



POLITECHNIKA WROCŁAWSKA

SIECI NEURONOWE I NEUROSTEROWNIKI

Sprawozdanie z projektu

Michał Leś, Jędrzej Kozal

prowadzący
Dr inż. Piotr CISKOWSKI

2017-10-01

1 Wstęp

Podstawowym założeniem projektu jest zaznajomienie się z neurosterownikami, ich zasadami działania oraz przygotowanie modelu predykcyjnego prostego obiektu dynamicznego. Dodatkowo przyjęto że bardziej interesujące będzie przyjęcie jakiegoś rzeczywistego obiektu niż badanie reakcji bardziej teoretycznego modelu o przyjętych charakterystykach.

Neurosterowniki są sposobem zaadresowania w Automatyce kwestii sterowania obiektami mocno nieliniowymi, z którymi tradycyjne metody sterowania jak sterowniki PID nie dają sobie rady. U podstaw działania neurosterowników leży idea działania sieci neuronowej, które ostatnio zdominowały pole uczenia maszynowego. Są one wykorzystywane w wielu dziedzinach nauki i techniki do rozpoznawania obrazów (computer vision), klasyfikacji, sterownia ruchem ulicznym, wspomagania użytkowników różnych aplikacji (jak np. podpowiadanie słów w trakcie pisania na smartphonie), czy nawet prowadzenie eksperymentów społecznych. W niniejszej pracy postawiono zbadać w jaki sposób szerokie możliwości oferowane przez sieci neuronowe mogą być wykorzystane w automatyce.

2 Podstawy teoretyczne

2.1 Przyjęty model obiektu

Obiekt w automatyce jest traktowany jako czarna skrzynka, która posiada wejścia i wyjścia. Na pobudzenie na wejściu reaguje odpowiedzią na wyjściu. Odpowiedź systemów dynamicznych zależy nie tylko od aktualnej wartości wejścia, ale także stanu obiektu w poprzednich chwilach. Zgodnie z tym można zapisać model nieliniowego dyskretnego obiektu dynamicznego jako:

$$\begin{cases} \mathbf{x}(k+1) &= \phi(\mathbf{x}(k), \mathbf{u}(k)) \\ \mathbf{y}(k) &= \psi(\mathbf{x}(k)) \end{cases} \quad (1)$$

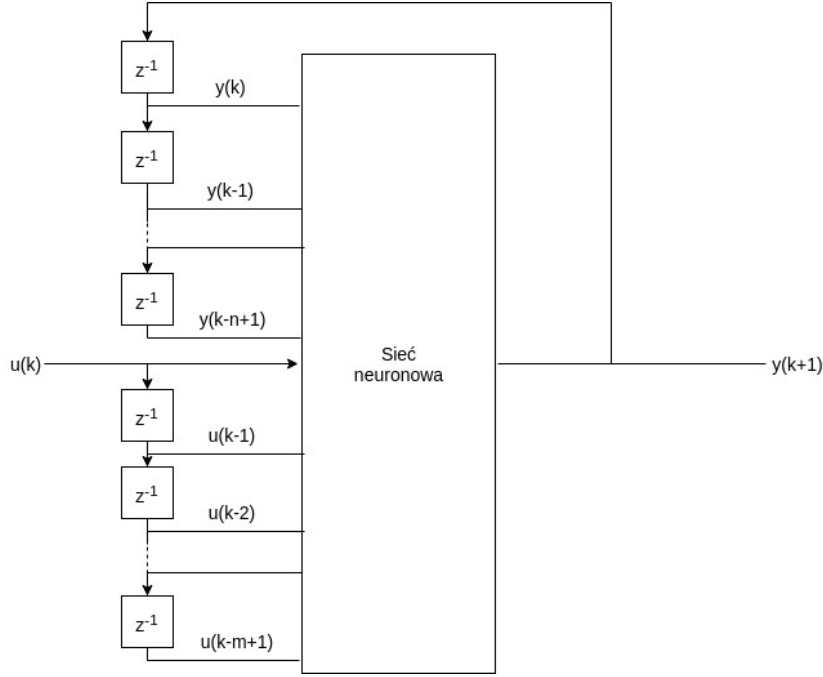
Układ 1 przedstawia równania stanu. Pierwsze równanie wiąże wewnętrzny stan obiektu z pobudzeniem, a drugie równanie wiąże stan obiektu z wyjściem. System opisywany tymi równaniami jest niezmienny w czasie (angl. time-invariant) - ϕ i ψ nie zależą bezpośrednio od czasu.

Jeśli zastąpimy funkcje nieliniowe przez odpowiednie macierze zyskamy system liniowy:

$$\begin{cases} \mathbf{x}(k+1) &= A\mathbf{x}(k) + B\mathbf{u}(k) \\ \mathbf{y}(k) &= C\mathbf{x}(k) + D\mathbf{u}(k) \end{cases} \quad (2)$$

W rzeczywistych systemach w równaniach oprócz stanu \mathbf{x} i pobudzenia \mathbf{u} pojawiają się także zakłócenia \mathbf{z} , które nie będą tutaj rozważane.

Klasa systemów liniowych jest od dawna dobrze znana. Znalezione wiele sposobów badania i algorytmów sterowania obiektami liniowymi. O wiele trudniejsze w sterowaniu są obiekty nieliniowe. Nawet znając funkcję ψ oraz $x(k)$ samo stworzenie modelu mogącego dokonać identyfikacji takiego obiektu jest



Rysunek 1: Model identyfikacji obiektu z wykorzystaniem sieci neuronowej.

trudne, ze względu na kumulujące się z czasem błędy numeryczne, oraz poprzez konieczność poczynienia założeń co do stabilności systemu.

Najprostsza reprezentacja systemu dynamicznego nie zawiera wewnętrznego stanu, który w wielu wypadkach może być pominięty

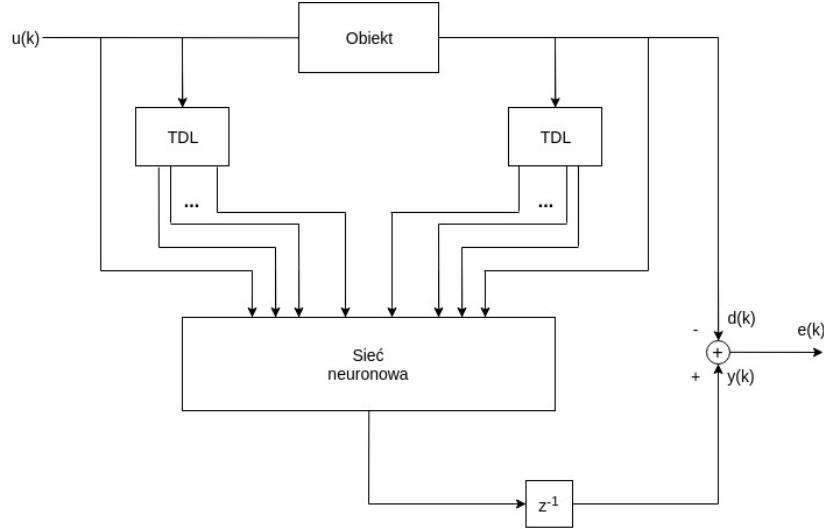
$$\mathbf{y}(k+1) = f(\mathbf{y}(k), \mathbf{u}(k)) \quad (3)$$

2.2 Zadanie identyfikacji obiektu

Zadanie identyfikacji sprowadza się do wyznaczeniu modelu reagującego na pobudzenie w sposób zbliżony do reakcji rzeczywistego obiektu:

$$\|\hat{y} - y\| < \epsilon \quad (4)$$

Uzyskanie takiego modelu umożliwia sterowanie adaptacyjne, co znacznie ułatwia pracę z obiektami nieliniowymi. Celem sterowania w ogólności jest uzyskanie na wyjściu obiektu $\mathbf{y}(k)$ wartości jak najbardziej zbliżonej do wartości zadanej $\mathbf{y}_m(k)$. W przypadku sterowania adaptacyjnego oznacza to znalezienie takiego pobudzenia $\mathbf{u}(k)$ aby została spełniona zależność:



Rysunek 2: Schemat połączenia sieci neuronowej w trakcie uczenia.

$$\lim_{k \rightarrow \infty} \|\mathbf{y}(k) - \mathbf{y}_m(k)\| < \epsilon \quad (5)$$

Często do takiego zadania są wykorzystywane sieci neuronowe przedstawione na rys. 1. Na wejście sieci podaje się wejścia systemu, oraz wyjścia z poprzednich chwil. Zadaniem sieci jest przewidzenie zachowania systemu w kroku $k+1$. Szereg bloków opóźniających będzie dalej oznaczany w pracy jako TDL (Tapped Delay Line).

W trakcie identyfikacji sieć neuronowa jest zwykle włączana sposobem szeregowo-równoległy, co zostało zilustrowane na rys 2. Taka konfiguracja umożliwia badanie błędu i minimalizację błędu zgodnie z nierównością 4. Sieć neuronowa nieustannie próbuje przewidzieć wyjście obiektu w następnej chwili. Różnica $e(k)$ jest wykorzystywana w procesie uczenia sieci. W [1] zaproponowano taką metodę oraz zaprezentowano wykorzystanie modelu do predykcji wyjścia filtru Butterwortha szóstego rzędu dla pobudzenia sygnałem trójkątnym oraz tłumionym sygnałem sinusoidalnym.

3 Zrealizowane zadania oraz otrzymane wyniki

Zadanie predykcji może zostać zrealizowane za pomocą modelowania liniowego ARX - modelu autoregresywnego z zewnętrznym wejściem. Jeżeli do modelowania zostały użyte sieci neuronowe model taki nazywa się NNARX (Neural Networ ARX).

Na podstawie danych zawartych w pliku 'actuator.m' żądany model został załadowany do środowiska programu Matlab. Wejściem obiektu (u) jest otwarcie zaworu, zaś odpowiedzią systemu(y) jest ciśnienie oleju. Pierwszym elementem potrzebnym do wykonania zadania była konstrukcja prawidłowego ciągu uczącego. W modelu ARX potrzebujemy stworzyć układ z opóźnionymi wejściami i odpowiedziami. W zależności od potrzeb modelu, możemy dostosować wielkość opóźnienia i ilość opóźnionych próbek. W naszym przypadku skonstruowana została następująca macierz wejściowa dla sieci:

$$u_s = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & u' & 0 & 0 \\ 0 & 0 & u' & 0 \\ 0 & p' & 0 & 0 \\ 0 & 0 & p' & 0 \\ 0 & 0 & 0 & p' \end{bmatrix} \quad (6)$$

gdzie u' i y' odpowiadają transponowanym wektorom otrzymanym z modelu. Analogicznie skonstruowany został wektor pożądanych wyjść dla sieci:

$$y_s = [p' \quad 0 \quad 0 \quad 0]' \quad (7)$$

Dane zostały pogrupowane na zbiór uczący (pierwsze 512 próbek), oraz zbiór sprawdzający(kolejne 512 próbek).

4 Wnioski

Nieliniowe procesy są zwykle za bardzo skomplikowane dla modelowania tradycyjnymi metodami statystycznymi. Modelowanie za pomocą nowoczesnych metod opierających się o uczenie maszynowe czy sieci neuronowe pozwala na przybliżenie konkretnych właściwości badanego układu. Nie pozwala jednak na pełną predykcję zachowań układu. Należy zwrócić uwagę, że do właściwej identyfikacji systemu niezbędna jest duża liczba danych i głębsza ich analiza. Sieć będzie się zachowywać tylko tak, jak ją tego nauczymy i tylko na podstawie informacji, które zawarte były w danych uczących. W sprawozdaniu zostały opisane najprostsze scenariusze dotyczące działania i modelowania rzeczywistych obiektów. Neurosterowniki i sieci NNARX nie są popularnie używanymi strukturami do predykcji i identyfikacji układów. Wynika to ze skomplikowanego aparatu matematycznego i obecności na rynku znacznie dokładniejszych narzędzi.

Literatura

- [1] Stanisław Osowski. *Sieci Neuronowe w ujęciu algorytmicznym* Wydawnictwo Naukowo-Techniczne, Warszawa 1996
- [2] Strona prowadzącego,
<http://staff.iiar.pwr.wroc.pl/piotr.ciskowski/skrypt/SNwMATLABie.htm>
- [3] J. Rhim, S. W. Lee *A neural network approach for damage detection and identification of structures* Computational Mechanics, Springer 1995
- [4] Jerzy Lipski *Identyfikacja procesów dynamicznych z zastosowaniem sieci neuronowych* Nauka i technika 2001