

*neuronowe regulatory prędkości, adaptacja on-line,
układ dwumasowy, napęd elektryczny*

Teresa ORŁOWSKA-KOWALSKA*, Marcin KAMIŃSKI*

ANALIZA PORÓWNAWCZA WYBRANYCH NEUROREGULATORÓW DLA NAPĘDU Z POŁĄCZENIEM SPRĘŻYSTYM – OPIS ZASTOSOWANYCH MODELI

W artykule przedstawiono zagadnienia adaptacyjnego sterowania napędem elektrycznym z połączeniem sprężystym przy wykorzystaniu modeli neuronowych trenowanych *on-line*. Jako regulatory neuronowe proponowane są sieci perceptronowe MLP z jedną warstwą ukrytą oraz sieci z radialnymi funkcjami bazowymi RBF, uczone metodą wstecznej propagacji błędów na podstawie algorytmów gradientowych. Opisano struktury regulatorów neuronowych oraz pokazano zależności analityczne wykorzystywane do adaptacji wag i centrów odpowiednich struktur neuroregulatorów.

1. WPROWADZENIE

Sztuczne sieci neuronowe, dzięki swoim właściwościom aproksymacyjnym, coraz częściej znajdują zastosowanie w sterowaniu napędami elektrycznymi [1]. W ostatnich latach powstało wiele publikacji na temat układów regulacji prędkości silników elektrycznych wykorzystujących sieci neuronowe [1]–[11]. Stosowane podejścia, różnią się zarówno rodzajem SN, metodami ich uczenia, sposobem oddziaływania na obiekt, jak i ogólną strukturą systemu sterowania.

Jedną z metod, często proponowanych w strukturach sterowania napędem elektrycznym, jest regulacja adaptacyjna z modelem odniesienia [2], [3]. Kolejny, również często spotykany neuronowy układ regulacji, opiera się na tzw. modelu odwrotnym obiektu [4], [5]. Jednak taka metodyka wymaga opracowania modelu neuronowego na podstawie danych reprezentujących obiektu. W związku z tym pojawiają się problemy związane z przygotowaniem danych treningowych, wybór parametrów uczenia oraz

* Politechnika Wroclawska, Instytut Maszyn, Napędów i Pomiarów Elektrycznych, ul. Smoluchowskiego 19, 50-372 Wrocław, e-mail: teresa.orlowska-kowalska @pwr.wroc.pl, marcin.kaminski@pwr.wroc.pl

struktury modelu neuronowego. Jeszcze inne rozwiązanie pokazane zostało w [6], gdzie SN wykorzystywana jest do uaktualniania nastaw regulatora klasycznego, w zależności od zmian parametrów obiektu. Uzyskano w ten sposób tzw. samo-nastrajający się neuroregulator. Spotykane są również kombinacje SN działającej razem z regulatorem klasycznym. W pracy [7] sieć neuronowa została wykorzystana do jednokierunkowej kompensacji momentu obciążenia, przy czym sygnał sterujący jest sumą sygnałów z regulatora PI i SN. Innym rozwiązaniem spotykanym w literaturze, jest zastosowanie neuronowego regulatora, który do generowania sygnału sterowania, wykorzystuje liniową sieć typu ADALINE [8]–[10] lub sieć perceptronową [11].

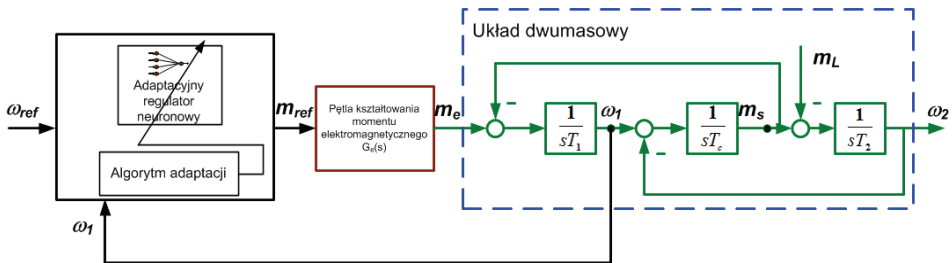
Regulatory neuronowe stosowane w układach napędowych można podzielić na trenowane *off-line* i *on-line*. Struktury sieci neuronowych, w których współczynniki wagowe dobierane są *off-line*, wymagają znacznie bardziej złożonej strategii projektowania, często opartej na doświadczeniu użytkownika. W kolejnych etapach opracowywania modelu należy podjąć decyzję o topologii wykorzystywanej sieci neuronowej, przy czym wybierana jest nie tylko liczba neuronów, ale również liczba warstw ukrytych. Dobór optymalnej struktury sieci neuronowej jest trudny. W wielu przypadkach wymaga zastosowania złożonych obliczeniowo algorytmów optymalizacyjnych [12], [13]. Poza tym opracowanie poprawnych, reprezentatywnych przebiegów treningowych w rzeczywistych aplikacjach jest skomplikowane, a często może być wręcz niemożliwe. Kolejnym zagadnieniem jest znacząca wrażliwość sieci neuronowej na parametry algorytmu treningowego, które najczęściej są dobierane doświadczalnie. W przypadku modeli statycznych, bez wewnętrznych sprzężeń zwrotnych, należy także opracować odpowiednią postać wektora wejściowego, ułatwiającą przetwarzanie przebiegów dynamicznych. W tym celu stosowane są filtry oraz opóźnienia sygnałów wejściowych, których zastosowanie nie jest jasno sprecyzowane w literaturze.

Wiele problemów projektowych związanych z zastosowaniem modeli neuronowych trenowanych *off-line* nie występuje w przypadku rozwiązań opartych na strukturach neuronowych trenowanych *on-line*, w których adaptacja wag sieci neuronowej następuje po pierwszym uruchomieniu układu sterowania, w trakcie pierwszych kilku cykli pracy systemu. Sposób aktualizacji współczynników determinuje możliwości aplikacyjne oraz proces projektowania układu wykorzystującego sieć neuronową [12]. W niniejszym artykule przedmiotem badań są sieci neuronowe trenowane *on-line*, zastosowane jako regulatory prędkości w napędzie elektrycznym z połączeniem elastycznym.

2. BADANY OBIEKT

Badanym obiektem sterowania jest napęd elektryczny, w którym silnik elektryczny jest połączony z maszyną obciążającą sprężystym wałem. Zakłada się, że wewnętrzna struktura sterowania momentem (prądem) silnika elektrycznego zapewnia bardzo do-

brą dynamikę (praktycznie bezinercyjne wymuszenie momentu silnika napędowego) [9], [10], w związku z czym opóźnienie w wewnętrznym obwodzie sterowania prędkością napędu jest pomijalnie małe (zakłada się, że o dynamice zmian momentu decyduje mała stała czasowa T_{me}). Założenie to umożliwia analizę struktury regulacji prędkości napędu niezależnie od zastosowanego silnika napędowego i pozwala odnosić uzyskane wyniki zarówno do napędu z silnikiem obcowzbudnym prądu stałego lub z silnikiem prądu przemiennego (indukcyjnym lub synchronicznym z magnesami trwałymi) sterowanym metodą polowo-zorientowaną lub metodą DTC. Część mechaniczna napędu z połączeniem sprężystym reprezentowana jest przez układ dwumasowy, w którym silnik reprezentuje pierwszą masę skupioną, a obciążenie drugą. Taką strukturę sterowania, z neuronowym regulatorem prędkości silnika napędowego przedstawiono na rys. 1. Na tym rysunku zastosowano następujące oznaczenia: ω_1 , ω_2 – prędkość silnika napędowego i maszyny obciążającej, m_e , m_s , m_L , m_{ref} – moment elektromagnetyczny, moment skrętny, moment obciążenia i moment zadany na wyjściu regulatora prędkości, T_1 , T_2 – mechaniczna stała czasowa silnika i maszyny roboczej, T_c – stała czasowa połączenia sprężystego [10].



Rys. 1. Schemat blokowy uproszczonego modelu układu napędowego z dynamicznym sterowaniem momentu elektromagnetycznego

W pętli regulacji prędkości kątowej, zamiast klasycznego regulatora PI, zastosowano adaptacyjny regulator neuronowy, którego zadaniem jest tłumienie drgań skrętnych układu z połączeniem elastycznym. Porównano właściwości napędu w przypadku zastosowania regulatora wykorzystującego sieć perceptronową MLP uczoną *on-line* za pomocą algorytmu wstecznej propagacji błędów oraz regulatora z siecią o radialnych funkcjach bazowych RBF. Wagi struktur neuronowych są aktualizowane w każdym kroku obliczeniowym struktury sterowania napędem z połączeniem elastycznym, tylko na podstawie pomiaru prędkości silnika napędowego. Ma to istotne znaczenie, gdyż nie wymaga odtwarzania innych zmiennych stanu napędu, jak na przykład momentu skrętnego czy prędkości obciążenia [13], jak to ma miejsce w przypadku struktur z regulatorami PI, PID i dodatkowymi sprzężeniami zwrotnymi [14], lub z regulatorem stanu [15].

Wejściem neuroregulatora może być błąd liczony jako różnica pomiędzy wartością zadaną i prędkością silnika lub różnica pomiędzy wyjściem modelu odniesienia (na wejście którego wprowadzana jest wartość zadana prędkości) i prędkością silnika. W takim przypadku model odniesienia można opisać następująco:

$$G_r(s) = \frac{\omega_0^2}{s^2 + 2\zeta\omega_0s + \omega_0^2}, \quad (1)$$

gdzie:

- ζ – założony współczynnik tłumienia,
- ω_0 – zadana częstotliwość rezonansowa,
- s – operator Laplace'a.

W układach sterowania oraz estymacji zmiennych stanu przy wykorzystaniu modeli neuronowych, w celu tłumienia sygnałów dynamicznie zmieniających się w czasie, zakłóceń i szumów, często wykorzystywane są filtry dolnoprzepustowe (człony inercyjne pierwszego rzędu). Jednak w opisywanym przypadku zastosowano transmi-tancję opisaną wzorem (1), ze względu na możliwość kształtowania dynamiki sygnału wejściowego dla regulatora adaptacyjnego. We wszystkich analizowanych regulatorach wykorzystano identyczne wartości: $\omega_0 = 20 \text{ s}^{-1}$ oraz $\zeta = 1$.

W niniejszej pracy analizowane są dwa rodzaje neuroregulatorów prędkości układu dwumasowego, bazujące na sieci perceptronowej MLP i sieci o radialnych funkcjach bazowych RBF. W następnych częściach artykułu przedstawiono strukturę obu neuroregulatorów oraz podano algorytmy adaptacji proponowanych struktur neuronowych.

3. BADANE STRUKTURY NEUROREGULATORÓW

3.1. REGULATOR MLP Z ALGORYTMEM WSTECZNEJ PROPAGACJI

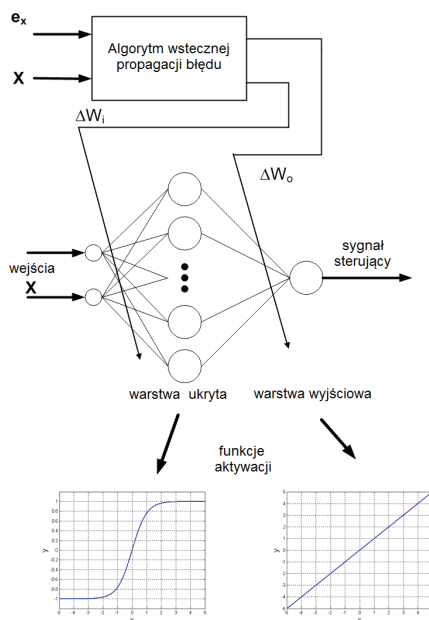
Analizowany w niniejszej części pracy regulator wykorzystuje perceptronową sieć MLP. Zastosowany model neuronowy zawiera dwa wejścia, kilka neuronów w warstwie ukrytej o sigmoidalnej funkcji aktywacji oraz jedno wyjście z liniową funkcją aktywacji. Strukturę sieci neuronowej MLP przedstawiono na rys. 2. Kolejne etapy obliczeń opisywanego modelu neuronowego mogą zostać przedstawione następująco:

- wejściowe sygnały \mathbf{X} są mnożone przez wartości współczynników wagowych \mathbf{W}_i ,
- wyniki są sumowane; w ten sposób uzyskiwane są sygnały wejściowe funkcji aktywacji neuronów warstwy ukrytej,
- uzyskane rezultaty na wyjściach neuronów sigmoidalnych są skalowane według wag wyjściowych \mathbf{W}_o ,
- otrzymane wartości są sumowane (zastosowany neuron liniowy na wyjściu modelu).

Wartości macierzy \mathbf{W}_i oraz \mathbf{W}_o regulatora są obliczane *on-line*. W tym celu zastosowano algorytm wstecznej propagacji błędów (*backpropagation*) [11]. Metoda w rzeczywistości jest jedną z technik gradientowych, przetwarzanie *backpropagation* jest zastosowane w celu uzyskania informacji o zmianach funkcji celu względem poszczególnych wag. W każdej iteracji algorytmu wagi są aktualizowane zgodnie z poniższą zależnością:

$$w_{ij}(k+1) = w_{ij}(k) + \Delta w_{ij} \quad (1)$$

gdzie w_{ij} – wagi pomiędzy i -tym oraz j -tym neuronem w k -tej iteracji.



Rys. 2. Struktura neuroregulatora wykorzystującego sieć MLP trenowaną *on-line*

W równaniu tym wartości wag są obliczane po wprowadzeniu korekty:

$$\Delta w_{ij} = \alpha \delta_j x_{ij} \quad (2)$$

gdzie:

α – współczynnik uczenia,

x_{ij} – i -te wejście j -tego neuronu,

δ_j – funkcja błędów opisanego poniżej.

Analizowany regulator posiada tylko jedną warstwę ukrytą z neuronami sigmooidalnymi i jednym wyjściem. Dla warstwy ukrytej wartość δ_j jest opisana następująco:

$$\delta_j = f'_j \delta_o w_{jo}, \quad (3)$$

gdzie:

f'_j – pochodna funkcji aktywacji j -tego neuronu w warstwie ukrytej,

w_{jo} – wagi pomiędzy wyjściem a warstwą ukrytą.

Dla warstwy wyjściowej z pojedynczym neuronem liniowym $\delta_j = \delta_o$:

$$\delta_o = d - y, \quad (4)$$

gdzie:

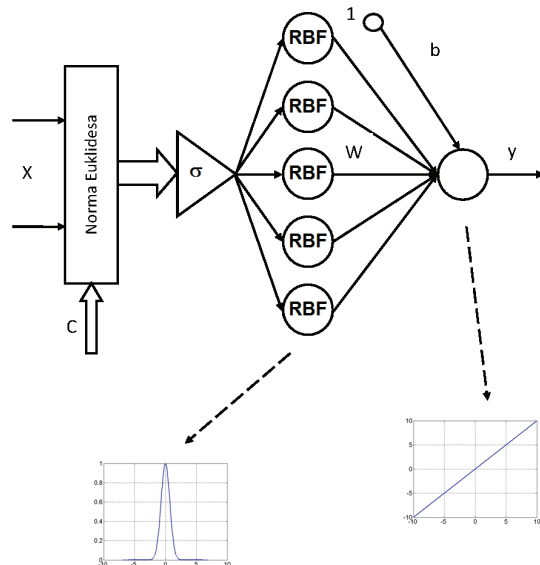
d – żądana wartość sygnału wyjściowego,

y – aktualna wartość tego sygnału.

Parametr α we wzorze (2) decyduje o kroku algorytmu zastosowanego w celu adaptacji wag modelu neuronowego. Na wyjściu sieci neuronowej otrzymywany jest sygnał sterujący dla podporządkowanego regulatora momentu (prądu).

3.2. REGULATOR ADAPTACYJNY OPARTY NA SIECI RADIALNEJ RBF

Sieć neuronowa radialna RBF różni się od klasycznej sieci MLP budową, ideą przetwarzania danych oraz metodyką uczenia. Dane wejściowe w sieci RBF są przetwarzane przez lokalne odwzorowania realizowane w neuronach radialnych, a ostateczny wynik jest uzyskiwany poprzez obliczenie ważonej sumy wartości wyjściowych z poszczególnych neuronów warstwy ukrytej. Strukturę sieci RBF przedstawiono na rysunku 3.



Rys. 3. Struktura sieci neuronowej RBF

Przetwarzanie sygnałów w sieci RBF, która jest również modelem typu *feedforward*, odbywa się według następującego schematu [11]:

- w warstwie wejściowej obliczane są odległości pomiędzy elementami wektora wejściowego a centrami neuronów radialnych (najczęściej wg normy Euklidesa),
- skalowanie za pomocą współczynnika σ ,
- w warstwie ukrytej realizowane są obliczenia wartości funkcji radialnych na podstawie argumentów z wcześniejszych obliczeń (neurony ukryte nie posiadają połączeń z sygnałami zewnętrznymi),
- w warstwie wyjściowej, zawierającej neuron liniowy, wartości uzyskane na wyjściu neuronów radialnych są mnożone poprzez współczynniki wagowe, a następnie sumowane.

Macierz radialnych funkcji aktywacji zastosowanych w neuronach warstwy ukrytej najczęściej wyznaczana jest z wykorzystaniem funkcji Gaussa:

$$\mathbf{f}_g(\mathbf{X}, \mathbf{C}, \sigma) = \exp\left(-\frac{\mathbf{v}^2}{\sigma^2}\right), \quad (5)$$

gdzie:

- σ – parametr skalujący kształtujący funkcję radialną,
- \mathbf{v} – różnica między centrum neuronu a wartościami wejściowymi,
- \mathbf{X} – wektor wejściowy sieci neuronowej,
- \mathbf{C} – macierz zawierająca centra sieci radialnej.

Najczęściej na tym etapie obliczeń wykorzystuje się następującą zależność (norma Euklidesa):

$$\mathbf{v}(\mathbf{X}) = \|\mathbf{X} - \mathbf{C}\| = \sqrt{\sum_{k=1}^n (x_k - c)^2}, \quad k = 1, 2, 3, \dots, n \quad (6)$$

gdzie:

- x_k – k -ty element wejściowego wektora \mathbf{X} ,
- c – centrum z macierzy centrów sieci neuronowej.

Obliczenia warstwy wyjściowej opisane są równaniem:

$$y_i = b + \sum_{h=1}^h w_{hi} f_{g_h} \quad (7)$$

gdzie:

- h – liczba neuronów ukrytych,
- y_i – aktualna wartość i -tego wyjścia sieci neuronowej,
- w_{hi} – współczynnik wagowy pomiędzy h -tym neuronem ukrytym RBF oraz i -tym wyjściem,
- b – bias.

W związku z powyższym wartość wyjściowa sieci neuronowej w zapisie macierzowym może być przedstawiona następująco:

$$y = b + \mathbf{W}\mathbf{f}_g(\mathbf{X}, \mathbf{C}, \sigma). \quad (8)$$

W sieci neuronowej typu RBF stosowanej jako neuroregulator, wagi i centra neuronów radialnych są adaptowane jednocześnie. Do tego celu można zastosować algorytm gradientowy, minimalizujący funkcję celu [11], [15]:

$$E(\mathbf{W}) = \frac{1}{2} \sum_{j=1}^M [e(\mathbf{W}_j)]^2 = \frac{1}{2} \sum_{j=1}^M (d_j - y_j)^2, \quad (9)$$

gdzie:

j – liczba wyjść sieci,

e – błąd pomiędzy wartością zadaną wyjścia i jej wartością aktualną.

W przypadku funkcji opisanej zależnością (8), równanie (9) przyjmuje postać:

$$E = \frac{1}{2} (e(\mathbf{W}))^2 = \frac{1}{2} [d - (b + \mathbf{W}\mathbf{f}_g(\mathbf{X}, \mathbf{C}, \sigma))]^2. \quad (10)$$

Gradient funkcji celu względem wag sieci przyjmuje następującą wartość:

$$\frac{\partial E}{\partial \mathbf{W}} = \frac{1}{2} \left(2e \frac{\partial e}{\partial \mathbf{W}} \right), \quad (11)$$

gdzie pochodne cząstkowe są wyznaczone według następującego równania:

$$\frac{\partial e}{\partial \mathbf{W}} = -\mathbf{f}_g(\mathbf{X}, \mathbf{C}, \sigma). \quad (12)$$

Wobec tego wartość gradientu (11) przyjmie postać:

$$\frac{\partial E}{\partial \mathbf{W}} = -[d - (b + \mathbf{W}\mathbf{f}_g(\mathbf{X}, \mathbf{C}, \sigma))] \mathbf{f}_g(\mathbf{X}, \mathbf{C}, \sigma) = -e \mathbf{f}_g(\mathbf{X}, \mathbf{C}, \sigma). \quad (13)$$

Tak więc aktualizacja wag w warstwie wyjściowej regulatora RBF w każdej iteracji algorytmu adaptacji odbywa się według następującej formuły:

$$\mathbf{W}(k) = \mathbf{W}(k-1) - \eta_w \frac{\partial E}{\partial \mathbf{W}}. \quad (14)$$

Podstawiając wartość gradientu (13) w równaniu (14), otrzymuje się:

$$\mathbf{W}(k) = \mathbf{W}(k-1) + \eta_w e \mathbf{f}_g(\mathbf{X}, \mathbf{C}, \sigma). \quad (15)$$

gdzie η_w – współczynnik adaptacji wag sieci RBF.

W podobny sposób można otrzymać algorytm adaptacji centrów regulatora RBF. Dal funkcji kosztu (10), wartość gradientu względem centrów sieci RBF przyjmuje postać:

$$\frac{\partial E}{\partial \mathbf{C}} = \frac{1}{2} \left(2e \frac{\partial e}{\partial \mathbf{C}} \right), \quad (16)$$

gdzie:

$$\frac{\partial e}{\partial \mathbf{C}} = -\mathbf{W} \frac{(\mathbf{X} - \mathbf{C})}{\sigma^2} \mathbf{f}_g(\mathbf{X}, \mathbf{C}, \sigma), \quad (17)$$

i wówczas:

$$\frac{\partial E}{\partial \mathbf{C}} = -e \mathbf{W} \frac{(\mathbf{X} - \mathbf{C})}{\sigma^2} \mathbf{f}_g(\mathbf{X}, \mathbf{C}, \sigma). \quad (18)$$

Zgodnie z algorytmem gradientowym, adaptacja centrów sieci RBF odbywa się zgodnie z zależnością:

$$\mathbf{C}(k) = \mathbf{C}(k-1) - \eta_c \frac{\partial E}{\partial \mathbf{C}}. \quad (19)$$

Po uwzględnieniu (18) w (19) otrzymuje się algorytm adaptacji centrów regulatora RBF:

$$\mathbf{C}(k) = \mathbf{C}(k-1) + \eta_c e \mathbf{W} \frac{\mathbf{X} - \mathbf{C}}{\sigma^2} \mathbf{f}_g(\mathbf{X}, \mathbf{C}, \sigma). \quad (20)$$

gdzie η_c – współczynnik adaptacji centrów sieci RBF.

Równania (15) i (20) opisują obliczenia macierzy wag oraz centrów w trakcie procesu treningu sieci neuronowej, który w prezentowanej aplikacji jest realizowany *on-line*. Współczynniki η_w i η_c są stałymi determinującymi szybkość działania algorytmu strojenia neuroregulatora. W drugiej części [17] artykułu zostaną przedstawione wyniki badań symulacyjnych i eksperymentalnych opisanych neuroregulatorów w strukturze sterowania prędkością napędu z połączeniem elastycznym ocena ich skuteczności w tłumieniu drgań skrętnych układu dwumasowego w przypadku zastosowania tylko podstawowego sprzężenia zwrotnego od prędkości silnika napędowego.

4. PODSUMOWANIE

Sieci neuronowe z jedną warstwą ukrytą o nieliniowych funkcjach aktywacji oraz jednym liniowym neuronem wyjściowym mogą być stosowane jako adaptacyjne neuroregulatory sterowane funkcją błędu układu napędowego z połączeniem sprzężystym.

Istotną cechą odróżniającą proponowane struktury sterowania od powszechnie stosowanych struktur z dodatkowymi sprzężeniami zwrotnymi i regulatorami liniowymi stosowanymi w celu tłumienia drgań skrętnych i oscylacji prędkości kątowej obciążenia [14] jest to, że w analizowanych układach z adaptacyjnymi neuroregulatorami proponuje się jedynie sprzężenie zwrotne od łatwomierzalnej prędkości silnika napędowego. Nie wymaga to stosowania estymatorów prędkości maszyny roboczej, momentu skrętnego i momentu obciążenia. Jak zostanie wykazane w drugiej części niniejszego artykułu [17], proponowane struktury sterowania z nieliniowymi neuroregulatorami zapewniają bardzo dobre właściwości dynamiczne w różnych warunkach pracy napędu.

Praca naukowa finansowana ze środków Narodowego Centrum Nauki w ramach projektu UMO-2011/01/B/ST7/04632.

LITERATURA

- [1] BOSE B.K., *Neural Network Applications in Power Electronics and Motor Drives – An Introduction and Perspective*, IEEE Transactions on Industrial Electronics, 2007, Vol. 54, No. 1, 14–33.
- [2] CHEN T.C., SHEU T.T., *Model Reference Neural Network Controller for Induction Motor Speed Control*, IEEE Trans. Energy Conversion, 2002, Vol. 17, No. 2, 157–162.
- [3] CHENG Q., WANG Y., CHENG Y., HU X., XUE Y., *The Application of Neural Network Model Reference Adaptive Control in Wind Turbines Variable Pitch System*, Journal of Computational Information Systems, 2011, Vol. 7, No. 10, 3422–3429.
- [4] CHEN D., YORK M., *Adaptive Neural Inverse Control Applied to Power Systems*, IEEE Power Systems Conf. and Exposition, 2006, 2109–2115.
- [5] COLINA-MORLES E., MORT N., *Inverse Model Neural Network – Based Control of Dynamic Systems*, IEE Control-Conf. Publ., 1994, Vol. 2, 955–960.
- [6] BOSE B.K., CHIO K.M., KIM H.J., *Self Tunning Neural Network Controller for Induction Motor Drives*, IEEE Annual Conf. of the IEEE Industrial Electronics Society, 2002, Vol.1, 152–156.
- [7] CHEN T.C., SHEU T.T., *Model reference robust speed control for induction-motor drive with time delay based on neural network*, IEEE Trans. Systems, Man and Cybernetics, 2001, Vol. 31, No. 6, 746–753.
- [8] KUSWSKI J.G., ENGELBRECHT R., HUI S., ŻAK S.H., *Application of Adaline to the Synthesis of Adaptive Controllers for Dynamical Systems*, American Control Confer., 1991, 1273–1278.
- [9] KAMIŃSKI M., Dyrca K., *Zastosowanie modeli ADALINE w strukturze sterowania prędkością silnika indukcyjnego*, Prace Naukowe IMNiPE, 2011, Nr 65, Ser.: Studia i Materiały, Nr 31, 192–199.
- [10] KAMINSKI M., ORŁOWSKA-KOWALSKA T., *FPGA Implementation of ADALINE-based Speed Controller for Two-mass System*, IEEE Trans. on Industrial Informatics, 2013, Vol. 9, No. 3, 1301–1311.
- [11] KAMIŃSKI M., ORŁOWSKA-KOWALSKA T., *Adaptive neural speed control of the induction motor drive*, Przegląd Elektrotechniczny, 2013, Vol. 89, Nr 2A, 21–24.
- [12] BISHOP M.C., *Neural Networks for Pattern Recognition*, Oxford University Press, 1996.
- [13] ORŁOWSKA-KOWALSKA T., KAMIŃSKI M., SZABAT K., *Mechanical state variable estimation of drive system with elastic coupling using optimized feed-forward neural networks*, Bulletin of the Polish Academy of Sciences: Technical Sciences, 2008, Vol. 56, No. 3, 239–246.

- [14] SZABAT K., OROWSKA-KOWALSKA T., *Vibration suppression in two-mass drive system using PI speed controller and additional feedbacks – comparative study*, IEEE Trans. Industrial Electronics, 2007, Vol. 54, No. 2, 1193–1206.
- [15] KAMIŃSKI M., ORŁOWSKA-KOWALSKA T., SZABAT K., *Analysis of the dynamical performance of the two-mass drive system with the modified state controller*, Computer Applications in Electrical Engineering, 2010, Vol. 8, (ed. by R. Nawrowski), Poznań, POLI-GRAF-JAK, 105–119.
- [16] NARENDRA K.S., PARTHASARATHY K., *Gradient Methods for the Optimization of Dynamical Systems Containing Neural Networks*, IEEE Trans. on Neural Networks, 1991, Vol. 2, No. 2, 252–262.
- [17] KAMIŃSKI M., ORŁOWSKA-KOWALSKA T., *Analiza porównawcza wybranych neuroregulatorów dla napędu z połączeniem sprężystym – wyniki badań*, Prace Naukowe IMNiPE, 2013, Nr 69, Ser.: Studia i Materiały, Nr 33.

COMPARATIVE ANALYZIS OF CHOSEN NEUROCONTROLLERS FOR THE DRIVE SYSTEM WITH ELASTIC JOINT – DESCRIPTION OF TESTED MODELS

In the paper the adaptive control problems of the drive system with elastic joint using neural models trained *on-line* have been presented. Perceptron-type MLP neural networks and networks with radial basis functions RBF have been proposed, which were trained with Back-Propagation Algorithm based on gradient algorithms. The structures of neurocontrollers and analytical expressions for weights and centers adaptation of the suitable neurocontrollers' structures have been presented.