

Robert POGORZELSKI*

PORÓWNANIE WŁAŚCIWOŚCI NEURONOWYCH I KLASYCZNYCH UKŁADÓW STEROWANIA NIELINIOWYM PROCESEM DYNAMICZNYM

W pracy zaprezentowano wybrane architektury neuronowych układów sterowania nieliniowym obiektem dynamicznym i porównano ich właściwości. Sterowanie procesem zostało zrealizowane za pomocą trzech algorytmów, wykorzystujących sztuczne sieci neuronowe: metody stosującej neuronowy model odwrotnej dynamiki obiektu, metody działającej w oparciu o linearyzację przez sprzężenie zwrotne oraz metody opartej o algorytm przeprowadzający na bieżąco linearyzację nieliniowego, neuronowego modelu obiektu. W sterowaniu wykorzystano sieci perceptronowe typu MLP (ang. Multilayer Perceptron). Dobór wag sieci przeprowadzono z wykorzystaniem algorytmu Levenberga–Marquardta. Zaproponowane metody sterowania zostały porównane z układem regulacji PID.

SŁOWA KLUCZOWE: sieć neuronowa, neuronowy układ sterowania

1. WSTĘP

Przemysł dążąc do optymalizacji procesów produkcyjnych, szuka nowych rozwiązań, które pozwoliłyby na lepsze wykorzystanie surowców i czasu. Zwiększenie wydajności układów można zrealizować przy zastosowaniu nowych i niestandardowych algorytmów sterowania.

Stosowane powszechnie klasyczne regulatory liniowe zapewniają poprawne działanie układu regulacji w określonym otoczeniu punktu pracy, dla którego były wyznaczane nastawy regulatora. Występujące w praktyce obiekty sterowania są jednakże z reguły nieliniowe. W przypadku procesów nieliniowych naturalnym rozwiązaniem jest zastosowanie nieliniowych algorytmów regulacji predykcyjnej, w których do predykcji stosuje się modele nieliniowe [2]. Dlatego przy procesach o których zachowaniu wiadomo niewiele, warto wykorzystać wielowarstwowe sztuczne sieci neuronowe, które umożliwiają realizację praktycznie dowolnych nieliniowych odwzorowań [10]. Ponadto dostępnych jest wiele efektywnych algorytmów uczenia modeli neuronowych i doboru ich struktury. W przeciwieństwie do modeli fizycznych,

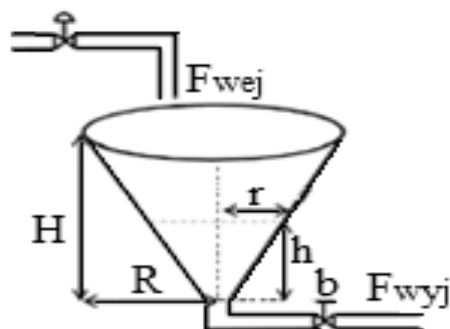
*Politechnika Białostocka.

modele neuronowe nie zawierają żadnych równań różniczkowych, które należy cyklicznie rozwiązywać w algorytmie regulacji predykcyjnej. W związku z tym sieci neuronowe zostały zastosowane w wielu dziedzinach, przede wszystkim w przemyśle chemicznym, petrochemicznym i przetwórczym [3].

W niniejszej pracy wykorzystano sztuczne sieci neuronowe do sterowania procesem w dwojaki sposób: bezpośredni i pośredni. W pierwszym podejściu sieć neuronowa pełniła rolę sterownika, który wyznaczał sygnały sterujące działające na obiekt. Sterowanie pośrednie opierało się na wykorzystaniu zlinearyzowanego neuronowego modelu sterowanego obiektu w doborze parametrów regulatora RST.

2. MODEL PRZEPLYWU W STOŻKOWYM ZBIORNIKU

Schemat budowy stożkowego zbiornika cieczy przedstawiono na rys. 1. Rozpatrywany układ sterowania ma za zadanie utrzymywać poziom płynu h w zbiorniku na zadanym poziomie. Układ sterowany jest poprzez zmianę dopływu płynu do zbiornika F_{wej} .



Rys. 1. Stożkowy przepływowy zbiornik cieczy

Model matematyczny przepływu cieczy w zbiorniku dany jest równaniem [1]:

$$\frac{dh}{dt} = \frac{H^2}{(3\pi R^2 h^2)(F_{wej} - b\sqrt{h})} \quad (1)$$

gdzie: h – poziom płynu w zbiorniku (m), R – średnica zbiornika (m), H - wysokość zbiornika (m), b – współczynnik zależny od parametrów zaworu odpływowego ($m^{5/2}/s$), F_{wej} – przepływ cieczy na wejściu (m^3/s).

Na podstawie modelu matematycznego (1) zbudowano w środowisku Matlab/Simulink, odpowiadający mu model symulacyjny. Przyjęte zostały następujące parametry modelu: $R = 0,1$ m, $H = 0,6$ m, $b = 4 \cdot 10^{-4}$ $m^{5/2}/s$.

Do modelowania odwrotnej dynamiki stożkowego zbiornika wykorzystano dwuwarstwową sieć neuronową. W warstwie ukrytej zastosowano neurony z sigmoidalną funkcją aktywacji, a w warstwie wyjściowej z liniową funkcją aktywacji. Zbiór danych treningowych (2000 próbek) został uzyskany w wyniku symulacji modelu w układzie otwartym. Sygnał F_{wej} zmieniał się w granicach $2\cdot 4\cdot 10^{-4}$ m³/s, a poziom h zmieniał się między 0,35-0,51 m. Proces uczenia sieci polegał na podaniu na jej wejście historycznych wartości poziomu płynu w zbiorniku h , a na wyjście sygnału zadanego F_{wej} . Sieć była trenowana z wykorzystaniem algorytmu gradientowego Levenberga-Marquardta [8], a wagi sieci były obliczane na podstawie minimalizacji funkcji celu:

$$J(\theta, Z^N) = \frac{1}{2N} \sum_{k=1}^N [u(k) - \hat{u}(k|\theta)]^2 \quad (4)$$

Przetestowano kilka różnych modeli, różniących się rzędem dynamiki (liczbą historycznych próbek sygnału wejściowego i wyjściowego) oraz liczbą ukrytych neuronów. Dla każdej konfiguracji sieci, uczenie było powtarzane 10 razy, a początkowe wartości wag były wybierane losowo. Eksperymentalnie stwierdzono, że model powinien mieć dynamikę drugiego rzędu:

$$\hat{u}(k) = f^{-1}[y(k+1), y(k), y(k-1), y(k-2), u(k-1), u(k-2), \theta_2] \quad (5)$$

z 6 neuronami ukrytymi i 1 neuronem na wyjściu. Tak zbudowaną sieć przetestowano w sterowaniu poziomem płynu w zbiorniku (rys. 5).

5. LINEARYZACJA PRZEZ SPRZĘŻENIE ZWROTNE

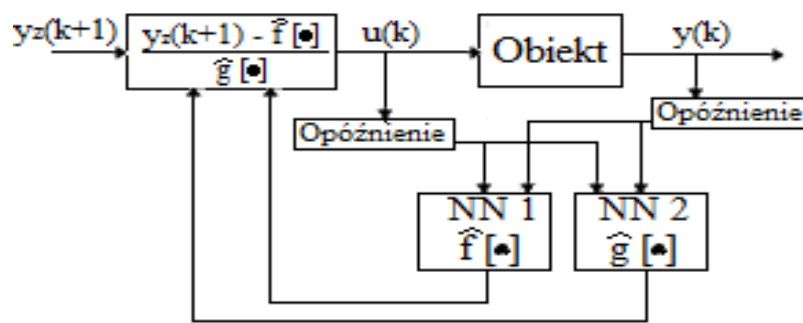
Główną ideą tego typu systemu sterowania jest skompensowanie nieliniowości obiektu sterowania. Pozbycie się nieliniowości poprzez zastosowanie sprzężenia zwrotnego umożliwia rozpatrywanie stworzonego układu regulacji jako układu liniowego. W metodzie tej model obiektu przybliża się za pomocą struktury NARMA – L2 [5]:

$$\begin{aligned} \hat{y}(k+1|\theta) = & \hat{f}[y(k), \dots, y(k-n+1), u(k-1), \dots, u(k-m+1), w] + \\ & + \hat{g}[y(k), \dots, y(k-n+1), u(k-1), \dots, u(k-m+1), v]u(k) \end{aligned} \quad (6)$$

Jest on parametryzowany przez dwie sieci neuronowe: \hat{f}, \hat{g} . Parametry (w, v) reprezentują wagi sieci neuronowych. Zastosowanie tego typu modelu sprawia, że sygnał sterujący $u(k)$ nie jest zawarty w nieliniowej części układu. Celem sterowania jest nadążanie próbki sygnału wyjściowego $y(k+1)$ za próbką sygnału zadanego $y_z(k+1)$, to znaczy $y(k+1) = y_z(k+1)$. Przekształcając równanie (6) sygnał sterujący ma postać:

$$u(k+1) = \frac{(y|z(k+1) - \hat{f}[y(k), \dots, y(k-n+1), u(k), \dots, u(k-m+1), w])}{\hat{g}[(y(k), \dots, y(k-n+1), u(k), \dots, u(k-m+1), v])]} \quad (7)$$

Schemat działania układu zaprezentowano na rys. 3.



Rys. 3. Schemat sterowania metodą linearyzacji przez sprzężenie zwrotne

Algorytm doboru struktury sieci neuronowych był podobny jak w poprzednim punkcie. Wyniki badań eksperymentalnych wskazywały, że model opisany przez sieć neuronową \hat{f} z 5 neuronami ukrytymi o tangensoidalnej funkcji aktywacji i sieć \hat{g} z 4 neuronami w warstwie ukrytej najlepiej spisywał się przy sterowaniu poziomem płynu w zbiorniku. Na wyjściu obu sieci znajdował się jeden neuron liniowy, a liczba przeszłych wyjść n i wejść m wynosiła po 2. Na rys. 5 przedstawiono rezultaty symulacji.

6. SUKCESYWNA LINEARYZACJA W PUNKCIE PRACY

Innym podejściem do wykorzystania sieci neuronowych w sterowaniu jest bazowanie na neuronowym modelu obiektu. Cechą szczególną tego typu suboptymalnego algorytmu jest cykliczna, bieżąca linearyzacja modelu nieliniowego w aktualnym punkcie pracy [6]. Uzyskane przybliżenie liniowe modelu jest następnie stosowane do syntezy układu z regulatorem RST [9].

Model NARX procesu można opisać za pomocą równania (2) przesuwając próbki o jeden krok wstecz. Funkcja f jest linearyzowana w czasie $k = \tau$ w aktualnym punkcie pracy $\varphi(\tau)$, dzięki temu uzyskujemy model przybliżony w postaci:

$$\hat{y}(k) = [1 - A(q^{-1})]y(k) + q^{-d}B(q^{-1})u(k) + \varepsilon(\tau) \quad (8)$$

gdzie wektor szumu $\varepsilon(\tau)$ dany jest równaniem:

$$\varepsilon(\tau) = y(\tau) + a_1 y(\tau - 1) + \dots + a_n y(\tau - n) - b_0 u(\tau - 1) - \dots - b_m u(\tau - m) \quad (9)$$

W wielomianach $A(q^{-1})$ i $B(q^{-1})$ zgromadzone są współczynniki modelu zlinearyzowanego $\{a_i\}$ oraz $\{b_i\}$:

$$A(q^{-1}) = 1 + a_1 q^{-1} + \dots + a_n q^{-n} \quad (10)$$

$$B(q^{-1}) = b_0 + b_1 q^{-1} + \dots + b_m q^{-m}$$

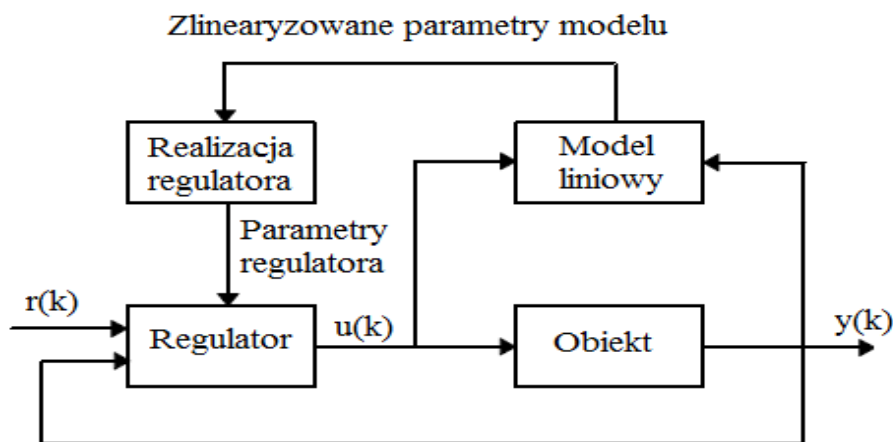
które oblicza się z analitycznych wzorów:

$$a_i = \frac{\partial f[\varphi(k)]}{\partial y(k-i)} \Big|_{\varphi(k)=\varphi(\tau)} \quad (11)$$

oraz

$$b_i = \frac{\partial f[\varphi(k)]}{\partial u(k-1-i)} \Big|_{\varphi(k)=\varphi(\tau)}$$

Dzięki takiemu zapisowi aproksymowany model może być interpretowany jako liniowy model ARX z zakłóceniami $\varepsilon(\tau)$ w aktualnym punkcie pracy. Przedstawiony system zaprezentowany jest na rys. 4.



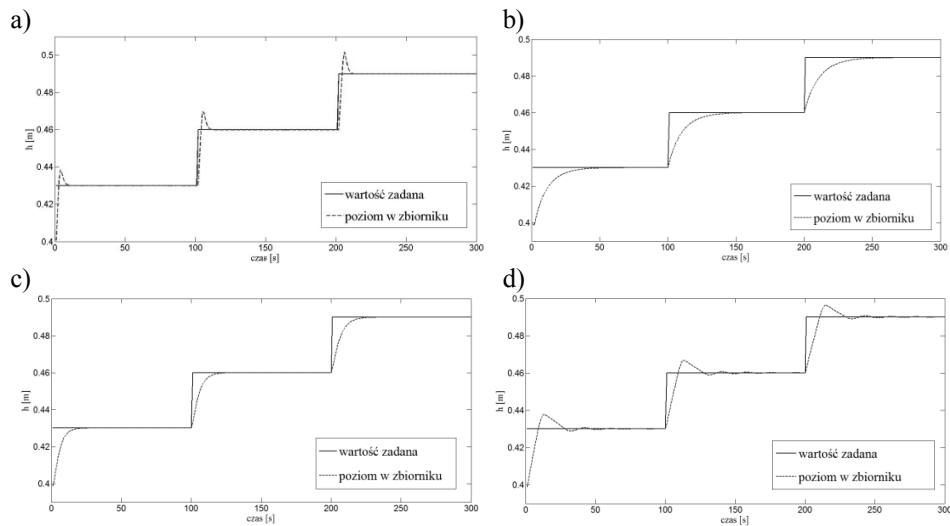
Rys. 4. Schemat sterowania z linearyzacją w punkcie pracy neuronowego modelu obiektu

Obiekt zamodelowany przy pomocy modelu NARX został zrealizowany za pomocą sieci o sigmoidalnych funkcjach aktywacji w warstwie ukrytej i liniowej funkcji aktywacji w warstwie wyjściowej. Najlepszą dokładność modelu uzyskano przy sieci o 6 neuronach ukrytych i 1 neuronie wyjściowym. Liczby przeszłych wyjść i wejść wynosiły: $n = 2$ i $m = 2$. Rezultaty sterowania zaprezentowano na rys. 5.

7. REZULTATY

Każdy algorytm regulacji został poddany trzem różnym skokowym wymuszeniom, zmieniającym wartość zadaną odpowiednio do: $h = 0,43, 0,46, 0,49$ m. Przyjęto, że w chwili $t = 0$, w układzie panowały warunki początkowe $h_0 = 0,4$ m. Sygnał sterujący został objęty ograniczeniami, i mógł przyjmować wartości $0-2 \cdot 10^{-3}$ m³/s. Porównanie wszystkich przebiegów sterowania zaprezentowano poniżej.

Na podstawie przeprowadzonych badań, widać, że żadna z metod nie daje jednoznacznie lepszych rezultatów. Najkrótszy czas regulacji uzyskano stosując regulację PID, natomiast w metodach stosujących neuronowy model odwrotnej dynamiki oraz linearyzacji przez sprzężenie zwrotne, przeregulowanie było równe 0. W metodzie sukcesywnej linearyzacji występuje przeregulowanie, które maleje wraz ze wzrostem wartości zadanej i w każdym punkcie pracy jest mniejsze niż w metodzie regulacji PID. Można zauważyć, że czas regulacji z regulatorem PID wzrasta wraz ze wzrostem wartości zadanej. W neuronowych układach sterowania czasy te są podobne w każdym regionie i nie rosną wraz ze wzrostem wartości zadanej. Dla każdego układu regulacji zaobserwowano brak uchybu ustalonego. W tabeli 1 zebrano wszystkie wskaźniki jakości regulacji, mierzone w 3 rozpatrywanych punktach pracy. Czas regulacji był mierzony z tolerancją $\Delta = 2\%$.



Rys. 5. Wyniki symulacji układu regulacji poziomu płynu w zbiorniku a) metodą PID, b) metodą z wykorzystaniem neuronowego modelu odwrotnej dynamiki, c) metodą linearyzacji przez sprzężenie zwrotne, d) metodą sukcesywnej linearyzacji w punkcie pracy przy skokowej zmianie wartości zadanej

Tabela. 1. Wskaźniki jakości regulacji

| Wskaźnik jakości regulacji | PID | | | Neuronowy modelu odwrotnej dynamiki | | | Linearyzacja przez sprzężenie zwrotne | | | Sukcesywna linearyzacja w punkcie pracy | | |
|----------------------------|--------|-----|-----|-------------------------------------|----|----|---------------------------------------|-----|---|---|-----|-----|
| | Region | | | Region | | | Region | | | Region | | |
| | 1 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 3 |
| Przeregulowanie [%] | 2,1 | 2,2 | 2,4 | 0 | 0 | 0 | 0 | 0 | 0 | 1,8 | 1,5 | 1,3 |
| Czas regulacji [s] | 4,1 | 6,1 | 6,7 | 14 | 12 | 12 | 7,3 | 6,5 | 7 | 6,8 | 6,4 | 7 |
| Całka z modułu uchybu | 3,89 | | | 9,36 | | | 8,32 | | | 8,12 | | |

8. PODSUMOWANIE

W pracy przedstawiono możliwości zastosowania sztucznych sieci neuronowych w układzie regulacji nieliniowego obiektu dynamicznego. Wyniki badań potwierdziły, że stosując sieci neuronowe można uzyskać dobrą jakość regulacji i że sztuczne sieci neuronowe mogą stanowić przydatne narzędzie w sterowaniu nieliniowym procesem dynamicznym.

Dalszym kierunkiem badań przewidywanym przez autora jest skupienie się nad algorytmami suboptymalnymi z linearyzacją oraz wykorzystanie ich wraz z metodą harmonogramowania wzmocnień do strojenia regulatora.

LITERATURA

- [1] Aravind P., Valluvan M., Ranganathan S., Modelling and Simulation of Non Linear Tank, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 2, Issue 2, 2013, s. 842-849.
- [2] Chowdhury F. N., Wahi P., A survey of neural networks applications in automatic control, IEEE Southeastern Symposium on System Theory, Athens, 2001, s. 349-353.
- [3] Ding-Li Yu, Ding-Wen Yu, Barry Gomm J., Neural Model Adaptation and Predictive Control of a Chemical Process Rig, IEEE Transaction on Control Systems, Volume: 14, Issue: 5, September 2006, s. 828 – 840.
- [4] Haider A. F. Almurib, Ahmad A. Mat Isa, Artificial Neural Networks - Industrial and Control Engineering Applications, Chapter 18: Direct Neural Network Control via Inverse Modelling: Application on Induction Motors, InTech, 2011.
- [5] Kumpati S. Narendra, Mukhopadhyay S., Adaptive Control Using Neural Networks and Approximate Models, IEEE Transaction on Neural Networks, Volume: 8, Issue: 3, May 1997, s. 475 - 485.

- [6] Ławryńczuk M., Sieci neuronowe w efektywnych obliczeniowo algorytmach regulacji predykcyjnej, Prace Naukowe Politechniki Warszawskiej, Elektronika, nr 180, Oficyna Wydawnicza PW, Warszawa, 2012.
- [7] O'Dwyer A., Handbook of PI and PID controller tuning rules. 3rd Edition, Imperial College Press, London, 2009.
- [8] Osowski S., Sieci neuronowe w ujęciu algorytmicznym, WNT, Warszawa, 1996.
- [9] Norgard M., Ravn O., Poulsen N. K., Neural Networks for Modeling and Control of Dynamic System, Springer, 2000.
- [10] Żurada J., Barski M., Jędruch W., Sztuczne sieci neuronowe, Wydawnictwo Naukowe PWN, Warszawa, 1996.



KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI



Podlaskie

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY



Projekt współfinansowany ze środków Europejskiego Funduszu Społecznego.

COMPARISON OF NEURAL AND TRADITIONAL CONTROL OF A NONLINEAR PROCESS

The paper presents a few neural control systems to control nonlinear process and compares their properties. Control of the process was carried out by three algorithms based on neural networks: direct inverse, feedback linearization, and instantaneous linearization. The Multilayer perceptron neural network (MLP) is chosen to represent a model of a nonlinear process. To find the best weights of an MLP, the Levenberg-Marquardt method was used. Presented neural methods were compared with traditional PID control. Research has been done in the Matlab/Simulink. The test results show that artificial neural networks can be a useful tool to control a nonlinear process.