



# Politechnika Wrocławska



## Neurosterowniki: Identyfikacja

LABORKA © Piotr Ciskowski



Politechnika Wroclawska

Przykład 1

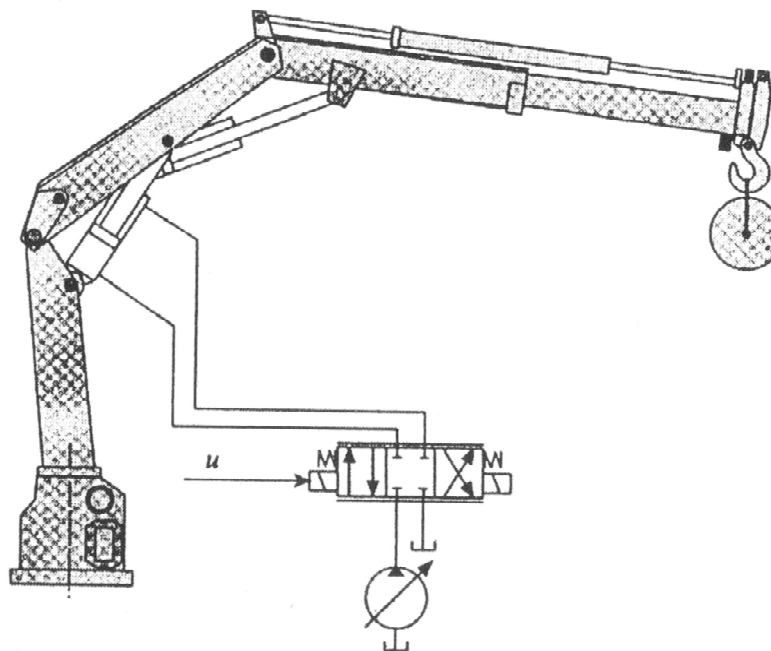
# MODELLING OF A HYDRAULIC ACTUATOR



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- hydraulic actuator - used for controlling the position of a crane arm
  - the boom = lower joint of the arm

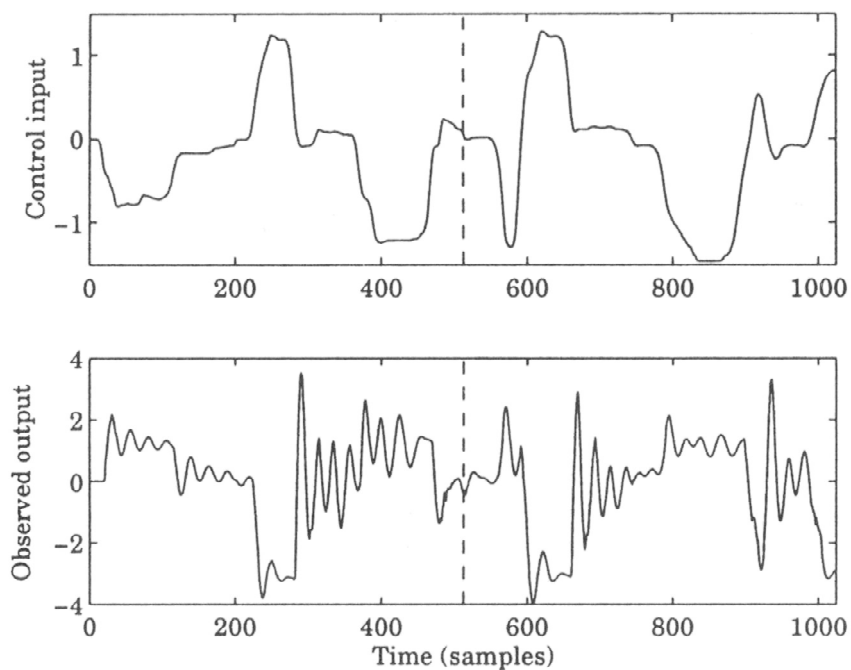




## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- hydraulic actuator - input:  $u$  - valve opening
  - output:  $p$  - oil pressure
  - 512 samples for training
  - 512 samples for validation and model comparison





## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- ściągnij plik actuator.mat
- załaduj go do przestrzeni roboczej MATLABa
- narysuj wykres zmiennych:  $u$  - pobudzenie (otwarcie zaworu - valve opening)  
 $p$  - odpowiedź (ciśnienie oleju - oil pressure)

```
load actuator.mat
```

```
subplot(2,1,1)  
plot(u)
```

```
subplot(2,1,2)  
plot(p)
```



## przykład 1. Modelling of a hydraulic actuator

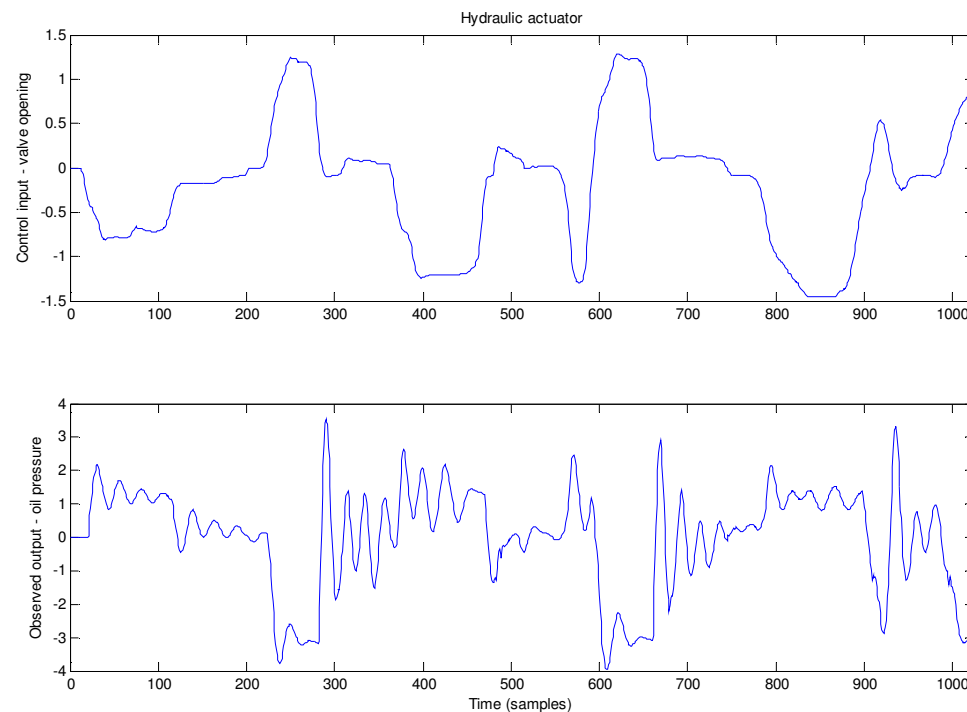
źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- ściągnij plik actuator.mat
- załaduj go do przestrzeni roboczej MATLABa
- narysuj wykres zmiennych:  $u$  - pobudzenie (otwarcie zaworu - valve opening)  
 $p$  - odpowiedź (ciśnienie oleju - oil pressure)

```
load actuator.mat
```

```
subplot(2,1,1)  
plot(u)
```

```
subplot(2,1,2)  
plot(p)
```





Przykład 1

# MODELLING OF A HYDRAULIC ACTUATOR

## MODEL LINIOWY ARX

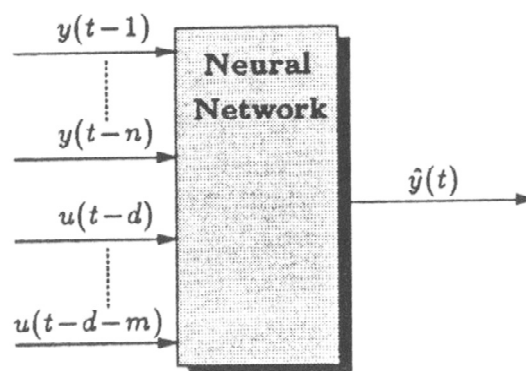


## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1y(t-1) - a_2y(t-2) - a_3y(t-3) + b_1u(t-1) + b_2u(t-2)$$







## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) + b_1 u(t-1) + b_2 u(t-2)$$

- skonstruuj ciąg uczący

```
wejściaSieci = [ 0 0 0 0 0 ;      % 1 kolumna: pobudzenia - opóźnione o 1
                  u 0 p 0 0 ;      % 2 kolumna: pobudzenia - opóźnione o 2
                  0 u 0 p 0 ;      % 3 kolumna: odpowiedzi - opóźnione o 1
                  0 0 0 0 p ] ;    % 4 kolumna: odpowiedzi - opóźnione o 2
                                     % 5 kolumna: odpowiedzi - opóźnione o 3

wyjście = [ p ;                  % żądane wyjście: odpowiedzi nieopóźnione
            0 ;
            0 ;
            0 ] ;
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) + b_1 u(t-1) + b_2 u(t-2)$$

- skonstruuj ciąg uczący

```
wejściaSieci = [ 0  0  0  0  ;           % trzeba je skonstruować „na boku”
                  0  u'  0  0  ;           % a potem transponować
                  0  0  u'  0  ;
                  0  p'  0  0  ;
                  0  0  p'  0  ;
                  0  0  0  p' ]' ;

wyjścieSieci = [ p'  0  0  0  ]' ;

[ wejścia(1:20,:)  wyjście(1:20,:) ]
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) + b_0 u(t) + b_1 u(t-1) + b_2 u(t-2)$$

---

- wytnij zera łączące się z przodu i z tyłu

```
wejściaSieci ( 1025 : 1027 , : ) = [ ] ;
```

```
wyjścieSieci ( 1025 : 1027 , : ) = [ ] ;
```

```
wejściaSieci ( 1 : 3 , : ) = [ ] ;
```

```
wejściaSieci ( 1 : 3 , : ) = [ ] ;
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) + b_0 u(t) + b_1 u(t-1) + b_2 u(t-2)$$

- ostateczne ciągi uczące:

% - dla uczenia wsadowego:

```
PuczWsadowe = wejsciaSieci( 1: 511, :) ' ; % ciąg uczący - pierwsza połowa przykładów  
TuczWsadowe = wyjscieSieci( 1: 511, :) ' ;
```

```
PsprWsadowe = wejsciaSieci(512:1021, :) ' ; % ciąg sprawdzający - druga połowa przykładów  
TsprWsadowe = wyjscieSieci(512:1021, :) ' ;
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) + b_0 u(t) + b_1 u(t-1) + b_2 u(t-2)$$

- ostateczne ciągi uczące:

% - dla uczenia przyrostowego:

```
% ciąg uczący - pierwsza połowa przykładów
PuczPrzyrostowe = num2cell ( wejsciaSieci( 1: 511,:) ' , 1 ) ;
TuczPrzyrostowe = num2cell ( wyjscieSieci( 1: 511,:) ' , 1 ) ;

% ciąg sprawdzający - druga połowa przykładów
PsprPrzyrostowe = num2cell ( wejsciaSieci(512:1021,:) ' , 1 ) ;
TsprPrzyrostowe = num2cell ( wyjscieSieci(512:1021,:) ' , 1 ) ;
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) + b_0 u(t) + b_1 u(t-1) + b_2 u(t-2)$$

---

- utwórz sieć jednowarstwową liniową 5-1 (lub 6-1)

```
siecARXprzed = newlin ( PuczWsadowe , TuczWsadowe ) ; % starszy MATLAB
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) + b_0 u(t) + b_1 u(t-1) + b_2 u(t-2)$$

---

- naucz sieć - wsadowo

```
disp ( 'Uczę sieć ARX - wsadowo...' )  
disp ( ' ' )  
  
siecARXpo = train ( siecARXprzed , Pwsadowe , Twsadowe ) ;
```

- podejrzuj przebieg uczenia  
- *Performance*



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) + b_0 u(t) + b_1 u(t-1) + b_2 u(t-2)$$

---

- naucz sieć - przyrostowo

```
siecARX = siecARXprzed ; % do użytku w pętli

siecARX.adaptParam.passes = 511 ; % aby jedno adapt = pokazanie wszystkich przykładów
liczbaEpok = 10 ;

bladMSE = zeros ( 1 , liczbaEpok ) ; % do wykresu
```





## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) + b_0 u(t) + b_1 u(t-1) + b_2 u(t-2)$$

- naucz sieć - przyrostowo

```
disp ( 'Uczę sieć - przyrostowo...' )
disp ( ' ' )

for i = 1 : liczbaEpok ,

    [ siecARX , yUczARX , eUczARX ] = adapt ( siecARX , PuczPrzyrostowe , ...
                                              TuczPrzyrostowe ) ;

    bladMSE(i) = mse ( eUczARX ) ;

    disp ( sprintf ( 'Krok %5d - błąd: %8.6f' , i , bladMSE(i) ) ) ;

end

siecARXpo = siecARX ;
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) + b_0 u(t) + b_1 u(t-1) + b_2 u(t-2)$$

---

- narysuj wykres przebiegu błędu uczenia w kolejnych epokach

**figure**

```
plot    ( bladMSE          )  
  
title   ( 'Training record' )  
xlabel   ( 'epoch'          )  
ylabel   ( 'MSE error'      )
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) + b_0 u(t) + b_1 u(t-1) + b_2 u(t-2)$$

- sprawdź działanie modelu - metodą *one-step ahead prediction* - na zbiorze uczącym

figure

```
subplot (3,1,1), plot ( TuczWsadowe , 'r' ) , hold on  
plot ( cell2mat(yUczARX) , 'b' )
```

```
title ( 'ARX - One-step ahead prediction - training data' )  
xlabel ( 'Time (samples)' )
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) + b_0 u(t) + b_1 u(t-1) + b_2 u(t-2)$$

- sprawdź działanie modelu - metodą *one-step ahead prediction* - na zbiorze sprawdzającym

```
disp ( ' ' )  
disp ( 'Sprawdzam sieć ARX na zbiorze sprawdzającym...' )  
disp ( ' ' )  
  
ySprARX = sim ( siecARXpo , PsprWsadowe ) ;  
  
subplot (3,1,2), plot ( TsprWsadowe , 'r' ), hold on  
plot ( ySprARX , 'b' )  
  
title ( 'ARX - One-step ahead prediction - validating data' )  
xlabel ( 'Time (samples)' )
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) + b_0 u(t) + b_1 u(t-1) + b_2 u(t-2)$$

- sprawdź działanie modelu - metodą *symulacji* - na zbiorze sprawdzającym

```
disp ( 'Symuluję sieć ARX na zbiorze sprawdzającym...' )  
disp ( ' ' )
```

```
ySimARX = zeros (1,510) ;
```

```
for i = 4 : 510 ,
```

```
    xSim = [ PsprWsadowe( 1:2 , i ) ; ...  
            ySimARX( 1 , i-1 ) ; ...  
            ySimARX( 1 , i-2 ) ; ...  
            ySimARX( 1 , i-3 ) ] ;
```

```
    ySimARX ( 1 , i ) = sim ( siecARXpo , xSim ) ;
```

```
end
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) + b_0 u(t) + b_1 u(t-1) + b_2 u(t-2)$$

---

- sprawdź działanie modelu - metodą *symulacji* - na zbiorze sprawdzającym

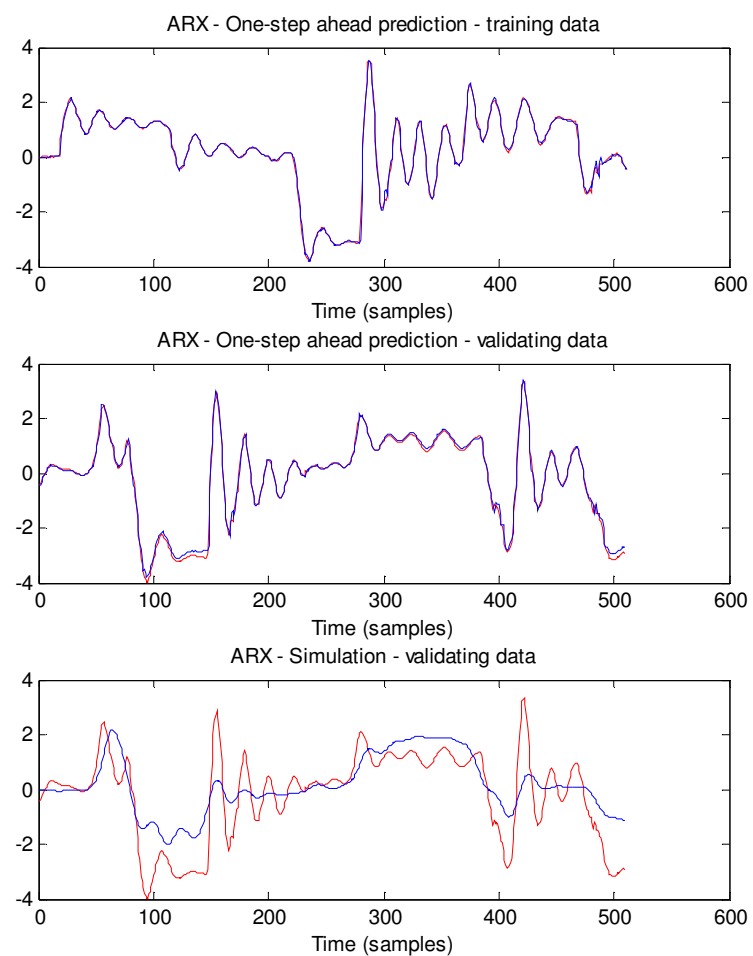
```
subplot (3,1,3), plot ( TsprWsadowe , 'r' ), hold on  
                  plot ( ySimARX      , 'b' )
```

```
title ( 'ARX - Simulation - validating data' )  
xlabel ( 'Time (samples)' )
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211





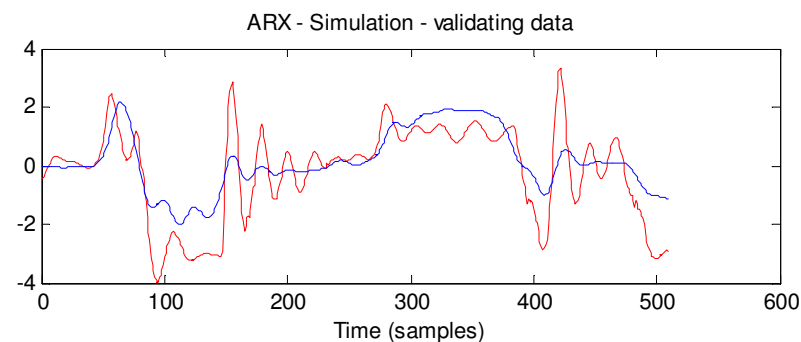
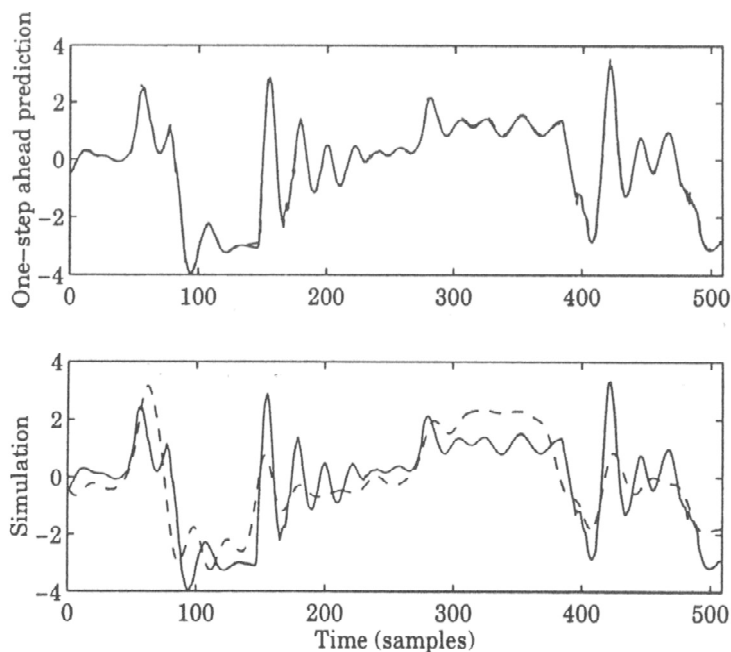
## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu ARX(3,2,1)

$$\hat{y}(t|\theta) = -a_1 y(t-1) - a_2 y(t-2) - a_3 y(t-3) + b_0 u(t) + b_1 u(t-1) + b_2 u(t-2)$$

- porównanie z książką







Przykład 1

# MODELLING OF A HYDRAULIC ACTUATOR

## MODEL NIELINIOWY NNARX



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u_0(t), u(t-1), u(t-2))$$

---

- utwórz sieć

```
liczbaNeuronow = input ( 'Podaj liczbę neuronów w warstwie ukrytej: ' ) ;  
disp ( ' ' ) ;
```

```
siecNNARXprzed = newfit ( PuczWsadowe , TuczWsadowe , liczbaNeuronow ) ;  
% starszy MATLAB
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u_0(t), u(t-1), u(t-2))$$

---

- naucz sieć - wsadowo

```
disp ( 'Uczę sieć NNARX - wsadowo (L-M)...' )  
disp ( ' ' )
```

```
siecNNARXpo = train ( siecNNARXprzed , PuczWsadowe , TuczWsadowe ) ;
```

- obejrzyj przebieg uczenia  
- *Performance*
- obejrzyj efekt uczenia  
- *Regression*



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u_0(t), u(t-1), u(t-2))$$

- sprawdź działanie modelu - metodą *one-step ahead prediction* - na zbiorze uczącym

```
disp ( 'Sprawdzam sieć NNARX na zbiorze uczącym...' )  
disp ( ' ' )
```

```
yUczNNARX = sim ( siecNNARXpo , PuczWsadowe ) ;
```

```
figure
```

```
subplot (3,1,1), plot ( TuczWsadowe , 'r' ), hold on  
plot ( yUczNNARX , 'b' )
```

```
title ( 'NNARX - One-step ahead prediction - training data' )  
xlabel ( 'Time (samples)' )
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u_0(t), u(t-1), u(t-2))$$

- sprawdź działanie modelu - metodą *one-step ahead prediction* - na zbiorze sprawdzającym

```
disp ( 'Sprawdzam sieć NNARX na zbiorze sprawdzającym...' )
disp ( ' ' )

ySprNNARX = sim ( siecNNARXpo , PsprWsadowe ) ;

figure

subplot (3,1,2), plot ( TsprWsadowe , 'r' ), hold on
                plot ( ySprNNARX      , 'b' )

title ( 'NNARX - One-step ahead prediction - validating data' )
xlabel ( 'Time (samples)' )
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u_0(t), u(t-1), u(t-2))$$

- sprawdź działanie modelu - metodą *symulacji* - na zbiorze sprawdzającym

```
disp ( 'Symuluję sieć NNARX na zbiorze sprawdzającym...' )  
disp ( ' ' )
```

```
ySimNNARX = zeros (1,510)
```

```
for i = 4 : 510 ,
```

```
    xSim = [ PsprWsadowe( 1:2 , i ) ; ...  
            ySimNNARX( 1 , i-1 ) ; ...  
            ySimNNARX( 1 , i-2 ) ; ...  
            ySimNNARX( 1 , i-3 ) ] ;
```

```
    ySimNNARX ( 1 , i ) = sim ( siecNNARXpo , xSim ) ;
```

```
end
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u_0(t), u(t-1), u(t-2))$$

---

- sprawdź działanie modelu - metodą *symulacji* - na zbiorze sprawdzającym

```
subplot (3,1,3), plot ( TsprWsadowe , 'r' ), hold on  
plot ( ySimNNARX , 'b' )
```

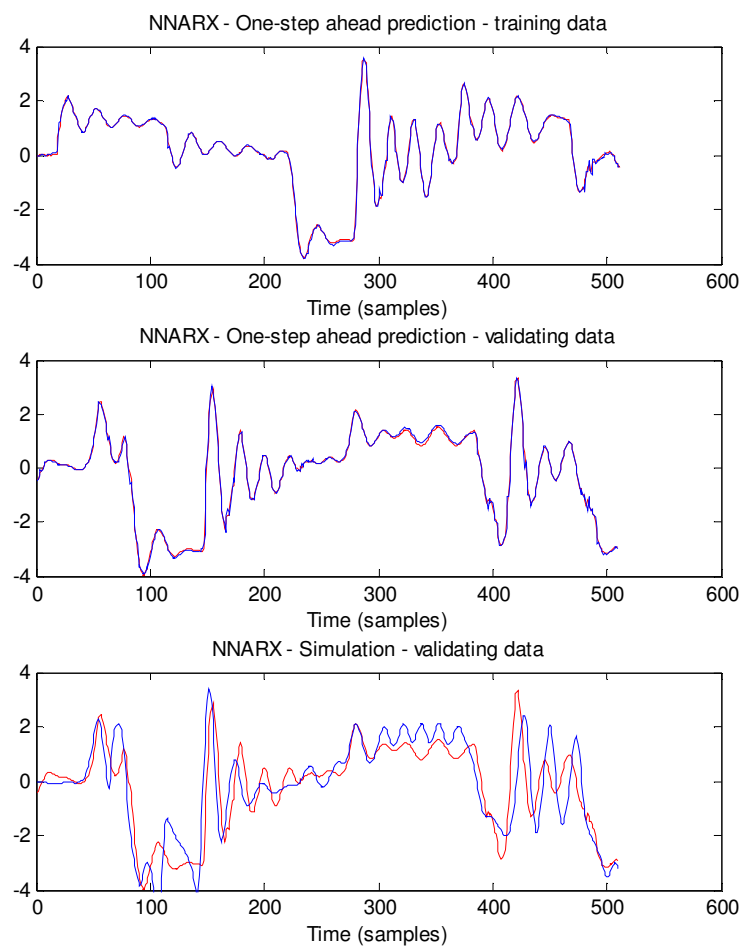
```
title ( 'NNARX - Simulation - validating data' )  
xlabel ( 'Time (samples)' )
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- naucz kilka sieci
- sieć 5-15-1



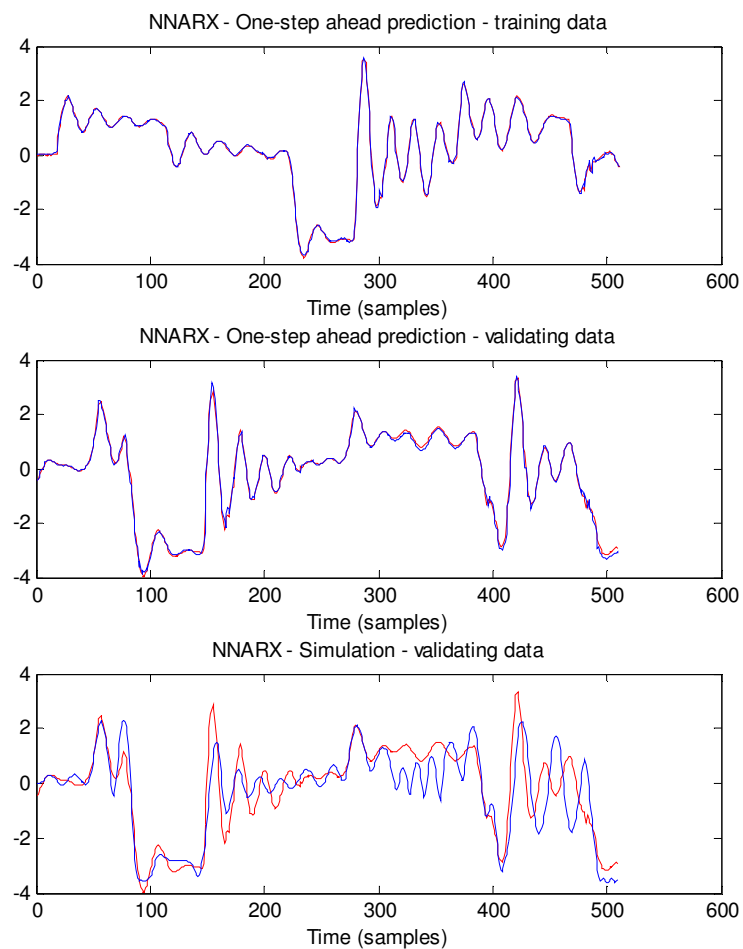




## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- naucz kilka sieci
- sieć 6-10-1

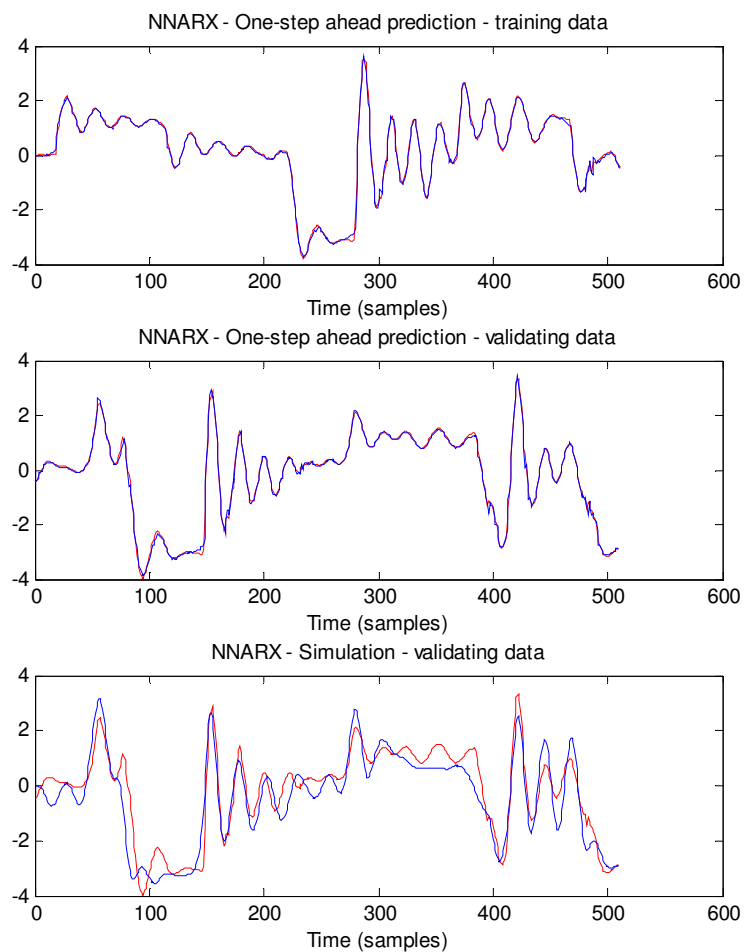




## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- naucz kilka sieci
- sieć 6-5-1

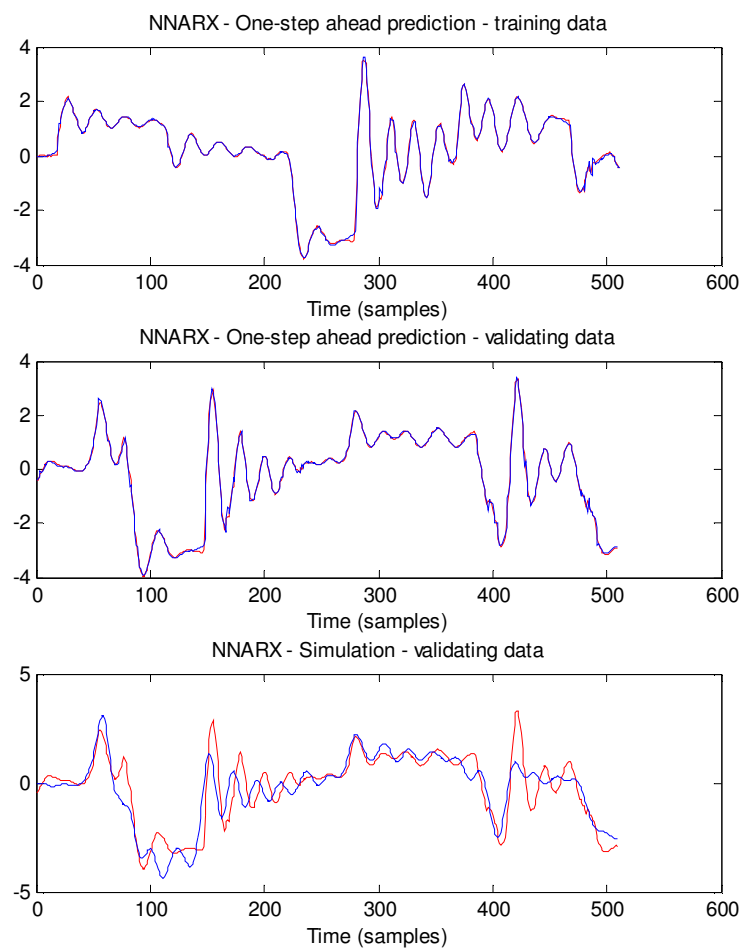




## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- naucz kilka sieci
- sieć 5-3-1





Przykład 1

# **MODELLING OF A HYDRAULIC ACTUATOR**

## **SIEĆ NARX Z TOOLBOXA MATLABA**

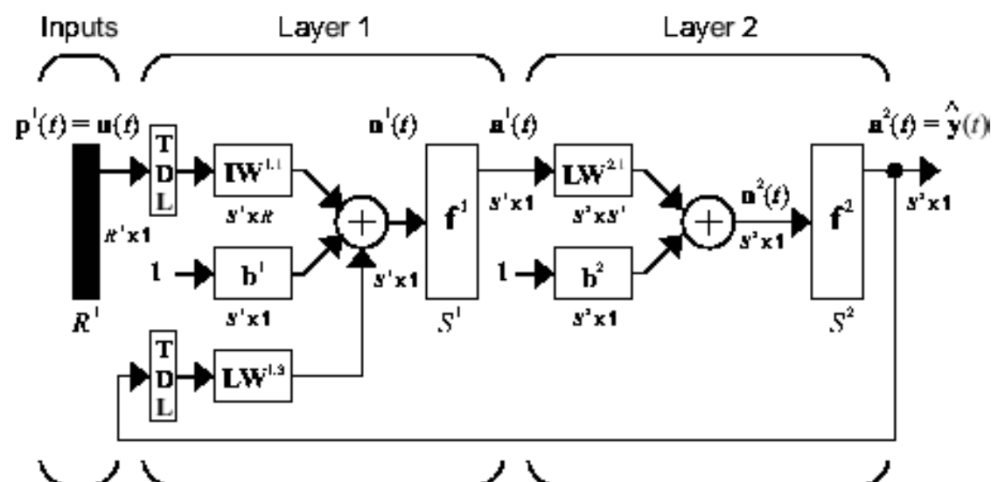


## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u(t-1), u(t-2))$$



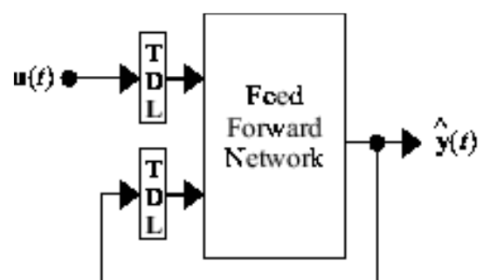


## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

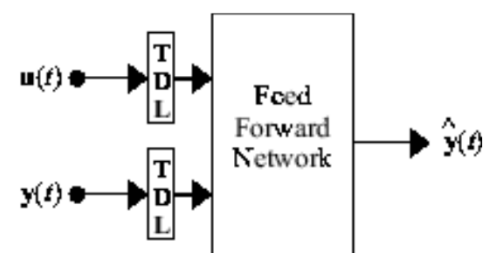
- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u(t-1), u(t-2))$$



Parallel Architecture

używanie



Series-Parallel Architecture

uczenie



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u(t-1), u(t-2))$$

---

- załaduj dane
- stransponuj do wierszy

```
load actuator.mat
```

```
p = p' ; % wyjście: ciśnienie oleju  
u = u' ; % wejście: otwarcie zaworu
```

```
daneUczace = figure ;
```

```
subplot(2,1,1), plot(u), ylabel ( 'u' ) ; title ( 'input: valve opening' ) ;  
subplot(2,1,2), plot(p), ylabel ( 'p' ) ; title ( 'output: oil pressure' ) ;  
xlabel ( 'time' ) ;
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u(t-1), u(t-2))$$

- przeskaluj dane na zakres [-1 1]
- przerób je na sekwencje

```
[ p , ps ] = mapminmax ( p ) ;  
[ u , us ] = mapminmax ( u ) ;
```

```
ps  
us
```

```
pUcz = con2seq ( p(1,1:512)      ) ;  
uUcz = con2seq ( u(1,1:512)      ) ;  
pSpr  = con2seq (   p(1,513:1024) ) ;  
uSpr  = con2seq (   u(1,513:1024) ) ;
```





## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

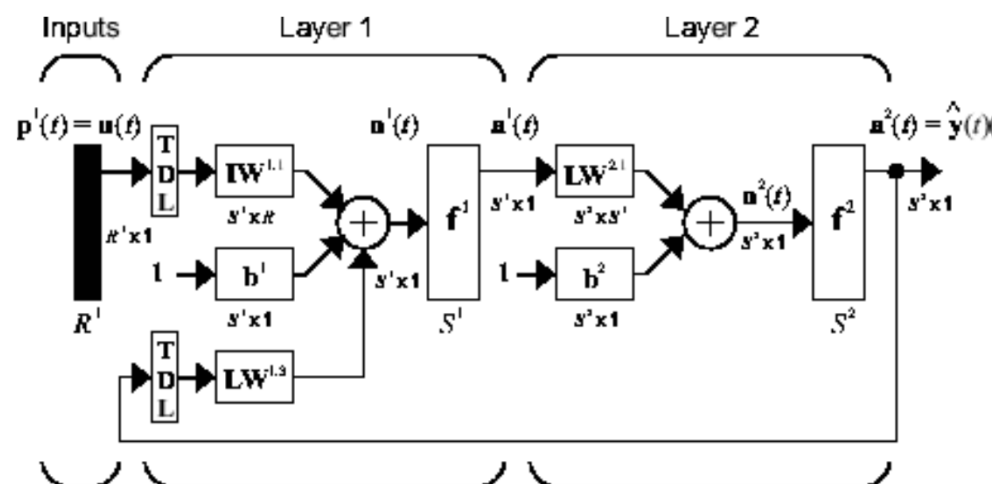
- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u(t-1), u(t-2))$$

- zdefiniuj opóźnienia dla sygnałów wejścia i wyjścia

```
uDelays = [ 1 2 ] ; % opóźnienia wejścia (pobudzenie systemu)
```

```
pDelays = [ 1 2 3 ] ; % opóźnienia wyjścia (odpowiedź systemu)
```





## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u(t-1), u(t-2))$$

---

- wybierz liczbę neuronów i utwórz sieć NARX

```
liczbaNeuronow = input ( 'Podaj liczbę neuronów w warstwie ukrytej sieci: ' ) ;
```

```
siecNARXprzed = narxnet ( uDelays , pDelays , liczbaNeuronow )
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u(t-1), u(t-2))$$

---

- przygotuj dane uczące

```
siecNARXprzed.divideFcn = '';  
siecNARXprzed.trainParam.min_grad = 1e-10 ;
```

```
[ PsekwencjaUcz , PiUcz , AiUcz , tUcz ] = preparets ( siecNARXprzed , uUcz , { } , pUcz ) ;  
[ PsekwencjaSpr , PiSpr , AiSpr , tSpr ] = preparets ( siecNARXprzed , uSpr , { } , pSpr ) ;
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u(t-1), u(t-2))$$

---

- naucz sieć

```
siecNARXpo = train ( siecNARXprzed , PsekwencjaUcz , tUcz , PiUcz ) ;
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u(t-1), u(t-2))$$

---

- *Performance...*
- *Training state...*
- *Error histogram...*
- *Regression...*
- *Time-series response...*
- *Error autocorrelation...*
- *Input-Error cross-correlation...*



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u(t-1), u(t-2))$$

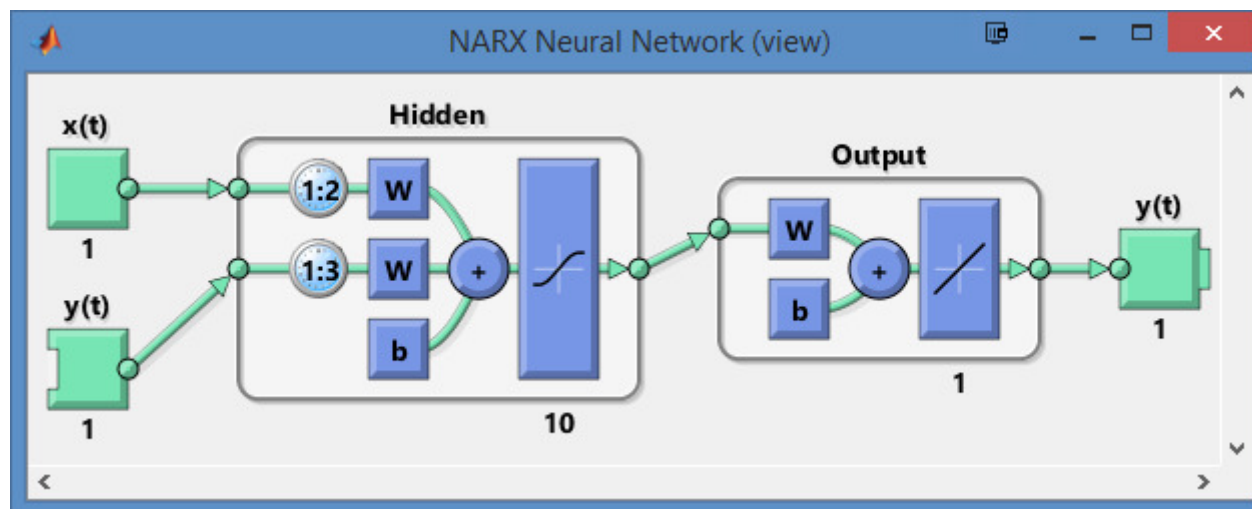
- sprawdź sieć - metodą *one-step-ahead prediction*

```
yUczNARX = sim ( siecNARXpo , PsekwencjaUcz , PiUcz ) ; % na zbiorze uczącym
```

```
ySprNARX = sim ( siecNARXpo , PsekwencjaSpr , PiSpr ) ; % na zbiorze sprawdzającym
```

- pokaż strukturę sieci

```
view ( siecNARXpo ) ;
```





## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u(t-1), u(t-2))$$

- pokaż odpowiedzi sieci - na sprawdzenie metodą *one-step-ahead prediction*

```
odpSieciNARXoneStep = figure ;
```

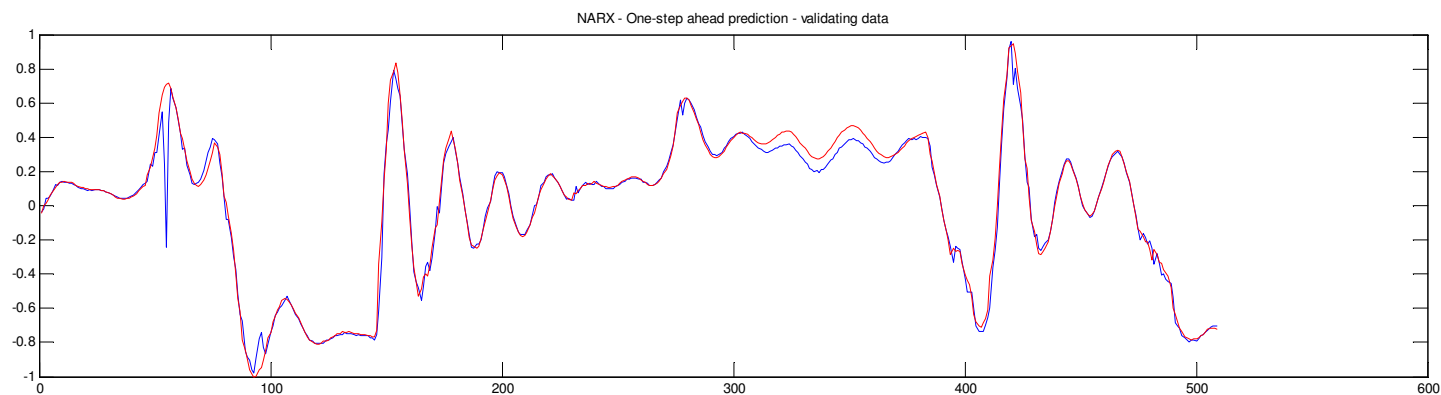
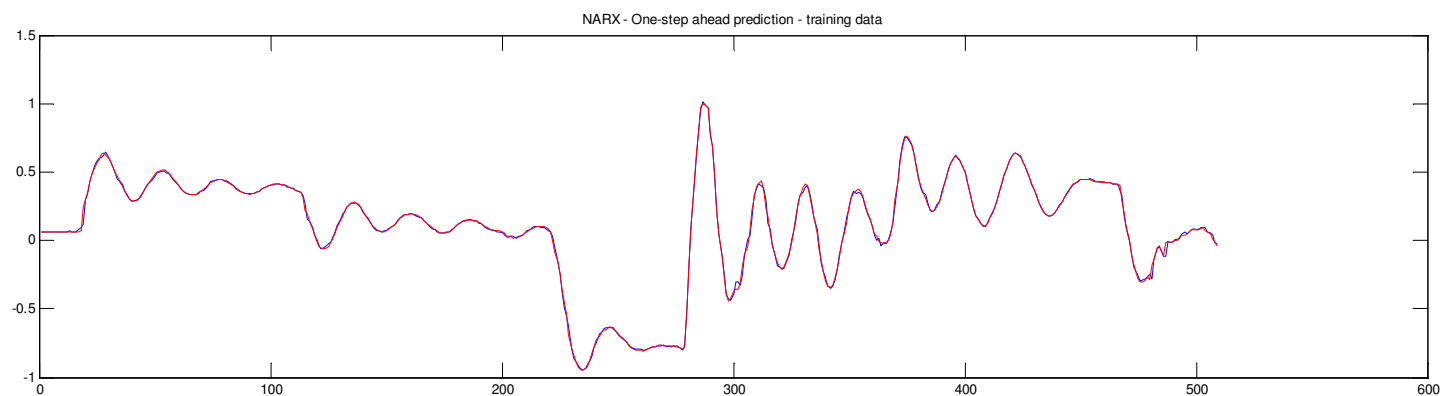
```
subplot ( 2,1,1 ), plot ( cell2mat(yUczNARX) , 'b' ), hold on  
                    plot ( cell2mat(tUcz      ) , 'r' )  
title ( 'NARX - One-step ahead prediction - training data' )
```

```
subplot ( 2,1,2 ), plot ( cell2mat(ySprNARX) , 'b' ), hold on  
                    plot ( cell2mat(tSpr      ) , 'r' )  
title ( 'NARX - One-step ahead prediction - validating data' )
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211







## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

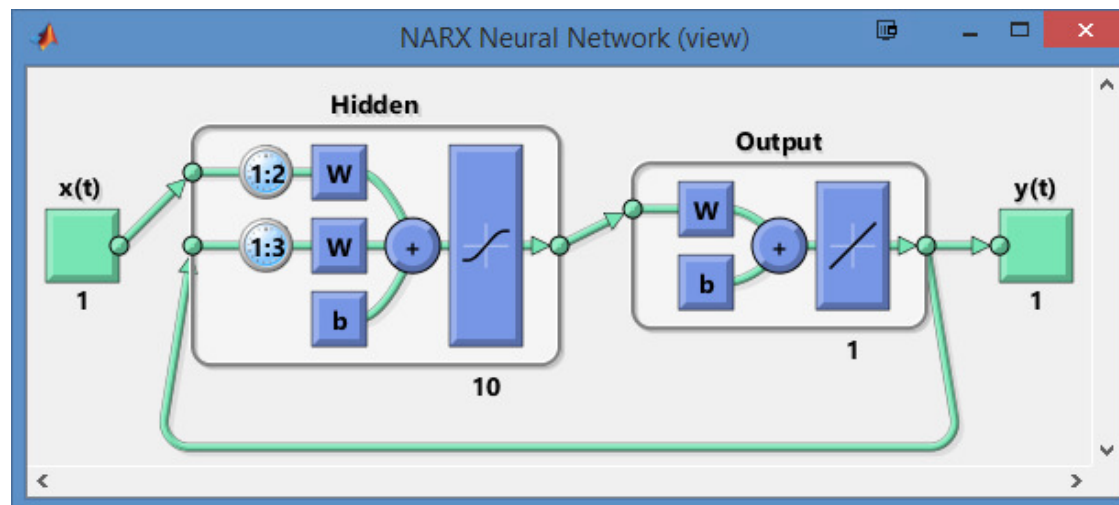
$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u(t-1), u(t-2))$$

- zamknij sieć w pętli sprzężenia

```
siecNARXpoClosed = closeloop ( siecNARXpo ) ;
```

- pokaż strukturę sieci

```
view ( siecNARXpoClosed ) ;
```





## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u(t-1), u(t-2))$$

- przygotuj dane do *symulacji - multi-step-ahead prediction*

```
[ PsekwencjaUcz, PiUcz, AiUcz, tUcz ] = preparets ( siecNARXpoClosed, uUcz, { } , pUcz ) ;  
[ PsekwencjaSpr, PiSpr, AiSpr, tSpr ] = preparets ( siecNARXpoClosed, uSpr, { } , pSpr ) ;
```

- sprawdź sieć - metodą *symulacji - multi-step-ahead prediction*

```
yUczNARXclosed = sim ( siecNARXpoClosed, PsekwencjaUcz, PiUcz ) ; % na uczącym  
ySprNARXclosed = sim ( siecNARXpoClosed, PsekwencjaSpr, PiSpr ) ; % na sprawdzającym
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- zidentyfikuj układ przy pomocy modelu NNARX(3,2,1)

$$\hat{y}(t|\theta) = f(y(t-1), y(t-2), y(t-3), u(t-1), u(t-2))$$

- pokaż wynik *symulacji - multi-step-ahead prediction*

```
odpSieciNARXmultiStep = figure ;
```

```
% na uczącym:
```

```
subplot ( 2,1,1 ), plot ( cell2mat(yUczNARXclosed) , 'b' ), hold on  
                    plot ( cell2mat(tUcz              ) , 'r' )  
title ( 'NARX - Multi-step ahead prediction (simulation) - training data' )
```

```
% na sprawdzającym:
```

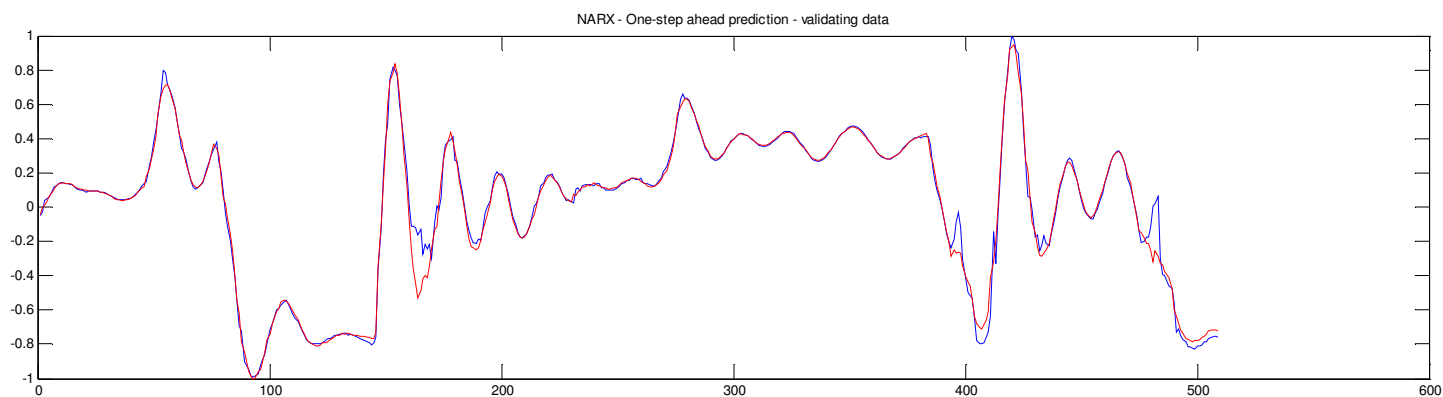
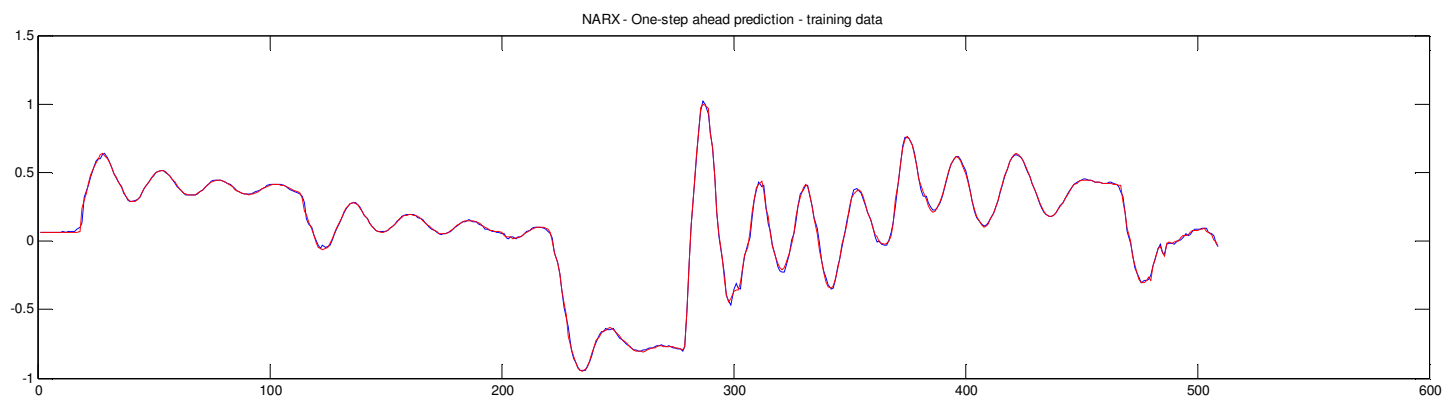
```
subplot ( 2,1,2 ), plot ( cell2mat(ySprNARXclosed) , 'b' ), hold on  
                    plot ( cell2mat(tSpr              ) , 'r' )  
title ( 'NARX - Multi-step ahead prediction (simulation) - validating data' )
```



## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- inna sieć 10-1 - *one-step-ahead*:

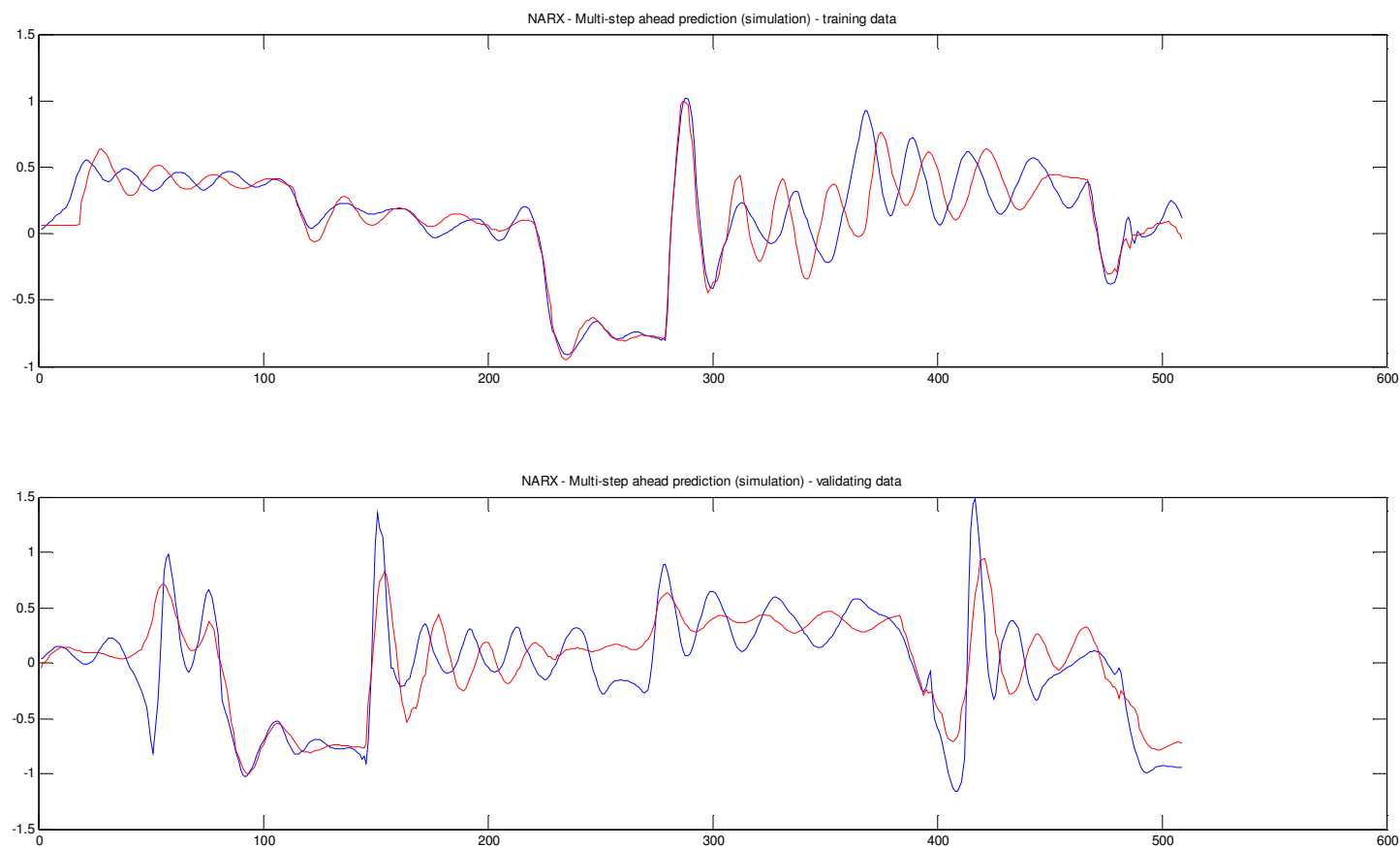




## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- inna sieć 10-1 - *multi-step-ahead*:

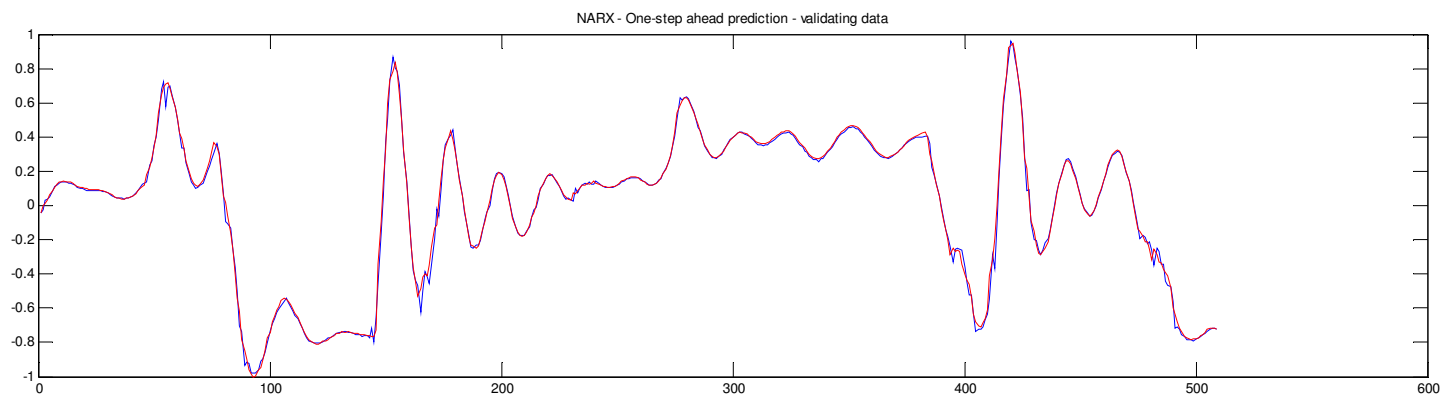
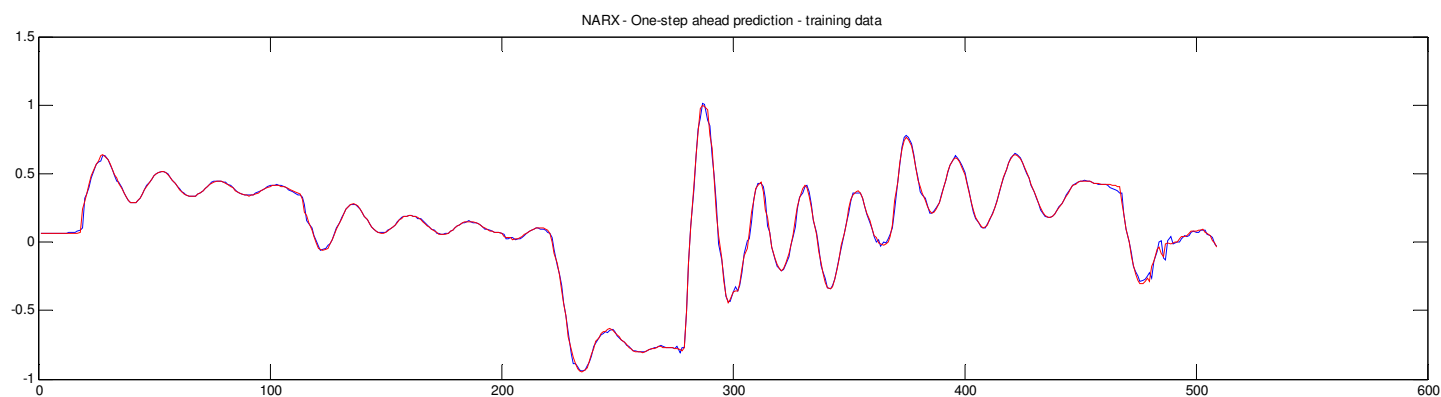




## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- sieć 5-1 - *one-step-ahead*:

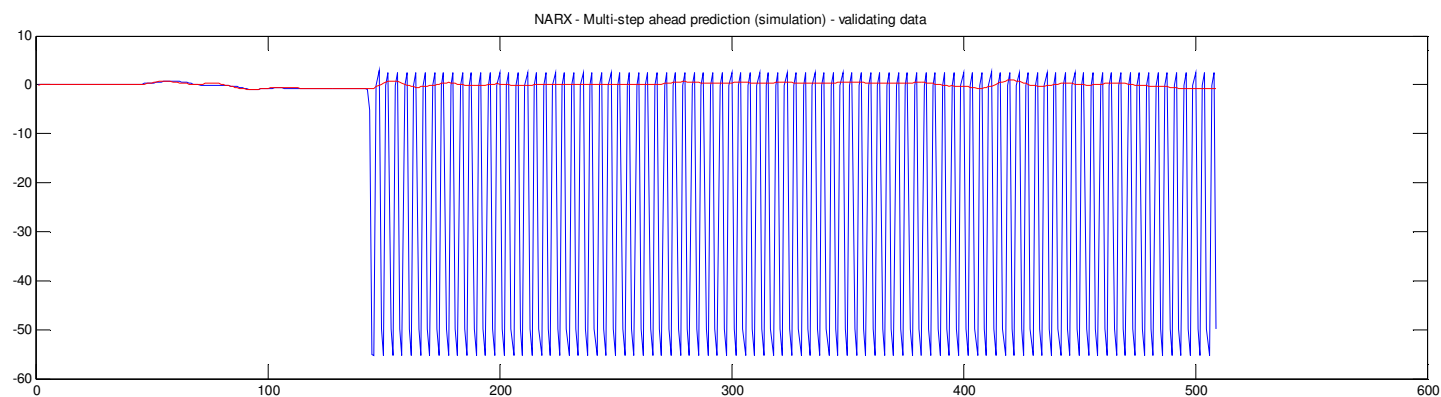
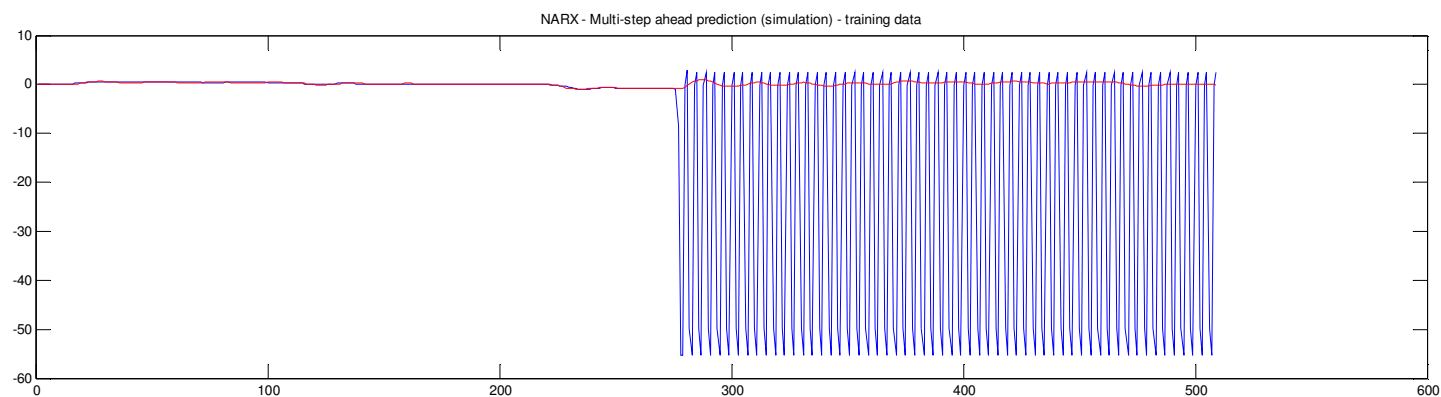




## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- sieć 5-1 - *multi-step-ahead*:

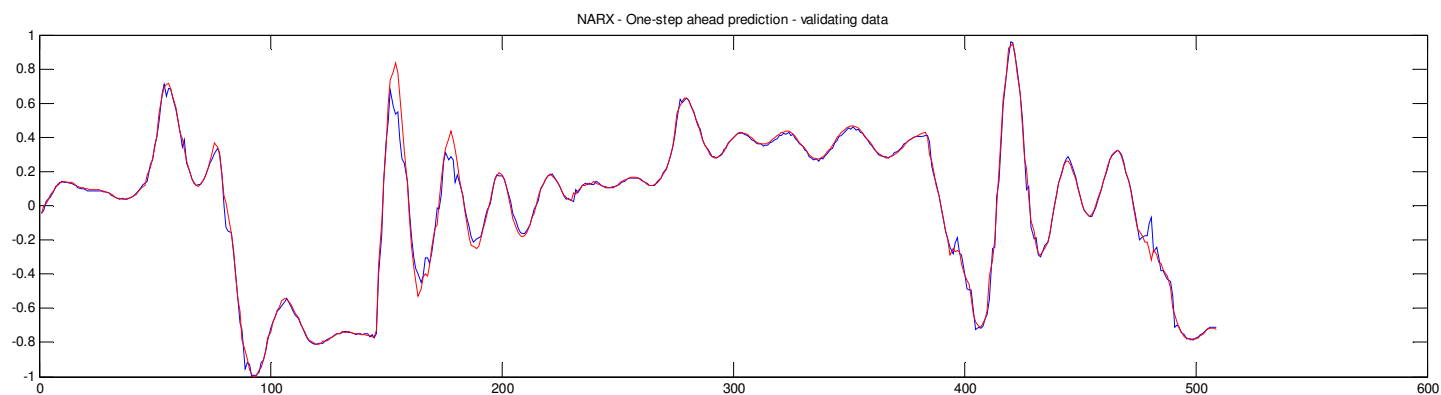
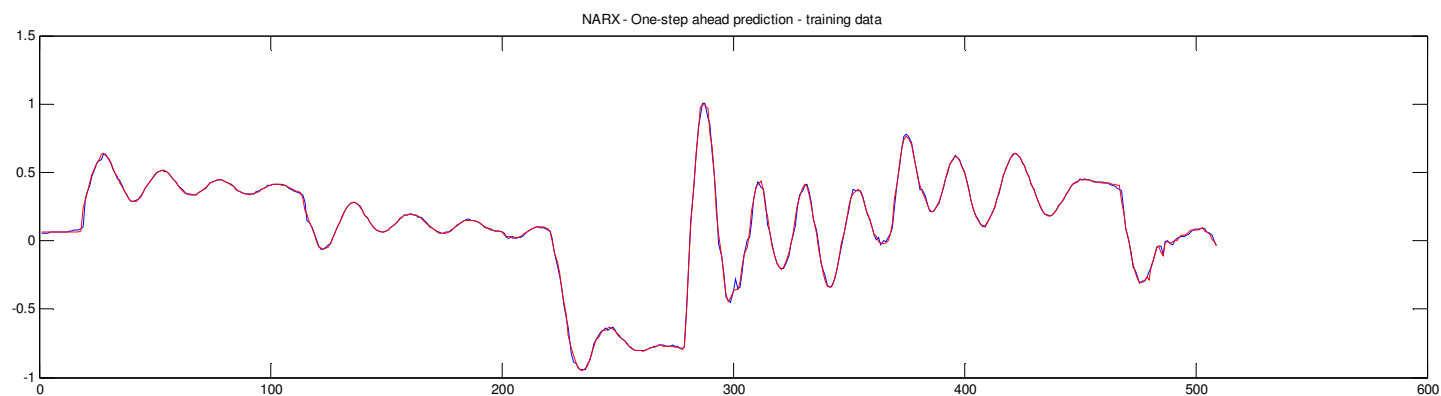




## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- inna sieć 5-1 - *one-step-ahead*:



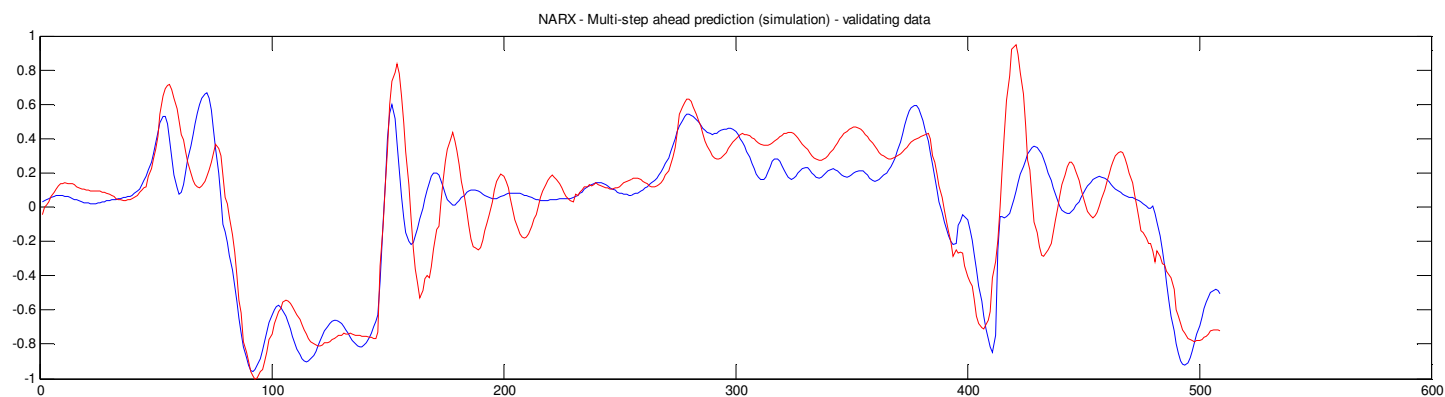
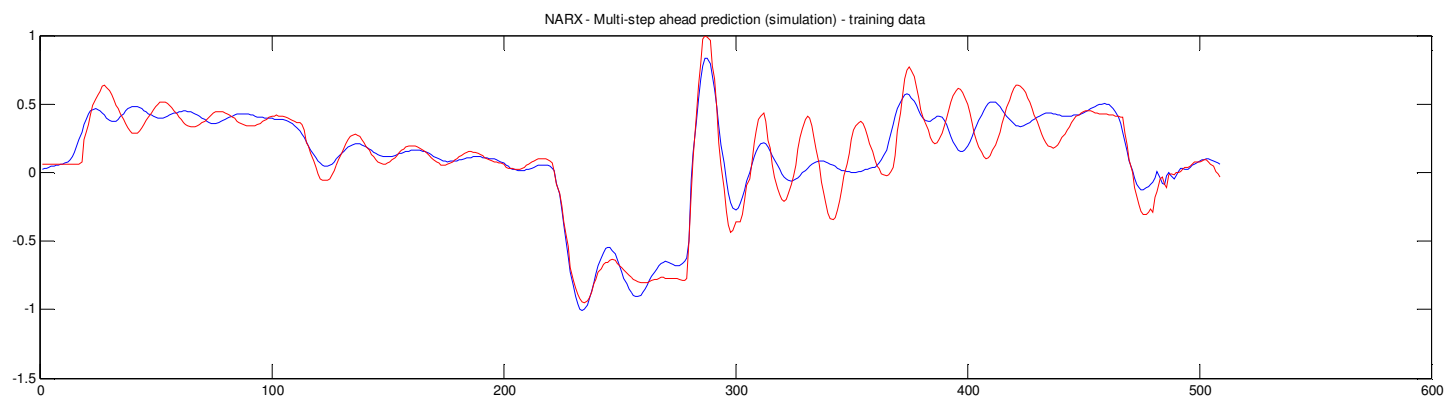




## przykład 1. Modelling of a hydraulic actuator

źródło: M. Nørgaard i in. Neural Networks for Modelling a Control of Dynamic Systems, rozdz. 4.2, str. 211

- inna sieć 5-1 - *multi-step-ahead*:





Przykład 2

# NNSYSID EXAMPLE

**TOOLBOX NNSYSID**



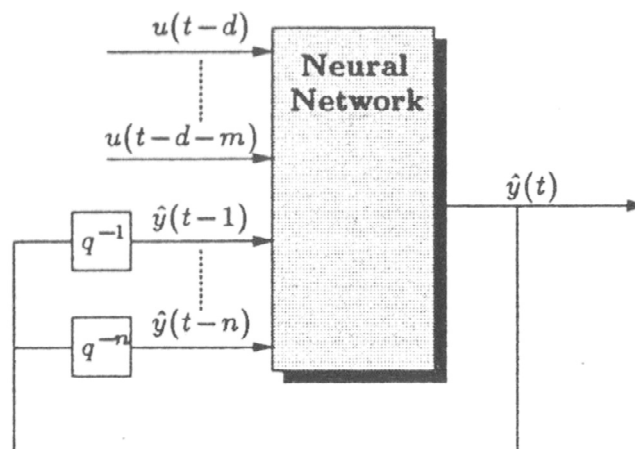
## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$





## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

---



## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

---

- przejdź do katalogu toolboxa: NNSYSID20
- załaduj dane uczące - przebieg symulacji systemu

```
close all  
clear all  
clc
```

```
load spmdata
```

```
whos
```



## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

---

- narysuj przebiegi systemu

```
przebiegiUczace = figure ;

subplot ( 211 )
    plot ( u1 )
    title ( 'Training data - Input sequence' )

subplot ( 212 )
    plot ( y1 )
    title ( 'Training data - Output sequence' )
```



## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

---

- narysuj przebiegi systemu

```
przebiegiSprawdzajace = figure ;
```

```
subplot ( 211 )  
    plot ( u2 )  
    title ( 'Validating data - Input sequence' )
```

```
subplot ( 212 )  
    plot ( y2 )  
    title ( 'Validating data - Output sequence' )
```



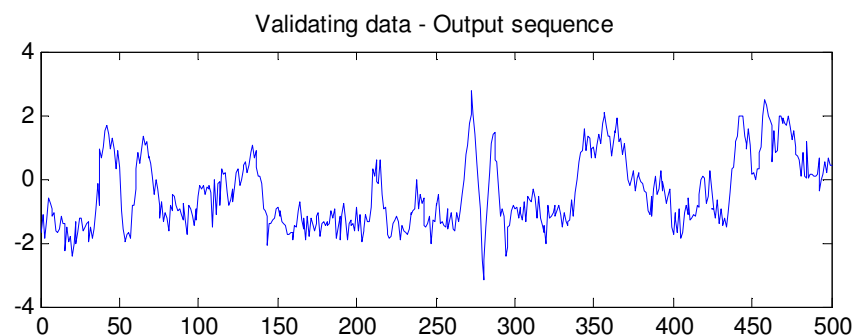
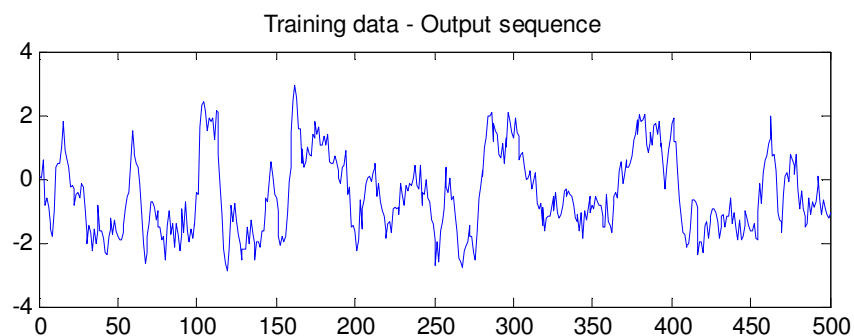
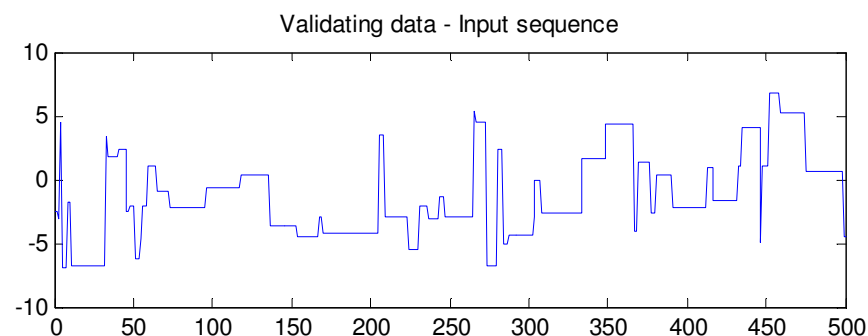
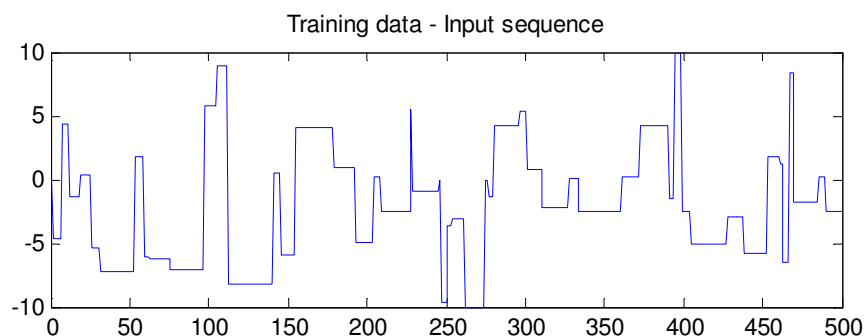
## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$







## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

- przeskaluj dane - średnia 0 , wariancja 1

```
[ u1s , uscales ] = dscale ( u1 ) ;  
[ y1s , yscale ] = dscale ( y1 ) ;  
  
u2s = dscale ( u2 , uscales ) ;  
y2s = dscale ( y2 , yscale ) ;
```



## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

---

- ustal rząd systemu - liczbę potrzebnych wejść do sieci

```
close all % zamykam wszystkie okienka z rysunkami
          % bo ta funkcja wtrynia swoje rysunki na istniejące okienka
```

```
OrderIndices = lipschit ( u1s , y1s , 1:5 , 1:5 ) ;
```



Przykład 2

## NNSYSID EXAMPLE

**MODEL LINIOWY OE**



## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

- dopasuj model liniowy OE (Output Error) - korzystając z: System Identification Toolbox'a

```
th = oe ( [ y1' u1' ] , [ 2 2 1 ] ) ;
```

- przedstaw model

```
present ( th ) ;
```



## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

---

- porównaj predykcje tego systemu z pomiarami
- narysuj wykres autokorelacji błędów  
i wykres korelacji wejść i błędów

```
close all % zamykam okienka pozostałe po Lipshitzu
```

```
figure, compare ( [ y2' u2' ] , th , 1 ) ;
```

```
figure, resid ( [ y2' u2' ] , th ) ;
```



Przykład 2

## NNSYSID EXAMPLE

**MODEL NIELINIOWY NNOE**



## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

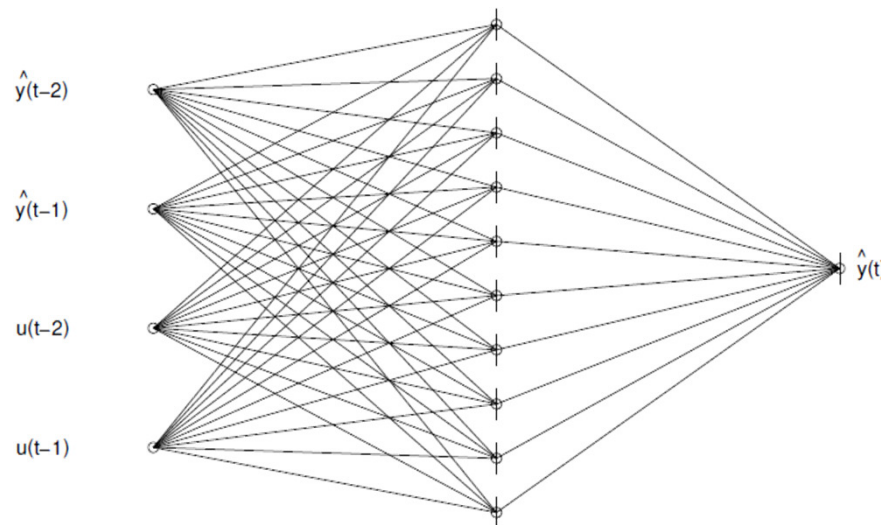
M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

- zdefiniuj strukturę sieci:
  - model drugiego rzędu
  - 10 neuronów ukrytych
  - tangensoidalne funkcje aktywacji

```
NetDef = [ 'HHHHHHHHHH' ; ...  
          'L-----' ] ;  
NN = [ 2 2 1 ] ;
```





## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

- naucz sieć:
  - maksymalnie 300 iteracji
  - mała redukcja wag

```
trparms = settrain ;  
trparms = settrain ( trparms , 'maxiter',300 , 'D',1e-3 , 'skip',10 ) ;  
  
[ W1 , W2 , NSSEvec ] = nnoe ( NetDef , NN , [] , [] , trparms , yls , u1s ) ;
```

- przeskaluj wagi

```
[ w1 , w2 ] = wrescale ( 'nnoe' , W1 , W2 , uscales , yscale , NN ) ;
```





## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

- przetestuj sieć - na zbiorze uczącym

```
close all % zamykam poprzednie okienka
```

```
[ yhat , NSSE ] = nnvalid ( 'nnoe' , NetDef , NN , w1 , w2 , y1 , u1 ) ;
```

- przetestuj sieć - na zbiorze sprawdzającym

```
close all % zamykam poprzednie okienka
```

```
[ yhat , NSSE ] = nnvalid ( 'nnoe' , NetDef , NN , w1 , w2 , y2 , u2 ) ;
```



## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

- zredukuj liczbę wag - algorytmem OBS (*Optimal Brain Surgeon*)
  - po redukcji każdej kolejnej wagi - douczene sieci, max. 50 iteracji

```
close all % zamykam poprzednie okienka
```

```
prparms = [ 50 0 ] ;
```

```
[ thd , trv , fpev , tev , deff , pv ] = ...
```

```
nnprune ( 'nnoe' , NetDef , W1 , W2 , u1s , y1s , NN , ...  
          trparms , prparms , u2s , y2s ) ;
```



## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

---

- przybliż wykres zależności błędu sieci od liczby neuronów

```
set ( gca , 'Ylim' , [ 0 0.25 ] ) ;
```

- ustal optymalną liczbę wag w sieci

```
[ mintev , index ] = min ( tev ( pv ) ) ;
```

```
index = pv ( index )
```



## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

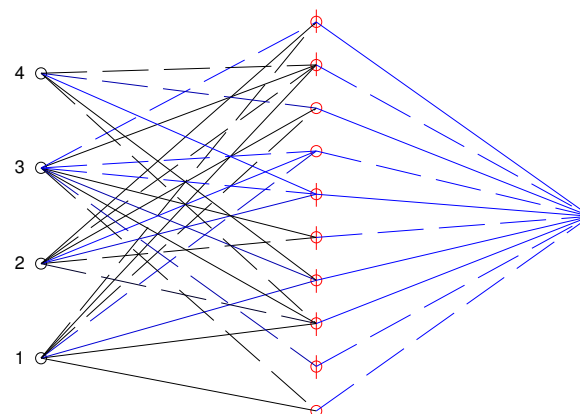
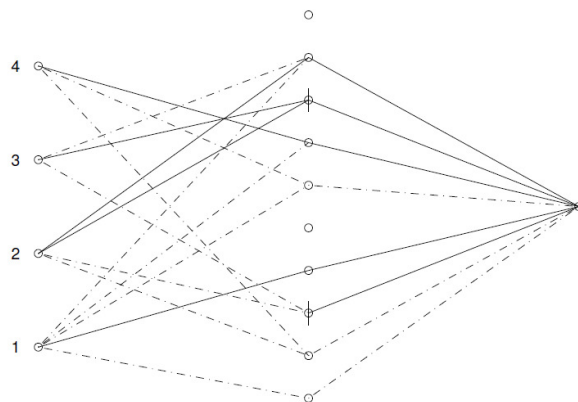
- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

- skonstruuj optymalną sieć

```
close all % zamykam poprzednie okienka
```

```
[ W1 , W2 ] = netstruc ( NetDef , thd , index ) ;
```





## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

- naucz od nowa optymalną sieć
  - bez redukcji wag podczas uczenia

```
trparms = settrain ( trparms , 'D' , 0 ) ;
```

```
[ W1 , W2 , NSSEvec ] = nnoe ( NetDef , NN , W1 , W2 , trparms , yls , u1s ) ;
```

- przeskaluj wagi

```
[ w1 , w2 ] = wrescale ( 'nnoe' , W1 , W2 , uscales , yscale , NN ) ;
```



## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- an open-loop stable, nonlinear, continuous system
- it is known that it is possible to describe it by a nonlinear output error model (NNOE)

$$\hat{y}(t|\theta) = f(\hat{y}(t-1), \hat{y}(t-2), \dots, \hat{y}(t-n), u(t-d), \dots, u(t-d-m))$$

---

- przetestuj ostateczną sieć
  - po pruningu
  - na zbiorze sprawdzającym

```
close all % zamykam poprzednie okienka
```

```
[ yhat , NSSE ] = nnvalid ( 'nnoe' , NetDef , NN , w1 , w2 , y2 , u2 ) ;
```

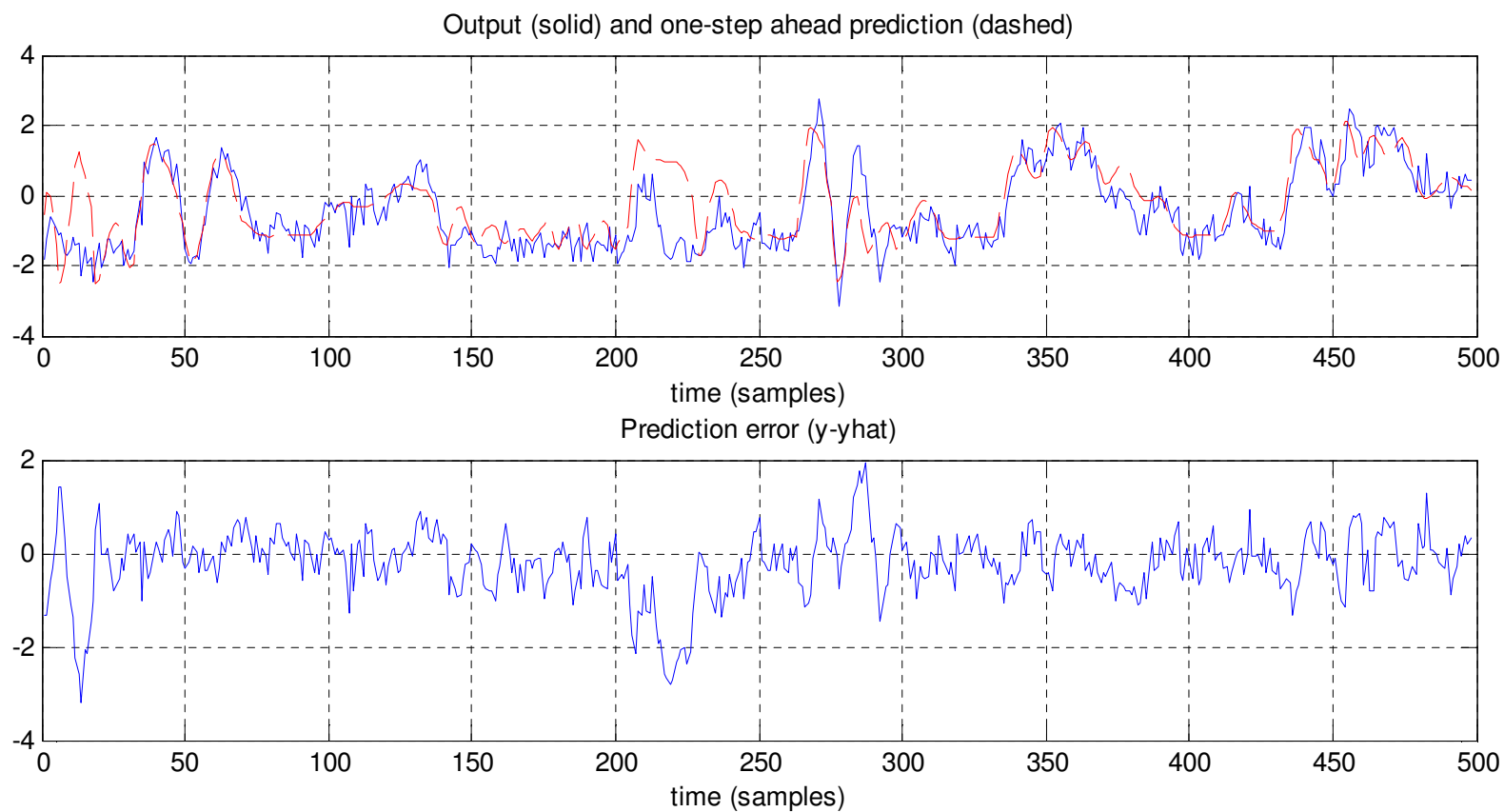


## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- zachowanie sieci na ciągu sprawdzającym - PRZED pruningiem:





## przykład 2. NNSYSID Example

źródło: NNSYSID Toolbox ver. 2, rozdział 3. Example, str. 1-20

M. Nørgaard: "Neural Network Based System Identification Toolbox," Tech. Report. 00-E-891, Department of Automation, Technical University of Denmark, 2000

- zachowanie sieci na ciągu sprawdzającym - PO pruningu:

