

Grzegorz DRAŁUS<sup>1</sup>

## POPRAWA JAKOŚCI DYNAMICZNEGO MODELU ZŁOŻONEGO POPRZEZ ZASTOSOWANIE INTERPOLOWANYCH DANYCH UCZĄCYCH

Artykuł prezentuje wpływ interpolowanych danych uczących na jakość modelu globalnego dynamicznego obiektu złożonego. Obiekt złożony może składać się z kilku dynamicznych obiektów prostych. Do modelowania dynamicznego obiektu złożonego zaproponowano szeregowo-równoległy model identyfikacji, w którym jako metodę uczenia można zastosować algorytm wstecznej propagacji błędów do obliczania gradientu z funkcji jakości, jak w statycznych sieciach neuronowych. Kryterium globalnym oceny jakości modelu jest suma kwadratów błędów między wyjściami obiektów prostych a wyjściami odpowiednich modeli prostych. Obiektem badań jest dynamiczny nieliniowy obiekt złożony, który składa się z szeregowego połączenia dwóch dynamicznych nieliniowych obiektów prostych opisanych nieliniowymi równaniami różnicowymi. Model globalny obiektu złożonego składa się z dwóch połączonych szeregowo dynamicznych modeli prostych. Jako model globalny użyto wielowarstwowe sieci neuronowe typu feedforward z liniami opóźnień w odpowiedniej konfiguracji. Do uczenia modelu globalnego wygenerowano dwa typy danych uczących. Pierwszy typ danych to 100-punktowy zbiór liczb losowych o rozkładzie normalnym z wartościami z przedziału  $\langle -1, 1 \rangle$ . Drugi typ danych powstał przez interpolację (trzema dodatkowymi punktami) 100-punktowego zbioru danych losowych. Zbiór danych interpolowanych zawiera 400 liczb o wartościach z przedziału  $\langle -1, 1 \rangle$ . W artykule przedstawiono wpływ obu typów danych na jakość modelu globalnego i modeli prostych oraz wpływ na szybkość uczenia się modelu.

**Słowa kluczowe:** dynamiczny obiekt złożony, modelowanie globalne, sieć neuronowe

### 1. Wstęp

Istnieje wiele rodzajów obiektów złożonych. Jeden rodzaj to obiekty złożone składające się z bloków (obektów prostych) połączonych w

---

<sup>1</sup> Grzegorz Drałus, Politechnika Rzeszowska, Katedra Elektrotechniki i Podstaw Informatyki ul. W. Pola 2, 35-959 Rzeszów, gregor@prz.edu.pl,

funkcjonalną strukturę [1]. Inny rodzaj to zbiór obiektów, które nie są dokładnie aproksymowalne ze skończonego sparametryzowanego zbioru modeli [3]. Jeszcze inny rodzaj to modele kompleksu operacji [7]. Jak dotąd nie opracowano kompleksowych metod modelowania obiektów złożonych, zarówno statycznych jak też dynamicznych, które są bardzo trudne do zamodelowania.

Modelowanie dynamicznych obiektów prostych jest dość dobrze opisane w literaturze [2,8,10]. Jednak modelowanie dynamicznych obiektów złożonych jest problemem trudnym, dotąd nie w pełni rozwiązany. Na przykład w procesach przemysłowych istnieje konieczność opracowania algorytmów sterowania dla różnych faz procesu. Konieczność opracowania modeli obiektów złożonych jest mocno związana ze sterowaniem tymi procesami. Modele obiektów złożonych bardzo często są budowane w celach poznawczych.

Matematyczne metody modelowania statycznych obiektów złożonych polegają na modelowaniu obiektów prostych, które są częściami obiektu złożonego i ich optymalizacji. Następnie buduje się model złożony z optymalnych modeli prostych, ale taki model nie jest modelem globalnie optymalnym [1,4,5]. Dużym problem było i jest budowa optymalnego modelu globalnego.

Sieci neuronowe dzięki swym podstawowym własnościom (elastyczna struktura, zdolność aproksymacji nieliniowych funkcji, dynamika, przewarzanie równoległe i wielowymiarowość) pozwoliły na budowę modeli globalnych obiektów złożonych. Optymalizacja parametrów modelu prowadzi do powstania modelu globalnie optymalnego. Model globalny, w którym przepływ sygnałów podczas uczenia modelu i podczas jego działania jest taki sam jak w obiekcie złożonym, czyli w pełni odzwierciedla strukturę obiektu złożonego i jego części składowe. Dobór parametrów modelu globalnego (czyli uczenie sieci) należy tak przeprowadzić, aby wyjścia modelu złożonego były jak najbardziej zbliżone do wyjść obiektu złożonego, przy zachowaniu jego struktury. W drugiej kolejności różnice między sygnałami wyjściowymi obiektów prostych i odpowiednich modeli prostych również były jak najmniejsze.

## **2. Model dynamicznego obiektu złożonego**

### **2.1. Szeregowo-równoległy model identyfikacji**

Jeżeli w obiekcie złożonym o dowolnej strukturze, nie występują sprzężenia zwrotne, to taki obiekt można sprowadzić do obiektu o strukturze szeregowej. W takiej strukturze wejściem do kolejnego elementu obiektu jest wyjście z obiektu poprzedniego. Konstruując modele obiektu złożonego można sformułować wielorakie kryteria oceny jakości tych modeli.

Do modelowania prostego obiektu dynamicznego z użyciem sieci neuronowych wielowarstwowych można zastosować szeregowo-równoległy model

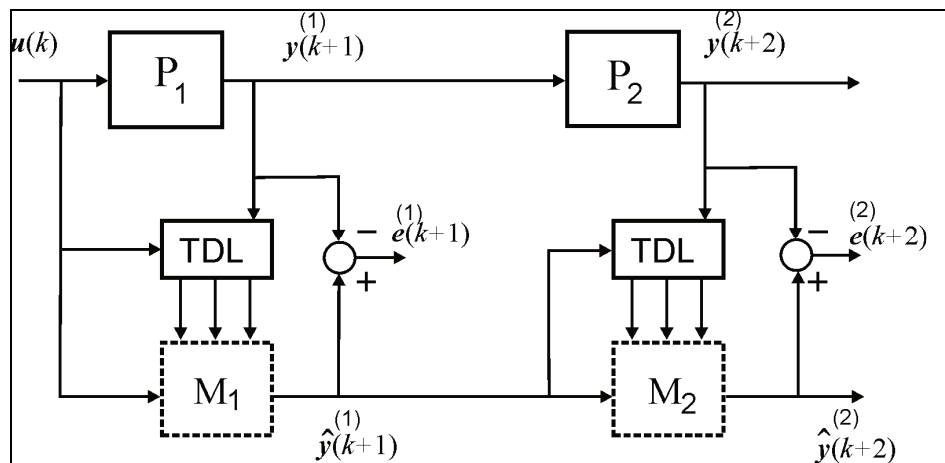
identyfikacji [9]. W takiej strukturze do uczenia sieci wykorzystuje się dane bieżące oraz dane przeszłe z wejścia i wyjścia obiektu przez linie opóźnień. Występuje tu sprzężenie zwrotne nie od sieci, ale od obiektu. Wtedy sieć neuronowa pracuje jak sieć jednokierunkowa i może być uczona według reguły delta (stacyczny algorytm propagacji wstecznej). Zakładając, że błąd modelowania zmierza asymptotycznie do małych wartości, tzn.  $y(t) \approx \hat{y}(t)$  model szeregowo-równoległy może zostać zastąpiony przez model równoległy bez poważniejszych konsekwencji [9].

## 2.2. Model globalny

Zwykle dynamiczny obiekt złożony składa się z kilku dynamicznych obiektów prostych połączonych w pewną strukturę. W świecie rzeczywistym występuje duża różnorodność struktur obiektów złożonych. Jednak jedną z najbardziej typowych struktur jakie występują w przypadku obiektów złożonych, zwłaszcza w przemyśle i technice, to struktura szeregowo-równoległa.

Na rys.1 przedstawiono dynamiczny obiekt złożony o strukturze szeregowo-równoległej. Obiekt składa się z dwóch dynamicznych obiektów prostych  $P_1$  i  $P_2$ , połączonych szeregowo. W takiej strukturze, wyjście pierwszego obiektu prostego jest wejściem do drugiego obiektu prostego. Zatem do modelowania złożonego obiektu dynamicznego o strukturze szeregowo-równoległej, można po pewnej adaptacji wykorzystać szeregowo-równoległą metodę identyfikacji.

Narzędziem (jednym z niewielu) jakie można zastosować do modelowania globalnego dynamicznego obiektu złożonego w połączeniu z szeregowo-równoległą metodą identyfikacji są sieci neuronowe wielowarstwowe.



Rys. 1. Struktura szeregowego dynamicznego obiektu złożonego i jego modelu globalnego

Fig. 1. A diagram of dynamic complex system and his global model

Użycie sieci neuronowych i szeregowo-równoległej metody identyfikacji wymaga pewnej autorskiej adaptacji. Po wyodrębnieniu struktury obiektu złożonego w postaci obiektów prostych  $P_1$  i  $P_2$  budowany jest model globalny o takiej samej strukturze. W tym modelu można wyróżnić dwa modele proste  $M_1$  i  $M_2$ , które wraz z liniami opóźnień sygnałów (ang. TDL), tworzą strukturę dynamiczną. Wyjście pierwszego modelu jest wejściem do modelu drugiego. Ponadto, wyjście obiektu pierwszego  $P_1$  jest podawane, poprzez blok TDL, na wejście modelu pierwszego  $M_1$ , oraz wyjście obiektu drugiego  $P_2$  jest podawane na wejście modelu drugiego  $M_2$  (również poprzez blok TDL, patrz rys.1). Taki rozptyw sygnałów obiektu złożonego i modelu globalnego można nazwać szeregowo-równoległą metodą identyfikacji obiektu złożonego. Ze względu na powyższą strukturę modelu globalnego, jako modele proste użyto wielowarstwowych sieci neuronowych typu feedforward z liniami opóźnień TDL. Kolejnym problemem do rozwiązania jest opracowanie algorytmów uczenia modelu w postaci złożonej sieci neuronowej. Ponieważ w tej postaci modelu nie ma sprzężeń zwrotnych, zatem do uczenia sieci wielowarstwowej można zaadaptować algorytmy oparte na wstecznej propagacji błędów jak w sieciach statycznych. Punktem wyjścia jest zdefiniowanie kryterium jakości dla modelu globalnego.

Na podstawie sygnałów wyjściowych z poszczególnych obiektów prostych i modeli prostych można wyznaczyć błędy modelowania. Zatem błąd dla  $r$ -tego modelu prostego dla aktualnej próbki czasu dyskretnego  $k$  można obliczyć ze wzoru:

$$e^{(r)}(k+r) = \hat{y}^{(r)}(k+r) - y^{(r)}(k+r) \quad (1)$$

gdzie:  $k$  – numer próbki  $\hat{y}^{(r)}$  - wyjście  $r$ -tego modelu prostego,  $y^{(r)}$  - wyjście  $r$ -tego obiektu prostego.

Błędy modelowania czyli różnice między wyjściami modelu prostego a wyjściami obiektu prostego wykorzystano do sformułowania wskaźnika jakości  $r$ -tego modelu prostego:

$$Q^{(r)}(\mathbf{w}^{(r)}) = \frac{1}{2} \sum_{k=1}^K \left( \mathbf{e}^{(r)}(k) \right)^T \mathbf{e}^{(r)}(k) = \frac{1}{2} \sum_{k=1}^K \sum_{j=1}^{J_r} \left( \hat{y}_j^{(r)}(k+r) - y_j^{(r)}(k+r) \right)^2 \quad (2)$$

gdzie:  $K$  – liczba próbek w sygnale,  $J_r$  - liczba wyjść  $r$ -tego obiektu prostego,  $\mathbf{w}^{(r)}$  - wagi  $r$ -tego modelu prostego.

Wypadkowe kryterium jakości dla modelu globalnego można sformułować jako sumę ważoną wszystkich kryteriów jakości modeli prostych:

$$Q_g(\mathbf{W}) = \sum_{r=1}^R \beta_r Q^{(r)}(\mathbf{w}^{(r)}) = \frac{1}{2} \sum_{k=1}^K \sum_{r=1}^R \beta_r \sum_{j=1}^{J_r} (\hat{y}_j^{(r)}(k+r) - y_j^{(r)}(k+r))^2 \quad (3)$$

gdzie:  $R$  – liczba obiektów prostych,  $\mathbf{W} = [\mathbf{w}^{(1)}, \dots, \mathbf{w}^{(r)}]$  – wagi modelu globalnego. Współczynniki  $\beta = [\beta_1 \beta_2 \dots \beta_R]$  określają wpływ poszczególnych modeli prostych na jakość modelu globalnego. Poprzez odpowiedni dobór wartości współczynników  $\beta$  można określić znaczenie modeli prostych w modelu globalnym o wymaganej jakości.

### 2.3. Algorytm uczenia neuronowego modelu globalnego

Do uczenia sieci wielowarstwowych, z których zbudowany jest model globalny konieczne było zmodyfikowanie gradientowego algorytmu wstecznej propagacji błędu. Ta modyfikacja musiała uwzględnić fakt, że model złożony składa się ze statycznych sieci neuronowych z liniami opóźnień (TDL) i sprzężenia zwrotnego od obiektu podczas uczenia oraz sprzężenia od modelu po zakończonym procesie uczenia. Ponadto model globalny składa się z kilku modeli prostych, których wyjścia biorą udział w kryterium globalnym.

W celu minimalizacji funkcji kryterialnej  $Q_g(\mathbf{W})$  zostanie użyta metoda największego spadku gradientu. Stosując tę metodę do uczenia sieci, przyrost wagi wylicza się z gradientu funkcji kryterialnej (3) według poniższego wzoru:

$$\Delta w_{ji} = -\eta \frac{\partial Q_g(\mathbf{W})}{\partial w_{ji}} = -\eta \frac{\partial Q_g(\mathbf{W})}{\partial z_j^m} \frac{\partial z_j^m}{\partial w_{ji}} \quad (4)$$

Przy uczeniu sieci neuronowych można nie używać informacji o wartości gradientu, lecz tylko o jego znaku, co pozwala na znaczne przyspieszenie procesu uczenia, w literaturze algorytm ten znany jest pod nazwą *Rprop* [11].

Poprzez adaptację algorytmu *Rprop* dla złożonej sieci neuronowej o strukturze szeregowo-równoległej powstał złożony dynamiczny algorytm *DCRprop* [6]. Wzory na przyrosty wag dla modelu globalnego po obliczeniu gradientu z kryterium jakości (3) są następujące

- w warstwie wyjściowej:

$$\Delta w_{ji}^{(R),M} = -\eta \sum_{k=1}^K f'(z_j^M(k+R)) \beta_R (\hat{y}_j^{(R)}(k+R) - y_j^{(R)}(k+R)) u_i^{M-1}(k+R-1) \quad (5)$$

- warstwach ukrytych:

$$\Delta w_{ji}^{(r),m} = -\eta \sum_{k=1}^K f'(z_j^m(k+r)) \sum_{l=1}^{I_{m+1}} \delta_l^{(r),m+1}(k) w_{lj}^{(r),m+1} u_i^{m-1}(k+r-1) \quad (6)$$

- w dodatkowych warstwach ukrytych:

$$\Delta w_{ji}^{(r),m} = -\eta_{ji}^p \sum_{k=1}^K f'(z_j^m(k+r)) \left( \sum_{l=1}^{I_{m+1}} \delta_l^{(r+1),m+1}(k+r) w_{lj}^{(r+1),m+1} + \beta_r (\hat{y}_j^{(r)}(k+r) - y_j^{(r)}(k+r)) \right) u_i^{m-1}(k+r-1) \quad (7)$$

gdzie:  $\eta^p$  jest adaptacyjnym współczynnikiem tempa uczenia,  $M$  – ilość warstw sieci globalnej,  $f'(z_j^m(k))$  jest funkcją aktywacji neuronu w  $m$ -tej warstwie,

$z_j^m(k+r)$  jest sygnałem wejściowym funkcji aktywacji  $j$ -tego neuronu,

$u_i^{m-1}(k+r-1)$  jest sygnałem wejściowym  $m$ -tej warstwy,

$\delta_l^{(r+1),m+1} = \frac{\partial Q_g}{\partial z_l^m(k+r)}$  jest błędem *delta*  $l$ -tego neuronu  $m$ -tej warstwy w  $r$ -tym

modelu prostym.

W neuronowym modelu globalnym oprócz zwykłych warstw ukrytych, występują dodatkowe warstwy ukryte, które są warstwami wyjściowymi poszczególnych modeli prostych.

Adaptacyjny współczynnik uczenia  $\eta^p$  w  $p$ -tej epoce uczenia jest obliczany według poniższego wzoru:

$$\eta_{ji}^p = \begin{cases} \min(a \cdot \eta_{ji}^{(p-1)}, \eta_{\max}^p) & \text{gdy } S_{ji}^p \cdot S_{ji}^{(p-1)} > 0 \\ \max(b \cdot \eta_{ji}^{(p-1)}, \eta_{\min}^p) & \text{gdy } S_{ji}^p \cdot S_{ji}^{(p-1)} < 0 \\ \eta_{ji}^{(p-1)} & \text{gdy } S_{ji}^p \cdot S_{ji}^{(p-1)} = 0 \end{cases} \quad (8)$$

gdzie:  $p$  – numer cyklu,  $S_{ji}^p = \frac{\partial Q_g(\mathbf{W}(p))}{\partial w_{ji}}$ , oraz  $a=1.2$ ;  $b=0.5$ ;  $\eta_{\max} = 50$ ;

$\eta_{\min} = 10^{-6}$ .

Dodatkowym kryterium oceny jakości modelowania przyjęto względny błąd procentowy BP. Błąd ten liczony jest dla każdego wyjścia  $r$ -tego modelu prostego:

$$\text{BP}^{(r)} = \frac{\sum_{k=1}^K |\hat{y}^{(r)}(k) - y^{(r)}(k)|}{\sum_{k=1}^K |y^{(r)}(k)|} \cdot 100\% \quad (9)$$

gdzie:  $K$  – ilość próbek,  $k$  – numer próbki,  $r$  – numer modelu,  $y^{(r)}$  – wyjście  $r$ -tego obiektu,  $y^{(r)}$  – wyjście  $r$ -tego modelu.

## 2.4. Cel badań

Celem badań jest analiza zasadności interpolacji próbek losowych i ich zastosowanie jako dane uczące. Zbadanie wpływu danych interpolowanych na jakość modelu globalnego i modeli prostych oraz na szybkość uczenia się modelu globalnego.

Aby zrealizować powyższy cel należy wygenerować dwa rodzaje danych uczących:

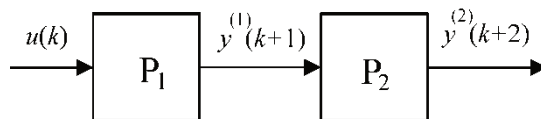
- dane losowe o rozkładzie normalnym o wartościach z przedziału  $\langle -1, 1 \rangle$ ,
- interpolować powyższe dane losowe.

Następnie przeprowadzić symulacje i porównać wyniki czyli wskaźniki jakości modelu globalnego i modeli prostych dla obu typu danych.

## 3. Symulacje i prezentacja wyników

### 3.1. Dynamiczny obiekt złożony

Do zamodelowania został wybrany nieliniowy dynamiczny obiekt złożony, który składa się dwóch dynamicznych nieliniowych obiektów prostych połączonych szeregowo (rys. 2). Obiekt pierwszy oznaczono jako  $P_1$  a obiekt drugi oznaczono jako  $P_2$ . Wejściem do obiektu pierwszego jest sygnał  $u(k)$ .



Rys. 2. Schemat blokowy obiektu złożonego

Fig. 2. A block diagram of a complex system

Sposób działania obu obiektów prostych można opisać za pomocą nieliniowych równań różnicowych drugiego rzędu. Wyjście  $y^{(1)}(k+1)$  pierwszego obiektu prostego  $P_1$  jest opisane następującym nieliniowym równaniem różnicowym:

$$y^{(1)}(k+1) = f_1(y^{(1)}(k), y^{(1)}(k-1), y^{(1)}(k-2), u(k), u(k-1)) \quad (10)$$

Wyjście  $y^{(2)}(k+2)$  drugiego obiektu prostego  $P_2$  również opisane jest nieliniowym równaniem różnicowym:

$$y^{(2)}(k+2) = f_2(y^{(2)}(k+1), y^{(2)}(k), y^{(2)}(k-1), y^{(1)}(k+1), y^{(1)}(k)) \quad (11)$$

Nieliniowe funkcje  $f_1()$  [9] oraz  $f_2()$  [5], które występują we wzorach (10) i (11) zdefiniowane są następującymi wzorami:

$$f_1(x_1, x_2, x_3, x_4, x_5) = \frac{x_1 x_2 x_3 x_5 (x_3 - 1) + x_4}{1 + x_2^2 + x_3^2} \quad (12)$$

oraz

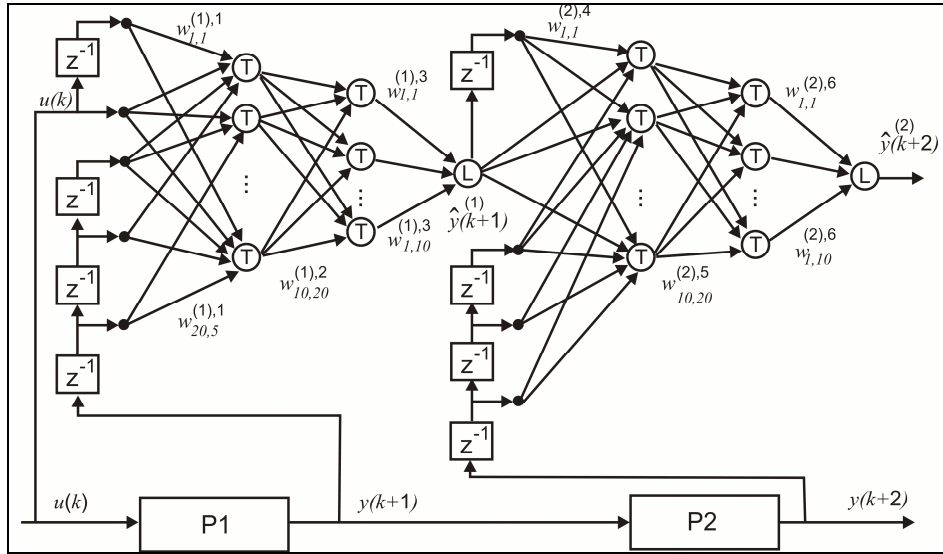
$$f_2(v_1, v_2, v_3, v_4, v_5) = \frac{v_2 v_3 v_5 (v_1 - 1) + v_4}{1 + 2v_1^2} \quad (13)$$

### 3.2. Struktura neuronowego modelu globalnego

Jako model globalny obiektu złożonego przyjęto sieć neuronową 6-cio warstwową (rys. 3) o następującej strukturze: 1 wejście zewnętrzne, 5 neuronów wejściowych, 20 neuronów (typu tangens hiperboliczny) w pierwszej warstwie, 10 neuronów (typu tangens hiperboliczny) w drugiej warstwie, 1 neuron liniowy w warstwie trzeciej, 20 neuronów w warstwie czwartej, 10 neuronów w warstwie piątej i 1 neuron liniowy w warstwie szóstej (wyjściowej modelu złożonego). Warstwa trzecia zwaną „dodatkową” warstwą ukrytą pełni specjalną rolę. Jest granicą między pierwszym a drugim modelem prostym. Wyjście trzeciej warstwy jest wyjściem pierwszego modelu prostego i jednocześnie wejściem do drugiego modelu prostego. Sieć pokazana na rys. 3 jest w trybie uczenia, czyli jest modelem o szeregowo-równoległej strukturze.

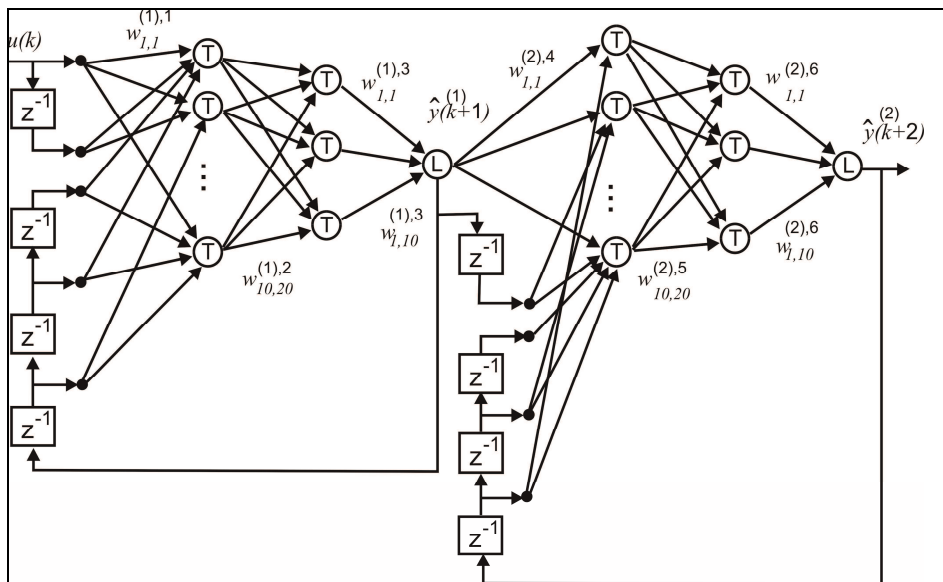
Na rys. 4 przedstawiono sieć w trybie pracy czyli model globalny, który ma strukturę równoległą. W tym modelu występuje sprzężenia zwrotne między wyjściami modeli prostych a ich wejściami. Zatem taki model jest modelem dynamicznym i może modelować dynamiczne obiekty złożone. Linie opóźnień TDL pozwalają realizować w modelu globalnym równania rekurencyjne opisujące obiekt złożony. Zmiana struktury modelu następuje po zakończeniu procesu uczenia sieci, gdy błąd jest dostatecznie mały. Niestety nie ma twardego kryterium kiedy można uznać, że błąd jest dostatecznie mały.





Rys. 3. Model globalny w postaci 6-warstwowej sieci neuronowej w trybie uczenia

Fig. 3. A global model as a 6-th layer neural network in the learning mode



Rys. 4. Dynamiczny model globalny w postaci 6-warstwowej sieci neuronowej w trybie pracy

Fig. 4. A dynamic global mode as a 6-th layer neural network in the working mode

Moment zakończenia procesu uczenia może nastąpić jeżeli wskaźnik jakości modelu globalnego osiągnął założoną minimalną wartość. Dodatkowo można jeszcze uczyć model do momentu, gdy wskaźniki jakości poszczególnych modeli prostych osiągną założone wartości. Jednak w celu określenia szybkości uczenia się modeli można zakończyć proces uczenia po określonej liczbie cykli.

### 3.3. Dane uczące i dane testowe

Do uczenia sieci neuronowych, których zadaniem jest modelowanie nieliniowych funkcji takich jak funkcje (13) i (14) zazwyczaj stosuje się dane losowe. Wygenerowano dwa rodzaje danych uczących. Pierwszy rodzaj to dane losowe o rozkładzie normalnym w zakresie wartości  $\langle -1, 1 \rangle$ , których zbiór zawiera 100 liczb. Dane losowe zostały wygenerowane przez m-funkcję *randn.m* w programie Matlab.

Dane losowe są danymi o dynamice dużej zmienności, co czasami utrudnia albo wydłuża uczenie sieci. Jednym ze sposobów zmniejszenia zmienności danych jest interpolacja danych. Interpolacja dodaje dodatkowe punkty między węzłami (punktami) pochodzącymi z oryginalnych danych losowych. W wyniku interpolacji powstaje funkcja quasi-deterministyczna.

Zatem drugi typ danych uczących wygenerowano poprzez interpolację danych losowych pierwszego typu (zawierających 100 punktów), dodając między każdym węzłem dodatkowe 3 punkty. Uzyskano w ten sposób drugi zestaw danych uczących, który zawiera 400 próbek (liczb). Liczba próbek danych interpolowanych jest 4-krotnie większa niż danych losowych. Do interpolacji danych losowych została użyta funkcja *interp1.m* z pakietu Matlab. Interpolacja danych losowych w tym przypadku nie zmieniała zakresu danych, czyli dane interpolowane pozostają w zakresie  $\langle -1, 1 \rangle$ .

Testowy sygnał wejściowy dla obiektu złożonego i modelu globalnego jest sygnałem deterministycznym w postaci:

$$\begin{aligned} u(k) &= \sin(2\pi k / 250) && \text{dla } k \leq 250 \\ u(k) &= 0.8\sin(2\pi k / 250) + 0.2\sin(2\pi k / 25) && \text{dla } 250 < k \leq 500 \end{aligned} \quad (14)$$

### 3.4. Wyniki symulacji

Symulacje modelowania wykonano na dwóch jednakowych modelach neuronowych (modele miały identyczną strukturę i ilość neuronów). Jeden model był uczony przez 100-punktowe dane losowe o rozkładzie normalnym. Drugi model był uczony przez 400-punktowe interpolowane dane uczące. Wartości współczynników w kryterium globalnym (3) przyjęto następujące:  $\beta_1=0.5$ ,  $\beta_2=0.5$  dla obu modeli. Do uczenia obu modeli użyto złożonego dynamicznego algorytmu *DCRprop*. Proces uczenia zatrzymywano po 500, 1000, 1500, 2000 i 3000 cyklach uczenia.

Wartości kryterium globalnego  $Q_g$ , oraz kryterium modeli prostych  $Q^{(1)}$  i  $Q^{(2)}$  przedstawiono w tabeli 1 dla modelu uczonego 100-punktowym losowym zbiorem uczącym oraz w tabeli 2 dla modelu uczonego 400-punktowym interpolowanym zbiorem uczącym. Wyniki przedstawiono dla modeli o strukturze szeregowo-równoległej dla danych uczących oraz dla modeli o strukturze równoległej dla danych testowych.

Tabela 1. Wartości wskaźników jakości po określonej liczbie cykli uczenia modelu (model uczony przez dane losowe (100 punktów))

Table 1. Values of quality indices after a certain stage of the learning (model learned by random data (100 points))

	$Q^{(1)}$	$Q^{(2)}$	$Q_g$	$Q^{(1)}$	$Q^{(2)}$	$Q_g$
<b>Cykle</b>	model szeregowo-równoległy - dane uczące			model równoległy - dane testowe		
500	0.00520	0.00195	0.00358	9.327	0.749	5.038
1000	0.00255	0.00097	0.00176	6.962	0.444	3.703
1500	0.00181	0.000663	0.00123	6.286	0.407	3.346
2000	0.00146	0.000498	0.000980	6.155	0.313	3.234
3000	0.00106	0.000355	0.000710	5.818	0.257	3.037

Tabela 2. Wartości wskaźników jakości po określonej liczbie cykli uczenia modelu (model uczone przez dane interpolowane (400 punktów))

Table 2. Values of quality indices after a certain stage of the learning (model learned by interpolated data (400 points))

	$Q^{(1)}$	$Q^{(2)}$	$Q_g$	$Q^{(1)}$	$Q^{(2)}$	$Q_g$
<b>Cykle</b>	model szeregowo-równoległy - dane uczące			model równoległy - dane testowe		
500	0.0241	0.00803	0.0160	0.765	0.133	0.449
1000	0.0122	0.00441	0.00831	0.475	0.0639	0.269
1500	0.00847	0.00320	0.00584	0.371	0.0336	0.203
2000	0.00662	0.00242	0.00452	0.312	0.0188	0.166
3000	0.00398	0.00159	0.00279	0.172	0.00641	0.0892

Tabela 3 zawiera błędy procentowe BP obliczone według wzoru (13) dla pierwszego i drugiego modelu prostego, dla 100-punktowych losowych danych uczących (model szeregowo-równoległy) oraz danych testowych (model równoległy). W tabeli 4 zamieszczono błędy procentowe BP dla 400-punktowych interpolowanych danych uczących (model szeregowo-równoległy) i danych testowych (model równoległy).

Tabela 3. Wartości błędów BP po określonych cyklach uczenia modelu (model uczoney przez dane losowe (100 punktów))

Table 3. Values of percentage errors after a certain stage of the learning (model learned by random data (100 points))

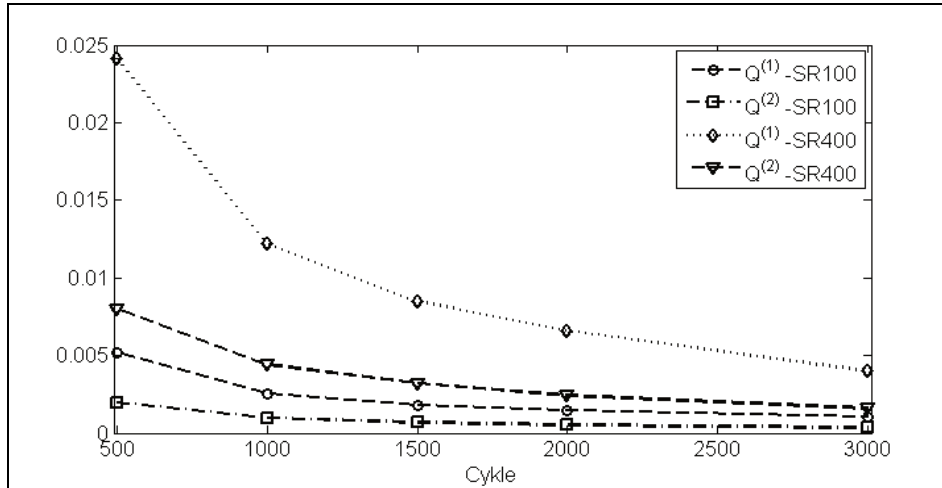
	<b>BP<sup>(1)</sup></b>	<b>BP<sup>(2)</sup></b>	<b>BP<sup>(1)</sup></b>	<b>BP<sup>(2)</sup></b>
	[%]	[%]	[%]	[%]
<b>Cykle</b>	model szeregowo-równoległy - dane uczące		model równoległy - dane testowe	
500	2.49	1.73	19.7	9.92
1000	1.76	1.23	16.7	7.48
1500	1.48	1.03	15.7	7.13
2000	1.34	0.88	15.9	6.20
3000	1.16	0.72	15.6	5.45

Tabela 4. Wartości błędów BP po określonych cyklach uczenia modelu (model uczoney przez dane interpolowane (400 punktów))

Table 4. Values of percentage errors after a certain stage of the learning (model learned by interpolated data (400 points))

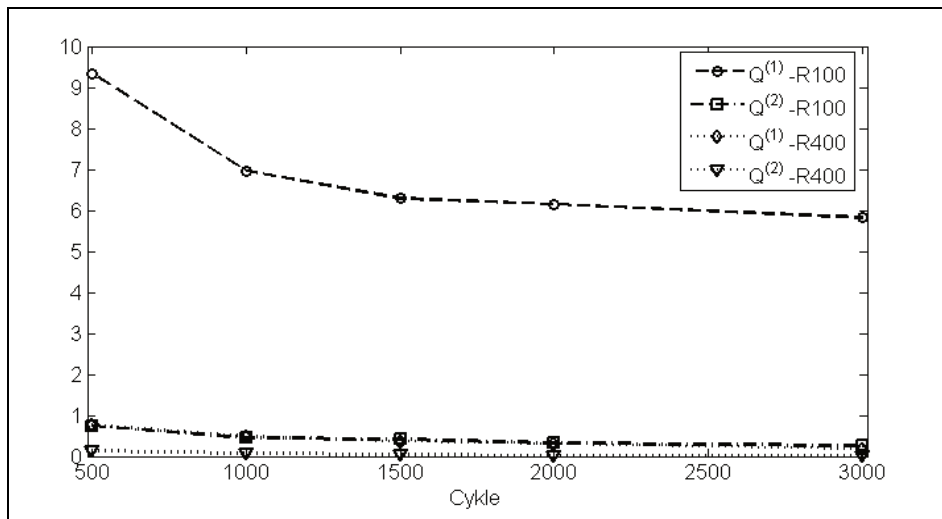
	<b>BP<sup>(1)</sup></b>	<b>BP<sup>(2)</sup></b>	<b>BP<sup>(1)</sup></b>	<b>BP<sup>(2)</sup></b>
	[%]	[%]	[%]	[%]
<b>Cykle</b>	model szeregowo-równoległy - dane uczące		model równoległy - dane testowe	
500	2.20	1.68	4.28	3.25
1000	1.53	1.30	2.84	2.18
1500	1.27	1.10	2.37	1.57
2000	1.14	0.96	2.16	1.19
3000	0.92	0.78	1.86	0.75

Na rys. 5 przedstawiono przebiegi kryteriów jakości modeli prostych dla danych uczących (model uczoney przez dane losowe (100 punktów) i przez dane interpolowane (400 punktów)) dla modelu szeregowo-równoległego w funkcji cykli uczenia. Wartości kryteriów jakości modeli prostych dla danych testowych obu modeli (tryb pracy, modele równoległe) w zależności od liczby cykli uczenia przedstawiono na rys. 6.



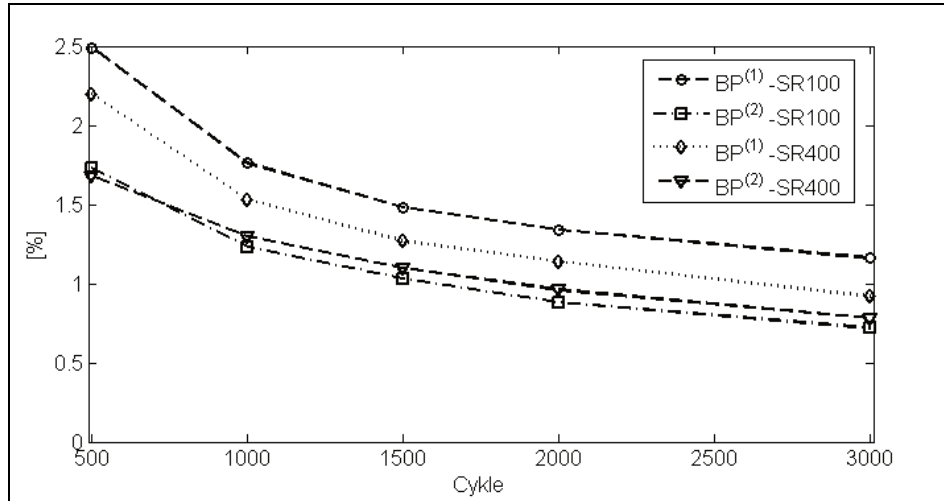
Rys. 5. Wartości kryteriów jakości modeli prostych dla danych uczących (modele w trybie uczenia, SR100– model szeregowo-równoległy uczony przez dane losowe (100 punktów), SR400– model szeregowo-równoległy uczony przez dane interpolowane (400 punktów))

Fig. 5. Values of quality indices of simple models for the training data (SR100 -series-parallel model trained by 100 samples of random data, SR400 -series-parallel model trained by 400 samples of interpolated random data)



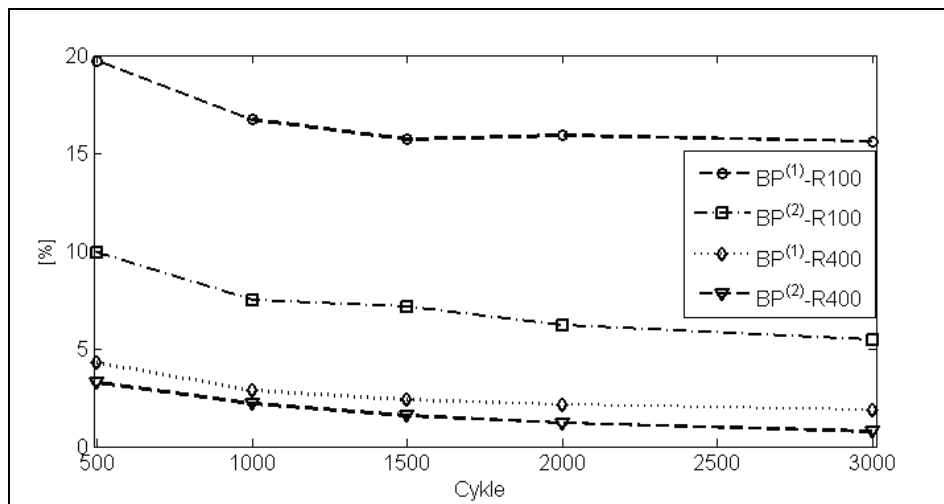
Rys. 6 Wartości kryteriów jakości modeli prostych dla danych testowych (modele w trybie pracy, R100 – model równoległy uczony przez dane losowe (100 punktów), R400 – model równoległy uczony przez dane interpolowane (400 punktów))

Fig. 6 Values of quality indices of simple models for the training data (R100-parallel model trained by 100 samples of random data, R400 - parallel model, trained by 400 samples of interpolated random data)



Rys. 7. Wartości błędów procentowych modeli prostych dla danych uczących (modele w trybie uczenia, SR100 – model szeregowo-równoległy uczony przez dane losowe (100 punktów), SR400 – model szeregowo-równoległy uczony przez dane interpolowane (400 punktów))

Fig. 7. Values of percent errors of simple models for the training data (SR100 – series-parallel model trained by 100 samples of random data, SR400 – series-parallel model trained by 400 samples of interpolated random data)

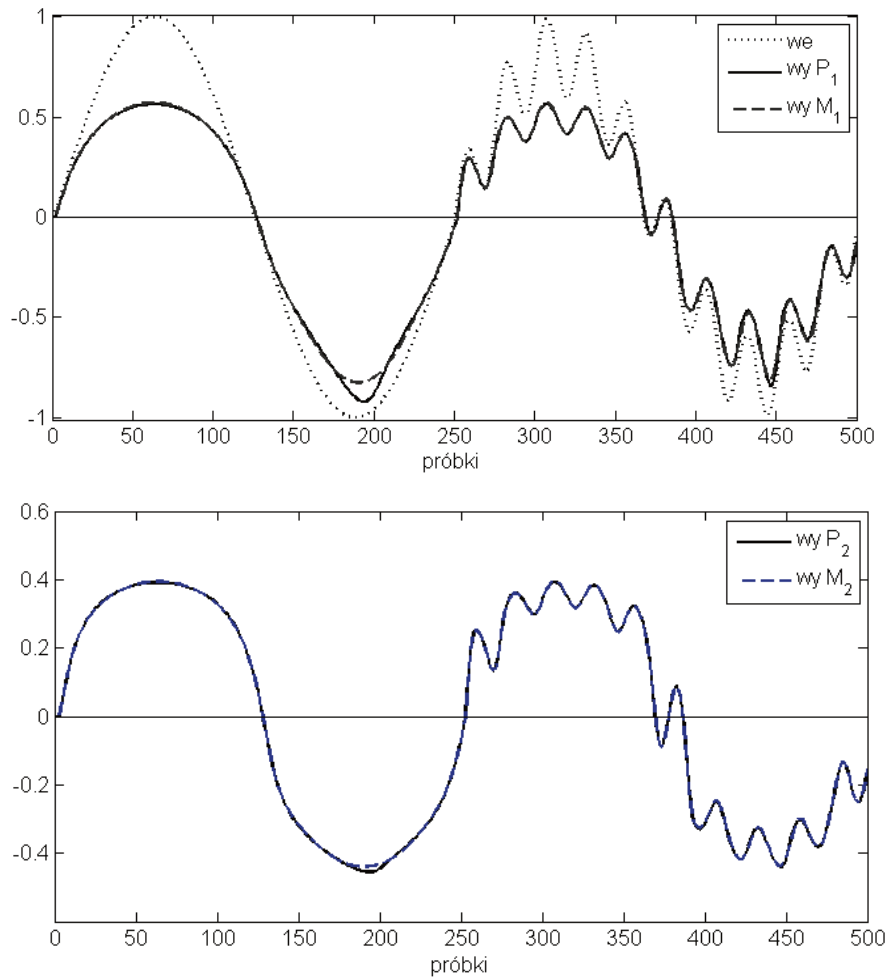


Rys. 8. Wartości błędów procentowych modeli prostych dla danych testowych (modele w trybie pracy, R100 – model równoległy, uczony przez dane losowe (100 punktów), R400 – model równoległy, uczony przez dane interpolowane (400 punktów))

Fig. 8. Values of percent errors of simple models for the training data (R100 – parallel model trained by 100 samples of random data, R400 – parallel model trained by 400 samples of interpolated random data)

Na rys. 7 przedstawiono przebiegi błędów procentowych BP modeli prostych dla danych uczących (dane losowe i dane interpolowane) dla modelu szeregowo-równoległego w zależności od liczby cykli uczenia. Wartości błędów BP modeli prostych dla danych testowych obu modeli (modele równoległe) w funkcji cykli uczenia przedstawiono na rys. 8.

Na rys. 9 przedstawiono przebiegi sygnału testującego dla obu modeli prostych po 3000 cyklach uczenia, w modelu globalnym uczonym przez dane interpolowane.



Rys. 9. Sygnał testowy na wejściu (we) i wyjściu pierwszego obiektu prostego (wy P<sub>1</sub>) i modelu prostego (wy M<sub>1</sub>), oraz sygnał testowy na wyjściu drugiego obiektu prostego (wy P<sub>2</sub>) i modelu prostego (wy M<sub>2</sub>)

Fig. 9. The test signals on the output of the first simple object and the first simple model (M<sub>1</sub>), the test signals on the output of the second simple object and the second simple model (M<sub>2</sub>)

## 4. Wnioski

### 4.1. Wpływu typu danych uczących na szybkość uczenia się modeli

Największy spadek wartości kryteriów jakości (im mniejsza wartość kryterium tym wyższa jakość modelu) modeli prostych występuje między 500 a 1000 cyklem uczenia, oprócz (nienotowanego) przedziału między 1 i 500 cyklem. Dalsze uczenie modeli postępuje coraz wolniej, co jest zgodne z charakterem tego typu algorytmów. Moment zakończenia uczenia następuje po 3000 cyklach uczenia, kiedy poprawa jakości modeli jest niewielka.

Wskaźnik jakości modelu pierwszego  $Q^{(1)}$  przyjmuje 2-3 krotnie większe wartości niż wskaźnik jakości modelu drugiego  $Q^{(2)}$ . Po pierwsze dlatego, że sygnał wyjściowy modelu pierwszego jest prawie dwukrotnie większy od sygnału wyjściowego modelu drugiego. Po drugie, obiekt pierwszy ma silniejszą nieliniowość niż obiekt drugi.

Model globalny uczony przez 100-punktowy zbiór danych losowych (w skrócie: dane losowe) uczy się 4-5 razy szybciej niż model uczony przez 400-punktowy zbiór danych interpolowanych (w skrócie: dane interpolowane). Na przykład, dla 1000 cykli dla danych losowych wartość wskaźnika  $Q^{(1)}$  wynosi 0.00255 a dla danych interpolowanych  $Q^{(1)}$  wynosi 0.0122. Na czterokrotnie większą szybkość uczenia się modelu dla danych losowych wpływa fakt, że ten zbiór jest 4-krotnie mniejszy niż zbiór danych interpolowanych.

### 4.2. Wpływu typu danych uczących na jakość modeli

W modelu globalnym uczonym 100-punktowym zbiorem danych losowych wartości wskaźników jakości są około 3-4 krotnie mniejsze niż modelu, który był uczony 400-punktowym zbiorem danych interpolowanych. Na przykład, po 3000 cyklach, wskaźniki jakości w szeregowo-równoległej konfiguracji modeli są następujące:  $Q^{(1)}=0.00106$ ,  $Q^{(1)}=0.000355$ ,  $Q_g=0.000710$  dla danych losowych, oraz  $Q^{(1)}=0.00389$ ,  $Q^{(1)}=0.00159$ , oraz  $Q_g=0.00279$  dla danych interpolowanych.

Dokładne oszacowanie jakości modelu dają wskaźniki w postaci błędów procentowych BP. Dla danych uczących, błąd procentowy pierwszego modelu prostego (uczonego danymi losowymi) wynosi  $BP^{(1)}=1.16\%$ , a uczonego danymi interpolowanymi wynosi  $BP^{(1)}=0.92\%$ . Natomiast błąd procentowy drugiego modelu prostego (a zarazem modelu globalnego) uczonego danymi losowymi wynosi  $BP^{(2)}=0.72\%$ , a uczonego danymi interpolowanymi wynosi  $BP^{(2)}=0.78\%$ . Dla danych uczących jakość pierwszego modelu prostego jest lepsza gdy uczonego danymi interpolowanymi, a jakość modelu globalnego (czyli drugiego modelu prostego) jest lesza, gdy uczonego był przez dane losowe.

Zupełnie inne wyniki zaobserwowano dla modelu równoległego (w trybie pracy) dla danych testowych. Dla danych testowych (obliczenia po 3000



cyklach), wartości wskaźników jakości są około 30-40 krotnie większe w modelu globalnym uczonym 100-punktowym zbiorem danych losowych niż w modelu uczonym 400-punktowym zbiorem danych interpolowanych. I tak, dla modelu, który był uczony przez dane losowe wskaźnik  $Q^{(1)}=5.818$ , a dla modelu, który był uczony przez dane interpolowane wskaźnik  $Q^{(1)}=0.172$ . Podobnie dla modelu uczonego danymi losowymi wskaźnik  $Q^{(2)}=0.257$ , a dla modelu uczonego danymi interpolowanymi wskaźnik  $Q^{(2)}=0.00641$ .

Również w błędach procentowych modeli obliczonych dla danych testowych widać dużą zależność jakości modeli od typu danych uczących.

Dla danych testowych, błąd procentowy pierwszego modelu prostego (uczonego danymi losowymi) wynosi  $BP^{(1)}=15.6\%$ , a uczonego danymi interpolowanymi wynosi  $BP^{(1)}=1.86\%$ . Natomiast błąd procentowy drugiego modelu prostego (a zarazem modelu globalnego) uczonego danymi losowymi wynosi  $BP^{(2)}=5.45\%$ , a uczonego danymi interpolowanymi wynosi  $BP^{(2)}=0.75\%$ . Zatem dla danych testowych jakość pierwszego modelu prostego jest około 8-krotnie lepsza, a jakość modelu globalnego jest około 7-krotnie lepsza, gdy model globalny uczony był danymi interpolowanymi niż w przypadku uczenia przez dane losowe.

Można też zauważyć, że błąd procentowy modelu globalnego dla danych testowych jest mniejszy ( $BP^{(2)}=0.75\%$ ) niż dla danych uczących ( $BP^{(2)}=0.78\%$ ).

## 5. Podsumowanie

W artykule przedstawiono wyniki komputerowych symulacji dla modelu globalnego dynamicznego obiektu złożonego. Otrzymane wyniki potwierdzają możliwość budowy modeli globalnych dynamicznych obiektów złożonych przy użyciu statycznych sieci wielowarstwowych z liniami TDL w strukturze dynamicznej.

Uzyskane wyniki wskazują, że użycie danych interpolowanych wydłuża czas uczenia modelu ze względu na wzrost liczby próbek proporcjonalnie do ich ilości w stosunku do ilości próbek losowych. W tym przypadku 4-krotny wzrost liczby próbek 4 krotnie wydłuża czas uczenia modelu globalnego. Jednak użycie danych interpolowanych powoduje około 8-krotny wzrost jakości modelu globalnego i modeli prostych.

Dla danych uczących, w konfiguracji szeregowo-równoległej modelu, błędy procentowe dla obu modeli nie różnią się znacząco (są na poziomie 1%). Dla danych testujących, w konfiguracji równoległej modelu, błędy procentowe są na poziomie jednego procenta ( $BP^{(2)}=0.75\%$ ) dla modelu globalnego uczonego danymi interpolowanymi oraz na poziomie około 6% ( $BP^{(2)}=5.45\%$ ) dla modelu uczonego danymi losowymi.

Interpolacja danych losowych znacząco polepsza jakość neuronowego modelu globalnego oraz dynamicznych modeli prostych złożonego obiektu

dynamicznego tej klasy, jak rozpatrywany w tym artykule.

## Literatura

- [1] Bubnicki Z., *Identyfikacja obiektów sterowania*, PWN, Warszawa, 1974.
- [2] Dahleh MA., Venkatesh Sr., *System Identification for Complex Systems: Problem Formulation and Resuls*, In Proceedings of the 36th IEEE Conf. on Dec. and Control, 1997.
- [3] Czemplik A.: *Modele dynamiki układów fizycznych dla inżynierów. Zasady i przykłady konstrukcji modeli dynamicznych obiektów automatyki*, WNT, Warszawa 2008.
- [4] Drałus G., Świątek J.: *Sieci neuronowe w modelowaniu złożonego obiektu chemicznego*. XIV Krajowa Konferencja Automatyki, Zielona Góra, 2002, s.825-830.
- [5] Drałus G., *Modeling of Dynamic Nonlinear Complex Systems Using Neural Networks*. Proceedings of the 15<sup>th</sup> "International Conference on Systems Science", Wrocław, 2004, vol. III, pp.87-96.
- [6] Drałus G., Świątek J., *Static and dynamic complex models: comparison and application to chemical systems*, *Kybernetes: The International Journal of Systems & Cybernetics*, Emerald, Vol. 38, No.7/8, 2009.
- [7] Józefczyk J., *Wybrane problemy podejmowania decyzji w kompleksach operacji*, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław 2001.
- [8] Ljung L.: *System Identification. Theory for the user*, Prentice Hall, 1987
- [9] Narendra K.S., Parthasarathy K., *Identification and Control of Dynamic Systems Using Neural Network*, *IEEE Trans. On Neural Networks*, 1990, vol 1. No.1, s.4-27
- [10] Osowski S.: *Modelowanie i symulacja układów i procesów dynamicznych*, Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa 2007.
- [11] Riedmiller M., Braun H., *RPROP – a fast adaptive learning algorithm*, Technical Report, University Karlsruhe, 1992.

## IMPROVING THE QUALITY OF DYNAMIC COMPLEX MODEL BY USE OF INTERPOLATED LEARNING DATA

### Summary

The main goal of this paper was study the impact of types of training data on the quality of the global model and the quality of simple models of dynamic complex system. It were considered two types of the training data: random data which contained 100 points with normal distribution and data which contained 400 points of interpolated random data. For simulations a dynamic complex system, which consists of two non-linear dynamic objects, connected in series was considered. The complex dynamic system is described by two nonlinear discrete functions. A global model of this system was built from multi-layer neural network in a dynamic structure. The global model was divided into two dynamic simple models in accordance to the construction of the complex system. As a global quality criterion was adopted the weighted sum of quality criteria of dynamic simple models. Two types of training data were generated: a set of random numbers with normal distribution in interval  $\langle -1,1 \rangle$  and a set of interpolated random numbers (i.e. interpolation of the first random set) in interval  $\langle -1,1 \rangle$ . The random set contains 100 numbers, the interpolated set contains 400 numbers.

Kind of training random data has an influence on the learning speed of the models, and has an influence on the quality of the simple models and the global model. Interpolated random data significantly improves (about 8 times) the quality of the dynamic global model of the complex system. The obtained results show, that by appropriate choice of the training data can be obtained very good quality of the global model (i.e. the  $BP^{(2)}$  index reaches less than 1 percent) and at the same time very high quality of the first simple model (i.e. the  $BP^{(1)}$  index reaches a value of about 2 percent).

**Keywords:** modeling systems, global models, dynamic complex systems, multilayer neural networks.

DOI: 10.7862/re.2015.18

*Tekst złożono w redakcji:* luty 2015

*Przyjęto do druku:* kwiecień 2015