varphi(j) y(j) \right] \\
&= P^{-1}(k) \left[\sum_{j=1}^{k-1} \varphi(j) y(j) + \varphi(k) y(k) \right] \\
&= P^{-1}(k) \left[P(k-1)\widehat{\theta}(k-1) + \varphi(k) y(k) \right] \\
&= P^{-1}(k) \left[[P(k) - \varphi(k)\varphi^\top(k)]\widehat{\theta}(k-1) + \varphi(k) y(k) \right] \\
&= \widehat{\theta}(k-1) + P^{-1}(k) \left[-\varphi(k)\varphi^\top(k)\widehat{\theta}(k-1) + \varphi(k) y(k) \right] \\
&= \widehat{\theta}(k-1) + P^{-1}(k)\varphi(k) \left[y(k) - \varphi^\top(k)\widehat{\theta}(k-1) \right]
\end{aligned}$$

CMMP: Proprietăţi

$$\boxed{\widehat{\theta}(k) = \widehat{\theta}(k-1) + P^{-1}(k)\varphi(k)\left[y(k) - \varphi^\top(k)\widehat{\theta}(k-1)\right]}$$

$$\widehat{\theta}(k) = \widehat{\theta}(k-1) + W(k)\varepsilon(k)$$

- Formulă recursivă: noua estimare $\widehat{\theta}(k)$ calculată pornind de la estimarea anterioară $\widehat{\theta}(k-1)$ şi noua măsurătoare $y(k)$.
- $\left[y(k) - \varphi^\top(k)\widehat{\theta}(k-1)\right]$ este o eroare de predicţie $\varepsilon(k)$, fiindcă $\varphi^\top(k)\widehat{\theta}(k-1) = \widehat{y}(k)$ este predicţia cu un pas înainte folosind vectorul anterior de parametri.
- $W(k) = P^{-1}(k)\varphi(k)$ este un vector de ponderi: elementele lui P cresc pe măsură ce pasul k creşte, aşadar $W(k)$ descreşte.

Calculul recursiv al matricii inverse

Formula anterioară necesită inversarea unei matrici, $P^{-1}(k)$, o operaţie costisitoare.

Matricea P se scrie uşor recursiv: $P(k) = P(k-1) + \varphi(k)\varphi^\top(k)$, dar această formă nu ne ajută direct; matricea încă trebuie inversată.

Formula Sherman-Morrison actualizează recursiv direct inversa P^{-1}:

$$P^{-1}(k) = P^{-1}(k-1) - \frac{P^{-1}(k-1)\varphi(k)\varphi^\top(k)P^{-1}(k-1)}{1 + \varphi^\top(k)P^{-1}(k-1)\varphi(k)}$$

Exerciţiu: Demonstraţi formula Sherman-Morrison! (algebră liniară)

Algoritm

Metoda CMMP recursivă

iniţializează $\widehat{\theta}(0)$, $P^{-1}(0)$
loop la fiecare pas $k = 1, 2, \dots$
  măsoară $y(k)$, formează vectorul de regresori $\varphi(k)$
  calculează eroarea de predicţie $\varepsilon(k) = y(k) - \varphi^\top(k)\widehat{\theta}(k-1)$
  actualizează $P^{-1}(k) = P^{-1}(k-1) - \frac{P^{-1}(k-1)\varphi(k)\varphi^\top(k)P^{-1}(k-1)}{1+\varphi^\top(k)P^{-1}(k-1)\varphi(k)}$
  calculează ponderile $W(k) = P^{-1}(k)\varphi(k)$
  actualizează parametrii $\widehat{\theta}(k) = \widehat{\theta}(k-1) + W(k)\varepsilon(k)$
end loop

Iniţializare

Metoda necesită vectorul iniţial de parametri $\widehat{\theta}(0)$, şi inversa iniţială $P^{-1}(0)$.

Valori tipice fără cunoştinţe a priori:

- $\widehat{\theta}(0) = [0, \dots, 0]^\top$ - un vector de n zerouri.
- $P^{-1}(0) = \frac{1}{\delta} I$, cu δ un număr mic, de ex. 10^{-3}.
  Valoarea iniţială corespunzătoare a lui P este $P(0) = \delta I$.

Intuiţie: P mic corespunde unui grad de încredere mic în parametrii iniţiali, ducând la valori mari în $P^{-1}(0)$. Aşadar, ponderile $W(k)$ sunt mari iniţial, şi se efectuează corecţii mari ale parametrilor $\widehat{\theta}$.

Dacă sunt disponibile valori a priori pentru $\widehat{\theta}$, ele se pot folosi la iniţializarea $\widehat{\theta}(0)$, şi δ poate fi mărit corespunzător. Interpretarea este un grad de încredere mai mare în estimarea iniţială, ducând la corecţii mai mici.

Verificare: Recuperarea cazului scalar

$$y(k) = b + e(k) = \varphi(k)\theta + e(k), \text{ unde } \varphi(k) = 1, \theta = b$$

Luăm $\widehat{\theta}(0) = 0$, $P^{-1}(0) \to \infty$. Folosind Sherman-Morrison, avem:

$$P^{-1}(k) = P^{-1}(k-1) - \frac{(P^{-1}(k-1))^2}{1 + P^{-1}(k-1)} = \frac{P^{-1}(k-1)}{1 + P^{-1}(k-1)}$$

$$P^{-1}(1) = 1$$

$$P^{-1}(2) = \frac{1}{2}$$

$$\dots$$

$$P^{-1}(k) = \frac{1}{k}$$

Avem $\varepsilon(k) = y(k) - \widehat{\theta}(k-1)$ şi $W(k) = P^{-1}(k) = \frac{1}{k}$, ducând la:

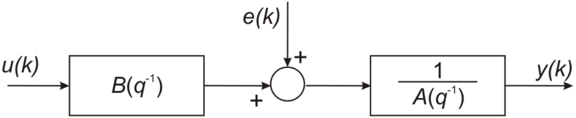
$$\widehat{\theta}(k) = \widehat{\theta}(k-1) + W(k)\varepsilon(k) = \widehat{\theta}(k-1) + \frac{1}{k}[y(k) - \widehat{\theta}(k-1)]$$

$\Rightarrow$ a fost recuperată formula din cazul scalar.

Conţinut

Reamintim: Model ARX

$$A(q^{-1})y(k) = B(q^{-1})u(k) + e(k)$$
$$(1+a_1 q^{-1} + \cdots + a_{na} q^{-na})y(k) = (b_1 q^{-1} + \cdots + b_{nb} q^{-nb})u(k) + e(k)$$



În formă explicită:

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

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

Reamintim: Forma ARX pentru regresie liniară

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

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

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

ARX recursiv

Instanţiem metoda CMMP recursivă din cazul general.

ARX recursiv

iniţializează $\widehat{\theta}(0)$, $P^{-1}(0)$
loop la fiecare pas $k = 1, 2, \ldots$
  măsoară $u(k)$, $y(k)$
  formează vectorul de regresori
  $\varphi(k) = [-y(k-1), \cdots, -y(k-na), u(k-1), \cdots, u(k-nb)]^\top$
  calculează eroarea de predicţie $\varepsilon(k) = y(k) - \varphi^\top(k)\widehat{\theta}(k-1)$
  actualizează $P^{-1}(k) = P^{-1}(k-1) - \frac{P^{-1}(k-1)\varphi(k)\varphi^\top(k)P^{-1}(k-1)}{1+\varphi^\top(k)P^{-1}(k-1)\varphi(k)}$
  calculează ponderile $W(k) = P^{-1}(k)\varphi(k)$
  actualizează parametrii $\widehat{\theta}(k) = \widehat{\theta}(k-1) + W(k)\varepsilon(k)$
end loop

Observaţie: Ieşirile mai vechi de *na* paşi, şi intrările mai vechi de *nb* paşi, nu sunt folosite şi pot fi “uitate”, reducând necesarul de memorie al algoritmului.

Garanţie de performanţă: Idee

Cu $\widehat{\theta}(0) = 0$, $P^{-1}(0) = \frac{1}{\delta} I$ şi $\delta \to 0$, ARX recursiv la pasul k este echivalent cu ARX offline pe setul de date $u(1), y(1), \dots, u(k), y(k)$.

$\Rightarrow$ aceeaşi garanţie ca ARX offline:

Teoremă

Date fiind ipotezele potrivite (incluzând existenţa parametrilor corecţi θ_0), identificarea ARX este consistentă: parametrii estimaţi $\widehat{\theta}$ converg la cei corecţi θ_0 când $k \to \infty$.

Iniţializarea lui P în acest fel asigură egalitatea inverselor cu cele calculate în cazul offline.

Conţinut

Sistem

Pentru a ilustra metoda ARX recursivă, luăm un sistem cunoscut:

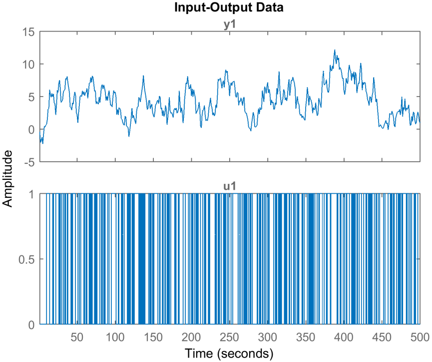
$$y(k) + ay(k-1) = bu(k-1) + e(k), \quad a = -0.9, b = 1$$

(Söderström & Stoica)

```
system = idpoly([1 -0.9], [0 1]);
```

Datele de identificare sunt obţinute în simulare:

```
sim(system, u, 'noise');
```



ARX recursiv

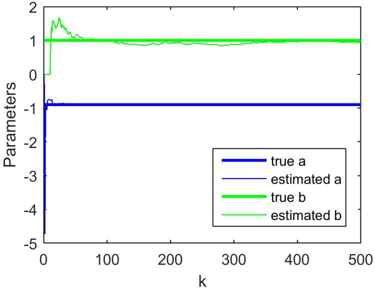
```
model = rarx(id, [na, nb, nk], 'ff', 1, th0, Pinv0);
```

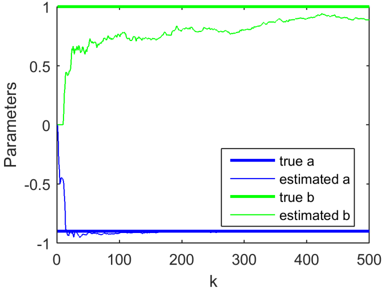
Argumente:

1. Datele de identificare.
2. Vector conţinând ordinele polinoamelor A şi B şi întârzierea nk.
3. `'ff', 1` selectează varianta de algoritm prezentată în curs.
4. `th0` este vectorul iniţial de parametri.
5. `Pinv0` este inversa iniţială $P^{-1}(0)$.

Rezultate

$na = nb = nk = 1$
$P^{-1}(0) = \frac{1}{\delta} I$, stânga: $\delta = 10^{-3}$, dreapta: $\delta = 100$.
În toate experimentele, $\theta_0 = [0, 0]^\top$





Concluzii:

- Converge la valorile corecte ale parametrilor...
- ...mai încet când δ este mai mare (în acest caz, gradul mare de încredere este greşit alocat unei soluţii iniţiale incorecte)

Perturbaţie colorată

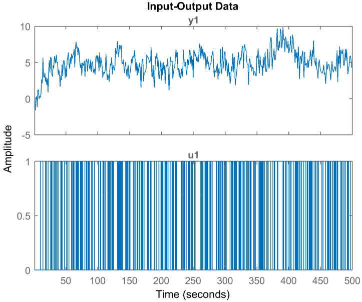
Luăm acum un sistem în forma *eroare de ieşire* (care nu satisface aşadar structura ARX cu zgomot alb):

$$y(k) = \frac{bq^{-1}}{1 + fq^{-1}} u(k) + e(k), \quad f = -0.9, b = 1$$

(Söderström & Stoica)

```
system = idpoly([], [0 1], [], [], [1 -.9]);
```

Date de identificare:

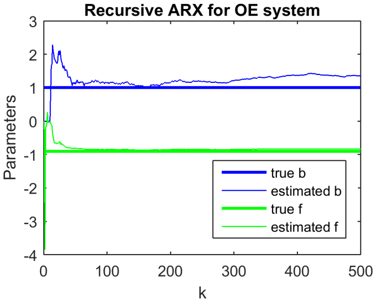


Rezultate

$nf = nb = nk = 1$, deci încercăm un ARX de ordinul 1:

$$y(k) + fy(k-1) = bu(k-1) + e(k)$$

Luăm şi $P^{-1}(0) = \frac{1}{\delta} I$, $\delta = 10^{-3}$.



Concluzie: Nu converge la valorile corecte! – datorită perturbaţiei colorate (sistemul nu este în clasa de modele considerată)

Alţi algoritmi de tip MEP recursivi

În Matlab sunt disponibile şi alte variante recursive de MEP, de ex.:

- ARMAX, `rarmax`
- eroare de ieşire, `roe`
- modele în forma generală de la minimizarea erorii de predicţie, `rpem`

Conţinut

Reamintim: Metoda variabilelor instrumentale

Metoda VI găseşte vectorul de parametri:

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

care este soluţia sistemului de ecuaţii:

$$\left[\frac{1}{N} \sum_{k=1}^{N} Z(k)[\varphi^\top(k)\theta - y(k)] \right] = 0$$

Vectorul de regresori:
$\varphi(k) = [-y(k-1), \cdots, -y(k-na), u(k-1), \cdots, u(k-nb)]^\top$,
iar vectorul VI este de obicei:
$Z(k) = [-x(k-1), \cdots, -x(k-na), u(k-1), \cdots, u(k-nb)]^\top$.

Cum vectorul VI $Z(k)$ este necorelat cu perturbaţia, metoda VI funcţionează pentru perturbaţii colorate.

Formulare recursivă (1)

Putem elimina împărţirea cu numărul N de puncte din setul de date, obţinând:

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

Împărţirea era necesară pentru obţinerea mediilor în VI offline, fiindcă sumele ar fi fost foarte mari pentru multe date, ducând la probleme numerice. Metoda VI recursivă va aduna termenii unul câte unul şi va fi aşadar mai stabilă d.p.d.v. numeric.

Rescriem ecuaţia pentru cazul recursiv:

$$\widehat{\theta}(k) = P^{-1}(k) \left[\sum_{j=1}^{k} Z(j)y(j)\right]$$

unde $P(k) = \sum_{j=1}^{k} Z(j)\varphi^\top(j)$.

Formulare recursivă (2)

Cu această definiţie P, formula recursivă se găseşte ca şi pentru CMMP:

$$\begin{aligned}
\widehat{\theta}(k) &= P^{-1}(k)\left[\sum_{j=1}^{k} Z(j)y(j)\right] \\
&= P^{-1}(k)\left[\sum_{j=1}^{k-1} Z(j)y(j) + Z(k)y(k)\right] \\
&= P^{-1}(k)\left[P(k-1)\widehat{\theta}(k-1) + Z(k)y(k)\right] \\
&= P^{-1}(k)\left[[P(k) - Z(k)\varphi^\top(k)]\widehat{\theta}(k-1) + Z(k)y(k)\right] \\
&= \widehat{\theta}(k-1) + P^{-1}(k)\left[-Z(k)\varphi^\top(k)\widehat{\theta}(k-1) + Z(k)y(k)\right] \\
&= \widehat{\theta}(k-1) + P^{-1}(k)Z(k)\left[y(k) - \varphi^\top(k)\widehat{\theta}(k-1)\right]
\end{aligned}$$

Formulare recursivă (3)

Formula finală:

$$\boxed{\widehat{\theta}(k) = \widehat{\theta}(k-1) + P^{-1}(k)Z(k)\left[y(k) - \varphi^\top(k)\widehat{\theta}(k-1)\right]}$$

$$\widehat{\theta}(k) = \widehat{\theta}(k-1) + W(k)\varepsilon(k)$$

cu Sherman-Morrison pentru actualizarea inversei:

$$P^{-1}(k) = P^{-1}(k-1) - \frac{P^{-1}(k-1)Z(k)\varphi^\top(k)P^{-1}(k-1)}{1 + \varphi^\top(k)P^{-1}(k-1)Z(k)}$$

Metoda VI recursivă: Algoritm

VI recursiv

iniţializează $\widehat{\theta}(0)$, $P^{-1}(0)$
loop la fiecare pas $k = 1, 2, \dots$
măsoară $y(k)$, formează $\varphi(k)$ şi vectorul VI
$Z(k) = [-x(k-1), \cdots, -x(k-na), u(k-1), \cdots, u(k-nb)]^\top$
calculează eroarea de predicţie $\varepsilon(k) = y(k) - \varphi^\top(k)\widehat{\theta}(k-1)$
actualizează $P^{-1}(k) = P^{-1}(k-1) - \frac{P^{-1}(k-1)Z(k)\varphi^\top(k)P^{-1}(k-1)}{1+\varphi^\top(k)P^{-1}(k-1)Z(k)}$
calculează ponderile $W(k) = P^{-1}(k)Z(k)$
actualizează parametrii $\widehat{\theta}(k) = \widehat{\theta}(k-1) + W(k)\varepsilon(k)$
end loop

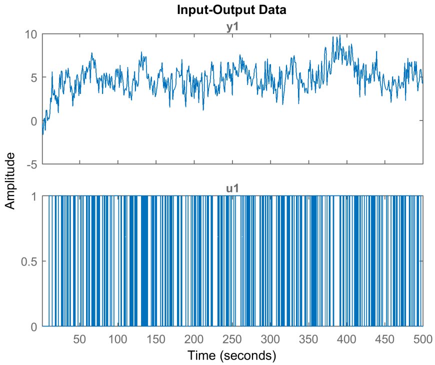
Table of contents

Sistem OE

Considerăm acelaşi sistem OE pe care l-am folosit în exemplul ARX:

$$y(k) = \frac{bq^{-1}}{1 + fq^{-1}} u(k) + e(k), \quad f = -0.9, b = 1$$

cu acelaşi set de date:

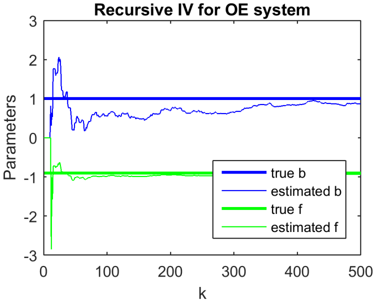


Rezultat cu metoda VI recursivă

Folosim modelul VI cu perturbaţie colorată $v(k)$:

$$y(k) + fy(k-1) = bu(k-1) + v(k)$$

Vector de VI: $Z(k) = [u(k-2), u(k-1)]$.
Ca mai sus, $P^{-1}(0) = \frac{1}{\delta} I$, $\delta = 10^{-3}$.



Concluzie: Mai bun decât ARX recursiv