

Proiectarea optima a arhitecturii RN pentru prognoza si identificare statica

Prof.dr.ing. M. TERTISCO, conf.dr.ing. Al. TERTISCO, ing. C. BAROIANU
Facultatea de Automatica si Calculatoare, Universitatea "Politehnica" Bucuresti

În lucrare se prezinta o noua metoda de proiectare a structurilor modelelor neurale de prognoza. Group Method of Data Handling (GMDH) a fost propusa de academicianul A.G. Ivahnenko de la Institutul de Cibernetica din Kiev în 1968, pentru rezolvarea problemelor de prognoza si obtinerea modelelor de mari dimensiuni, în general de regresie. Metoda are la baza descompunerea modelului problemei în modele simple si reasamblarea acestora, dupa o selectie prealabila. În prezenta lucrare, GMDH a fost preluat si prezentat ca o solutie de proiectare etapizata a structurii RN pentru prognoza. Noua metoda de proiectare propusa este prezentata prin comparatie cu metoda bazata pe teorema aproximarii universale a lui Kolmogorov [1].

Cuvinte cheie: retea neuronală, prognoza, parametru predictor.

Principiul metodei GMDH

Metoda se bazeaza pe faptul ca, pentru obtinerea modelului de prognoza a unui proces aleator y , exista disponibil un sir de valori ale marimii aleatoare y , la momente

de timp succesive si echidistante $t=1<t=2<\dots<t=N$. Se considera ca setul de date $y_1 \dots y_N$ provine dintr-un proces aleator autoregresiv neliniar, descris de:

$$y_t = F(\underbrace{y_{t-1}, y_{t-2}, \dots, y_{t-n}}_{\text{parte prognozabila}}) + \underbrace{e_t}_{\text{parte neprognozabila}} \quad (1)$$

în care e_t este zgomot alb ($e_t=z_a$) nepredictibil, iar n este ordinul autoregresiei neliniare, care este necunoscut. Considerând $n=1$, înseamna ca valoarea la momentul $t=N+1$ ar putea fi prognozata cu modelul:

$$\hat{y}_{N+1} = F(y_N) \quad (2)$$

Banalizând problema, am putea considera ca prognoza pentru anul $N+1$ este o functie

liniara, cu parametri necunoscuti:

$$\hat{y}_{N+1} = \hat{F} = a + b \cdot y_N \quad (3)$$

Modelul (3) pentru $n=1$ poate fi implementat sub forma unui neuron de tip perceptron (figura 1), cu o singura intrare $u_1=y_t$ si pragul b_1 , unde $f(x) = (1 + e^{-x})^{-1}$ reprezinta functia de activare a neuronului, iar y_{t+1} prognoza.

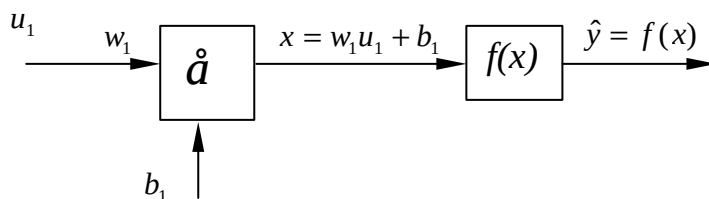


Fig.1. - Structura neuronului folosit ca model de prognoza

În acest caz, la intrarea u_1 se aplica valoarea cunoscuta a procesului de prognozat, la momentul de timp t , iar la ieseire se obtine predictia viitoarei valori, la momentul de timp $t+1$. Aceasta, desigur, numai dupa ce a fost aleasa functia de activare si dupa ce a fost efectuata instruirea acestui perceptron sa prognozeze procesul, din care dispunem de secventa de valori, care formeaza setul de instruire:

$$S = \{y_1 \ y_2 \ \dots \ y_N\} \quad (4)$$

Instruirea perceptronului se face în doua etape: *etapa de învățare (EI)*; *etapa de verificare (EV)*. Ambele etape folosesc datele de instruire (4), în acest scop, setul de date (4) este despartit în doua submultimi aproximativ egale S_i si S_v ,

din punct de vedere al cardinalitatii (numarului de elemente):

$$S = \{S_i, S_v\} \quad (5)$$

Antrenarea neuronului consta în corectarea ponderii w si a pragului b , printr-o metoda de gradient pentru minimizarea erorii patratic, la fiecare pas $k=1 \dots N-1$.

Metoda GMDH este utila pentru cazurile în care secventa de date disponibile apartine unui proces stohastic neliniar, de forma $y_t = F(y_{t-1}, y_{t-2}, \dots, y_{t-n}) + e_t$,

pentru care ordinul modelului autoregresiv este $n > 3$. În acest caz, conform teoremei Kolmogorov, rețeaua contine stratul inactiv cu n noduri de intrare, un strat ascuns cu $2n+1$ neuroni activi în nodurile total interconectate cu cele de intrare si un neuron în stratul de ieseire, ca în figura 2.

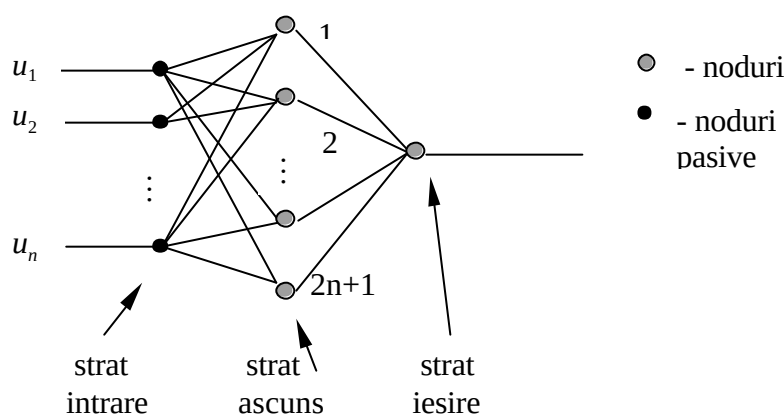


Fig. 2. - Structura RN în conformitate cu teorema Kolmogorov

Teorema Kolmogorov fiind o teorema numai de suficiență, este posibilă găsirea unei rețele neuronale cu structură optimă, cu un număr mult mai mic de conexiuni și de neuroni în stratul ascuns, care să asigure o eroare de aproximare suficient de mică. Prin GMDH se generează o clasă de mode-

le parțiale simple, modelabile printr-un singur neuron.

Vom prezenta pentru cazul $n = 4$, modul în care se construiește primul strat ascuns. Pot fi construite în total $C_4^2 = 6$ modele parțiale (carora le corespund structurile neuronale prezentate în figura 3):

- neuronul 1	$\hat{y}_{N+1} = f(w_1(1) \cdot y_N + w_2(1) \cdot y_{N-1} + b(1))$	
- neuronul 2	$\hat{y}_{N+1} = f(w_1(2) \cdot y_N + w_2(2) \cdot y_{N-2} + b(2))$	
- neuronul 3	$\hat{y}_{N+1} = f(w_1(3) \cdot y_N + w_2(3) \cdot y_{N-3} + b(3))$	
- neuronul 4	$\hat{y}_{N+1} = f(w_1(4) \cdot y_{N-1} + w_2(4) \cdot y_{N-2} + b(4))$	
- neuronul 5	$\hat{y}_{N+1} = f(w_1(5) \cdot y_{N-1} + w_2(5) \cdot y_{N-3} + b(5))$	
- neuronul 6	$\hat{y}_{N+1} = f(w_1(6) \cdot y_{N-2} + w_2(6) \cdot y_{N-3} + b(6))$	(6)

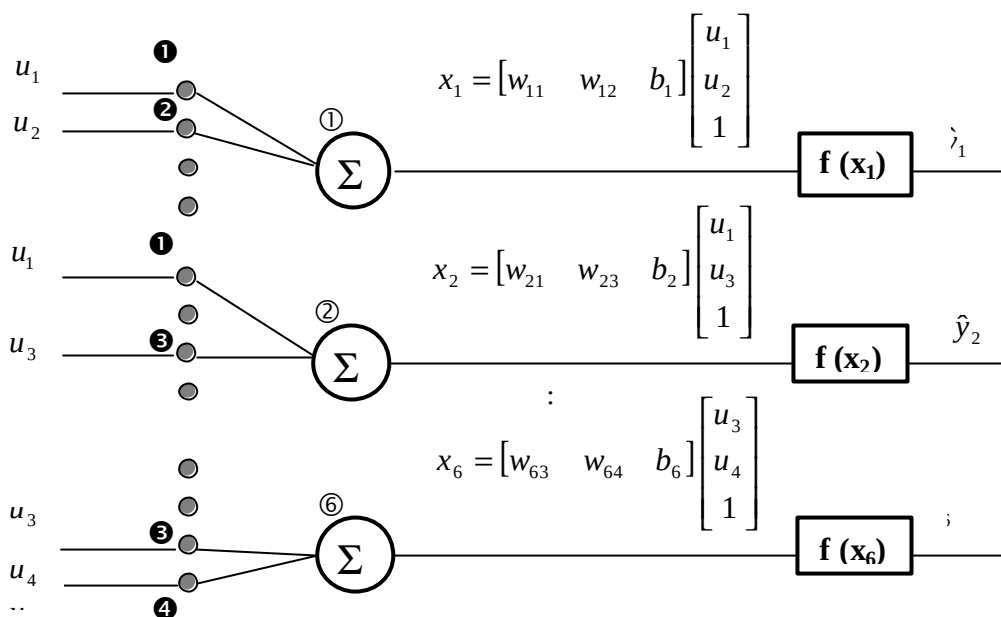


Fig. 3. - Modele pariale pentru prognoza procesului stochastic $y_{N+1} = F(y_N, y_{N-1}, \dots) + e_t$

Deoarece alegerea lui n este arbitrara (ipoteza), este perfect posibil ca numai unul din aceste modele sau combinatii ale unora dintre acestea sa fie acceptabile.

Pregatirea modelelor pariale pentru competitie

Pentru exemplificare, consideram cazul modelului 1 din figura 3. Datele initiale

constituie elemente ale vectorului $y = \{y_1 \dots y_{15}\}$ organizate ca în figura 4. Pentru neuronul 2 din figura 3, prognoza y_{t+1} se face pe baza valorii $y_t = u_1$ si a celei intarziata cu doi pasi $y_{t-2} = u_3$, ca în figura 5.

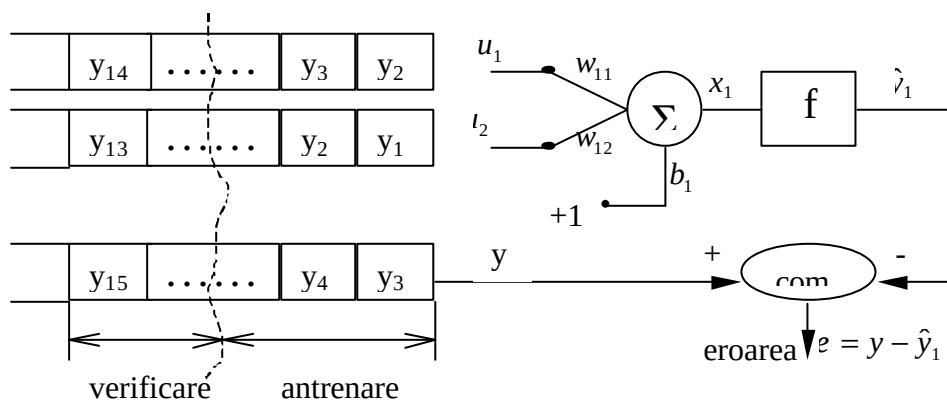


Fig. 4. - Formarea vectorilor de date u_1 si u_2 pentru modelul 1

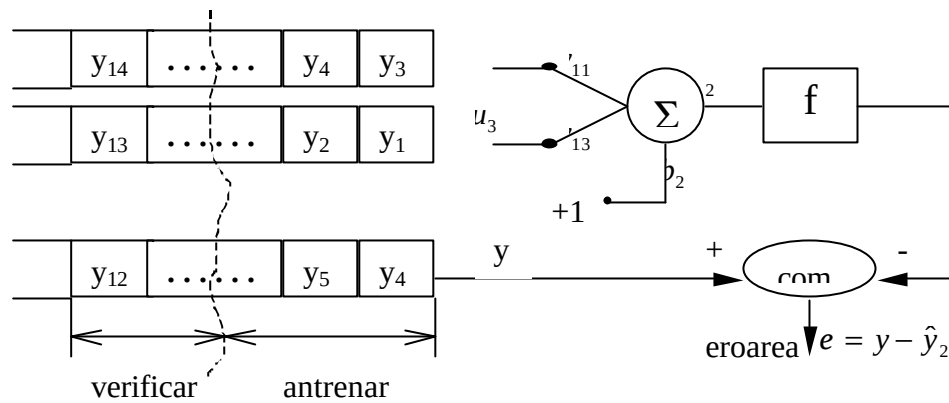


Fig. 5. - Organizarea datelor initiale pentru instruirea modelului 2

Exista, deci 6 astfel de modele parțiale care trebuie pregătite în vederea selectării câtorva, care vor face parte din primul strat ascuns al viitorului predictor neural. În acest scop, prin metoda CMMP se calculează ponderile w și pragul b pentru fiecare model separat, considerând funcția de activare f liniară, adică

$$\hat{y}_i = x_i, \quad i = 1, 2, \dots, 6.$$

De exemplu, pentru modelul 1, în acest caz, datele disponibile din figura 4 permit scrierea următoarelor egalități bazate pe datele din setul de antrenare $y_1 \dots y_8$:

$$U^T U q = U^T Y \quad (7), \text{ în care:}$$

$$U^T = \begin{bmatrix} y_2 & y_3 & y_4 & y_5 & y_6 & y_7 \\ y_1 & y_2 & y_3 & y_4 & y_5 & y_6 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix},$$

$$y^T = [y_1 \quad y_2 \quad y_3 \quad y_4 \quad y_5 \quad y_6 \quad y_7 \quad y_8], \quad q^T = [w_{11} \quad w_{12} \quad b_1].$$

Soluția sistemului (7) furnizează estimări ale parametrilor predictorului din figura 4:

$$\hat{q}^T = [\hat{w}_{11} \quad \hat{w}_{12} \quad \hat{b}_1] \quad (8)$$

În mod similar se calculează parametrii pentru celelalte cinci modele din figura 4. Având valorile parametrilor $\hat{w}_{11}, \hat{w}_{12}, \hat{b}_1$ se pot calcula valorile $\hat{y}$ pentru datele de

intrare din figura 4. Valorile $\hat{y}_1^1, \hat{y}_1^2, \dots, \hat{y}_1^6$ obținute la ieșirea neuronului se pot compara cu valorile dorite din setul de verificare, rezultând vectorul erorilor E_1 și suma erorilor pătratice E_1^2 pentru acest model:

$$E_1 = \begin{bmatrix} y_3 - \hat{y}_1^1 \\ y_4 - \hat{y}_1^2 \\ y_5 - \hat{y}_1^3 \\ y_6 - \hat{y}_1^4 \\ y_7 - \hat{y}_1^5 \\ y_8 - \hat{y}_1^6 \end{bmatrix} = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \\ e_6 \end{bmatrix} \quad \text{și} \quad E_1^2 = E_1^T E = \sum_{k=1}^6 e_k^2 \quad (9)$$

În mod analog se obțin valorile $E_2^2 \dots E_6^2$, pentru celelalte cinci modele considerate la concursul de selecție pentru ocuparea unui

loc în primul strat ascuns al rețelei neuronale. Conform criteriului erorii pătratice, pe primul loc se afla candidatul cu eroarea pătratică cea mai mică, pe ultimul

loc aflându-se modelul cu eroarea patratică cea mai mare și apoi se aleg maxim n modele parțiale, care prezintă erorile cele mai mici.

Să presupunem că neuronul ③ se afla pe poziția 1, neuronul ④ pe poziția 2 iar neuronul ⑤ pe poziția 3. Acești neuroni au ieșirile y_{11} , y_{12} , y_{13} , în care primul indice se referă la primul strat ascuns, iar cel de al doilea la poziția neuronului în strat. S-a construit astfel o parte din rețea, reprezentată în figura 6. Ieșirile acestor neuroni constituie intrări pentru neuronii din stratul următor ascuns.

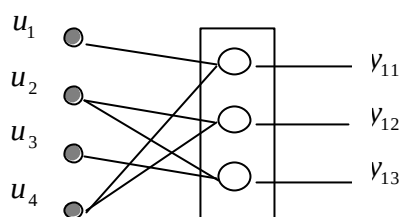


Fig. 6. - Primul strat ascuns al rețelei neuronale

Folosind același procedeu se pot propune similar trei modele, pentru cel de al doilea strat ascuns, prezentate în figura 7.

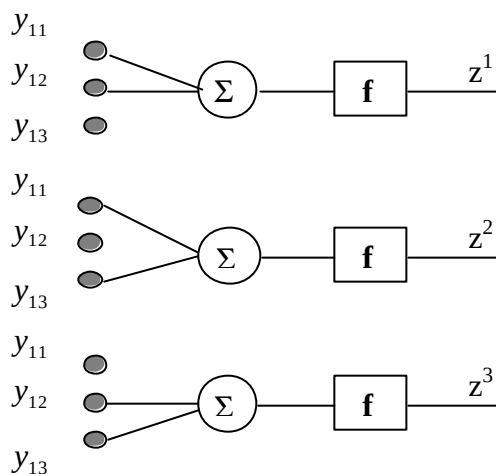


Fig. 7. – Modele parțiale pentru al doilea strat ascuns

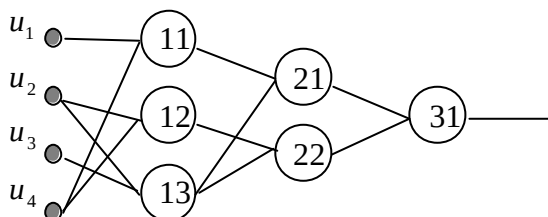


Fig. 8. - Rețea neuronală de predicție cu arhitectura GMDH

Pentru stratul doi au fost selectate numai două modele parțiale, iar ieșirile neuronilor din acest strat, sunt intrări pentru neuronul din stratul de ieșire (al treilea strat). Atunci rețeaua devine cea din figura 8, din care se poate observa clar că aceasta nu este total

conectată ca cea din figura 4. De remarcat, că încercarea de a forma o RN cu trei straturi ascunse a condus la performanțe mai slabe, deoarece eroarea patratică în acest caz s-a dovedit mai mare decât în cazul rețelei cu două straturi.

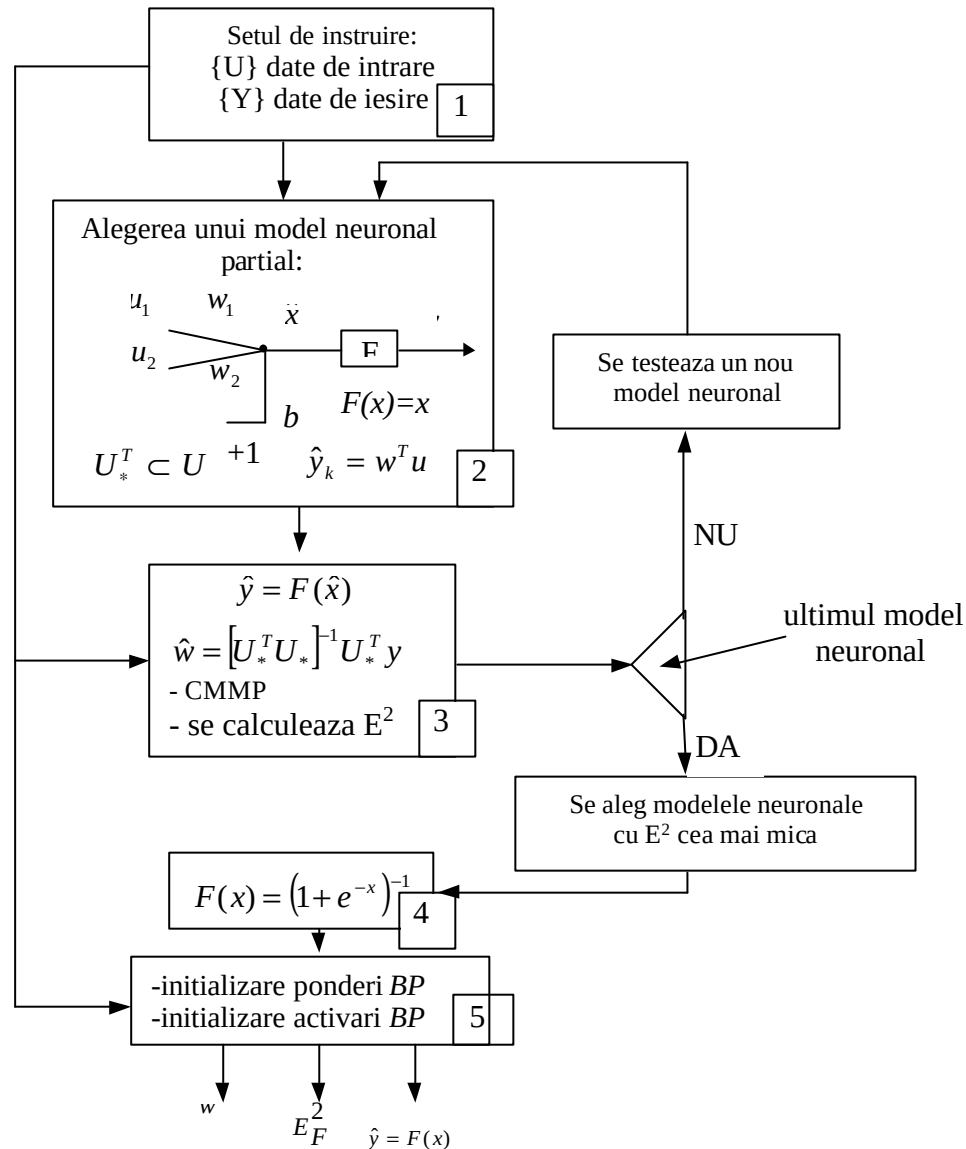


Fig. 9. - Initializarea algoritmului BP pentru GMDH, prin folosirea criteriului CMMP

În figura 9 sunt prezentate schematic etapele de prelucrare a datelor în faza de antrenare, cu algoritmul back propagation (în cadrul unuia din pașii algoritmului GMDH). Initializarile algoritmului back propagation (BP) se fac cu $\hat{w}$, obținut prin CMMP conform (7).

Exemplu

Algoritmul de proiectare a fost aplicat pentru modelarea unui proces [2] caracterizat de setul de date experimentale (10).

k	$u_1(k)$	$u_2(k)$	$u_3(k)$	$u_4(k)$	y_k
1	-1	-1	-1	+1	0.088
2	-1	-1	+1	+1	0.141
3	-1	+1	-1	-1	0.177
4	-1	+1	+1	-1	0.230
5	+1	-1	-1	-1	0.088

(10)

6	+1	-1	+1	-1	0.141
7	+1	+1	-1	+1	0
8	+1	+1	+1	+1	0.053
Σ	0	0	0	0	0.918

A rezultat rețeaua din figura 10.

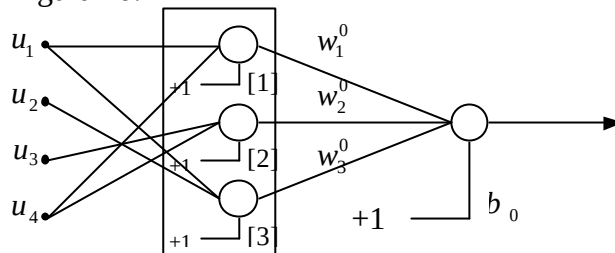


Fig. 10. - RN proiectata cu GMDH pentru datele (10)

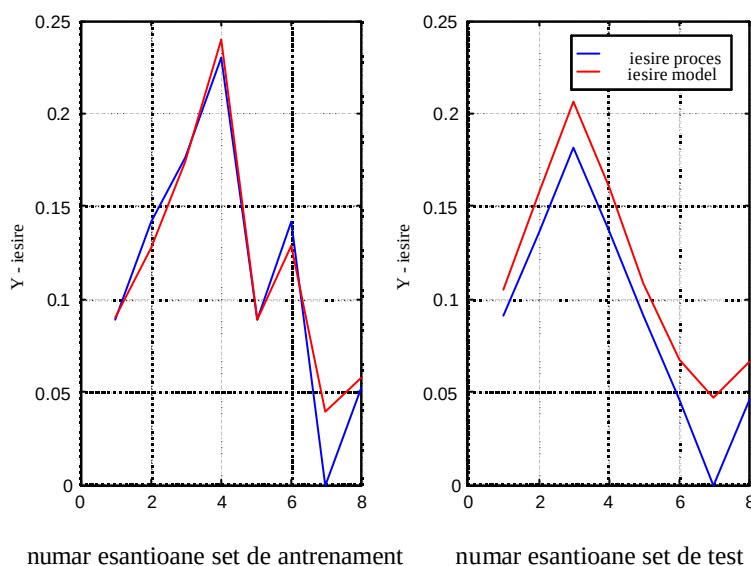


Fig. 12. – Rezultate obtinute pe rețeaua din figura 10

Pentru aceasta rețea s-au obtinut rezultatele din figura 11, eroarea de antrenament fiind de $2.5 \cdot 10^{-4}$ iar cea de test $6.6 \cdot 10^{-4}$. Pentru rețeaua de tip Kolmogorov construita pe baza acelorasi date (10),

rezultatele sunt cele prezentate în figura 12. Se observa ca RN cu structura GMDH (fig.10) este superioara ca precizie.

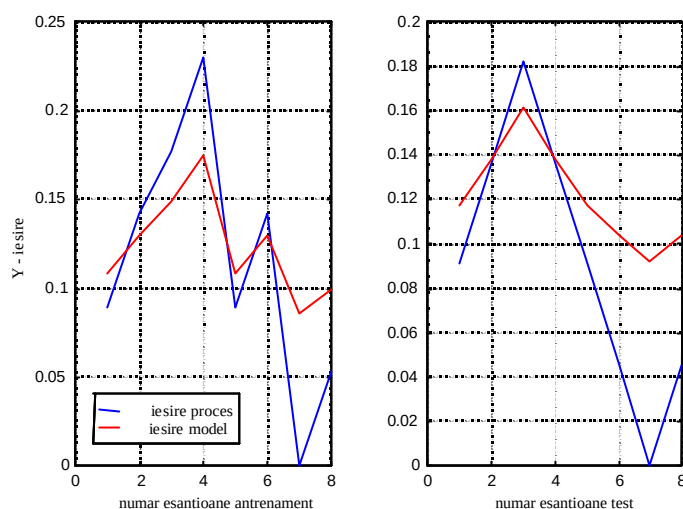


Fig. 12. – Rezultate obtinute pe o retea de tip Kolmogorov (4/9/1)

Concluzii

Structurile RN obtinute prin metoda GMDH, propusa în lucrare, sunt de preferat celor de tip Kolmogorov datorita timpului de instruire mai redus, a performantelor de generalizare superioare si a numarului mai mic de conexiuni.

Majoritatea lucrarilor publicate, indica drept solutie de initializare a algoritmului BP valori aleatoare pentru ponderi sau valori nule. În lucrare s-a folosit o metoda

noua de initializare a algoritmului BP prin ponderi estimate cu metoda CMMP, fapt care a micorat considerabil timpul de instruire a retelei.

Bibliografie

- [1] A. Dumitras - *Proiectarea retelelor neurale artificiale*, Ed. Odeon, Bucuresti, 1997
- [2] M. Tertisco, Gh. Petrache – *Identificarea proceselor*, I.P.B., 1978