

# IDENTYFIKACJA PROCESÓW DYNAMICZNYCH Z ZASTOSOWANIEM SIECI NEURONOWYCH

## APPLICATION OF NEURAL NETWORK TO IDENTIFICATION OF DYNAMICAL PROCESS

*Zadanie identyfikacji polega na dokonaniu matematycznego opisu modelu systemu dynamicznego na podstawie serii pomiarów. Motywacją do podjęcia takich działań może być, między innymi, predykcja zachowania się systemu, wykrywanie uszkodzeń lub projektowanie układu sterowania. W artykule zwrócono szczególną uwagę na metodykę identyfikacji systemów nieliniowych z zastosowaniem sieci neuronowych. Przytoczono przykłady zastosowania opisanych metod do identyfikacji procesu skrawania.*

*System identification is the task of inferring a mathematical description a model of a dynamical system from series of measurements. Can be several motives of so actions, typical: prediction behaviors of system, detect of damage, design of control system. In this paper the attention is drawn to identification nonlinear systems with application of neural network. Has been shown an example described methods to modeling cutting process.*

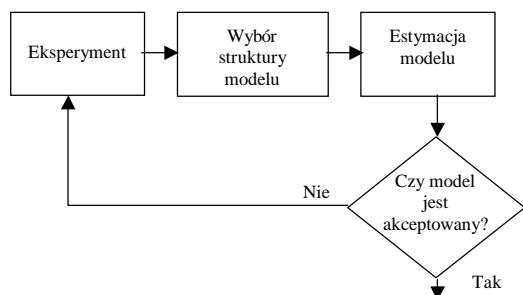
### 1. Wprowadzenie w zagadnienia identyfikacji systemów dynamicznych

W zależności od informacji *a priori* o badanym systemie, problem identyfikacji może być rozwiązywany na kilka sposobów. Jeżeli identyfikacja opiera się wyłącznie na danych z pomiarów, zakładając brak wiedzy o istocie procesów fizycznych zachodzących w systemie, to proces identyfikacji jest nazywany modelowaniem „czarnej skrzynki”. Dla kontrastu określenie modelowanie „białej skrzynki” odnosi się do czysto fizycznego podejścia w procesie modelowania. Jeżeli tylko część wiedzy o modelowanym obiekcie jest znana i wykorzystana do modelowania o pozostała pochodzi z pomiarów, to taki sposób modelowania nazywamy modelowaniem „szarej skrzynki”.

Rozważania przytoczone w niniejszym artykule dotyczą modelowania „czarnej skrzynki”. Biorąc pod uwagę fakt, że większość systemów z jakimi spotykamy się na co dzień jest nieliniowych, może budzić zdziwienie, że literatura identyfikacji systemów dotyczy najczęściej systemów liniowych. Jest kilka powodów takiego stanu. Wiele systemów dynamicznych może być z wystarczającą dokładnością dla celów modelowania opisane modelem liniowym. Obliczenia zarówno numeryczne jak i statystyczne są mniej skomplikowane na takim modelu. Można w prosty sposób dobrać zarówno strukturę jak i nastawy regulatora do sterowania obiektem opisanym modelem liniowym.

Jednakże, nieliniowości będące przyczyną powstawania drgań, wymuszają niekiedy uwzględnienie modelu nieliniowego w celu poprawy wydajności systemu sterowania obiektem. W takiej sytuacji modelowanie „czarnej skrzynki” może znaleźć zastosowanie do sterowania systemem o nieznanymi właściwościami.

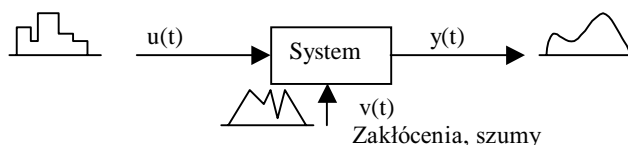
Identyfikacja modelu systemu dynamicznego przebiega zazwyczaj według algorytmu pokazanego na rys. 1.



Rys. 1. Podstawowy algorytm identyfikacji systemu

### 1.1. Eksperyment

Celem eksperymentu jest zebranie danych opisujących jak badany system zachowuje się pod wpływem zewnętrznych oddziaływań w zakresie spodziewanym podczas normalnej pracy. Polega to na dokonywaniu zmian sygnału wejściowego  $u(t)$  i obserwacji zmian wyjściowych  $y(t)$  jak na rys.2.



Rys. 2. Schemat przeprowadzenia identyfikacji

Zbiory wartości sygnałów przypisanych do wejścia i wyjścia ogólnie można zapisać:

$$Z^N = \{[u(t), y(t)], t = 1, \dots, N\} \quad (1)$$

gdzie:  $Z^N$  –  $N$  elementowy zbiór danych doświadczalnych;  
 $u(t)$  – sygnał wejściowy;  
 $y(t)$  – sygnał wyjściowy;

Jeżeli identyfikowany system będzie pracował w stanie niestabilnym może być koniecznym wielokrotne powtarzanie procesu identyfikacji, po uprzednim ręcznym usunięciu przyczyn niestabilności. Jednym z głównych zadań eksperymentatora prowadzącego proces identyfikacji jest określenie: częstotliwości próbkowania, dobór odpowiedniego sygnału wejściowego oraz wstępna obróbka danych. Wstępna obróbka danych dotyczy zazwyczaj przeprowadzenia testów nieliniowości oraz usunięcia zakłóceń, szumów i innych niepożądanych efektów.

### 1.2. Wybór struktury modelu

Najczęściej dysponujemy zbiorem potencjalnych modeli, wśród których poszukuje się tego najlepszego. Problem ten można rozwiązać dwoma sposobami:

1. Wybrać rodzinę struktury modelu najlepiej pasującą do opisywanego systemu np. strukturę modelu liniowego, sieć neuronową w formie wielowarstwowego perceptronu lub opis falkowy (wavelets).

2. Wybrać podzbiór w określonej rodzinie struktur. Dla przykładu w rodzinie modeli liniowych może to być model strukturalny ARX(2,3,1) gdzie (2,3,1) oznacza opóźnienie czasowe okresu próbkowania i to, że bieżący sygnał wyjściowy zależy od dwóch próbek sygnału wyjściowego i trzech sygnałów wejściowych z przeszłości.

### 1.3. Estymacja modelu

Gdy został wybrany zbiór potencjalnie pasujących modeli, następnym krokiem jest wybór jednego szczególnego modelu. Polega to na sprawdzeniu, który model spełnia w możliwie najwyższym stopniu założone kryteria. Kryteria te mogą być formułowane w bardzo różny sposób lecz zawsze powinny pasować do celów budowy modelu. Najczęściej stosowaną strategią jest wybór modelu, który charakteryzuje się najlepszą predykcją sygnału w chwili przyszłej odległej od bieżącej o okres próbkowania w sensie minimalnego błędu średniokwadratowego między obserwowanym sygnałem wyjściowym a przewidywanym przez model. Proces wyboru modelu jest nazywany w literaturze procesem estymacji. Ten etap budowy modelu neuronowego nosi nazwę „uczenia sieci” lub „trenowania sieci”.

### 1.4. Wiarygodność modelu

Gdy model został już wybrany powinien być poddany testowaniu w celu sprawdzenia czy spełnia nałożone wymagania. Test dotyczy szczególnie obszaru przyszłego zastosowania modelu. Kryteria w dużej mierze zależą od założonej dokładności modelowania i sposobu wykorzystania sygnału z modelu. W sytuacji, gdy podczas próbnego zastosowania modelu istnieje podejrzenie o nieadekwatność należy powrócić do jednego z poprzednich etapów tj. eksperymentu, wyboru struktury lub tylko estymacji.

## 2. Nieliniowe modele strukturalne bazujące na sieciach neuronowych

Jeżeli założyć, że budowany model będzie powstawał na zasadzie identyfikacji „czarnej skrzynki” przydatną jest struktura sieciowa nazywana wielowarstwowym perceptronem (MLP- *multilayer perceptron network*). Badania pokazały [1], że odwzorowuje on z dobrą dokładnością zależności funkcyjne między zbiorami danych wejściowych i wyjściowych po uprzednim nauczaniu na ciągach danych uczących. Pozostaje wybrać wejścia informatyczne sieci oraz wewnętrzną architekturę sieci. Najczęściej zakłada się strukturę wewnętrzną typu *feedforward* sieci MLP, ma ona szereg zalet:

- √ Jest naturalnym rozszerzeniem dobrze znanej struktury liniowej modelu;
- √ Wewnętrzna architektura może być rozszerzana stopniowo w miarę potrzeby podwyższania elastyczności w celu odwzorowania bardziej złożonych zależności funkcyjnych;
- √ Jest wygodna do zastosowań w systemach sterowania.

Tak wybrany model sieciowy można opisać równaniem:

$$y(t) = g[\varphi(t, \theta), \theta] + e(t) \quad (2)$$

lub dla formy predykcyjnej:

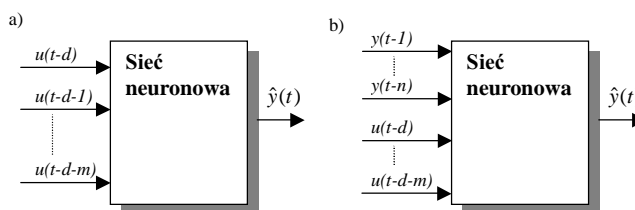
$$\hat{y}(t | \theta) = g[\varphi(t, \theta), \theta] \quad (3)$$

$\varphi(t, \theta)$  jest wektorem regresji a  $\theta$  zawiera wektor parametrów regulowanych sieci neuronowej zwany wektorem wag, jest funkcją realizowaną przez sieć przy założeniu struktury *feedforward*. W zależności od wyboru wektora regresji otrzymuje się różne struktury modeli nieliniowych. Jeżeli wektor regresji zostanie wybrany dla modeli ARX

to strukturę sieci neuronowej określa się jako NNARX jako akronim Neural Network ARX. Podobnie mogą być tworzone modele NNFIR, NNARMAX, NNOE lub NNSSIF.

### 2.1 Modele NNFIR i NNARX

W modelach tych zachodzi czysto algebraiczna relacja między przewidywanym sygnałem wyjściowym a zmierzonym w przeszłości oraz stanem wejść. Modele charakteryzują się dużą stabilnością co jest szczególnie ważne w przypadku nieliniowości gdy problem stabilności bardzo się komplikuje. Struktura modeli została pokazana na rys.3.



Rys.3. Modele strukturalne realizowane w sieci neuronowej: a) NNFIR, b) NNARX

Model NNFIR wymaga dla określenia predykcji sygnału wyjściowego ciągu sygnałów wejściowych przesuniętych w czasie o wielokrotność okresu próbkowania. Natomiast model NNARX wymaga dodatkowo informacji o wartościach od  $y(t-1)$  do  $y(t-n)$ .

Jeżeli zostanie przyjęty model regresji ARMAX to postać funkcji realizowanej przez sieć typu *feedback* przyjmie postać:

$$\varphi(t, \theta) = [y(t) \dots y(t-n), u(t-d) \dots u(t-d-m), \varepsilon(t, \theta), \dots, \varepsilon(t-k, \theta)]^T \quad (4)$$

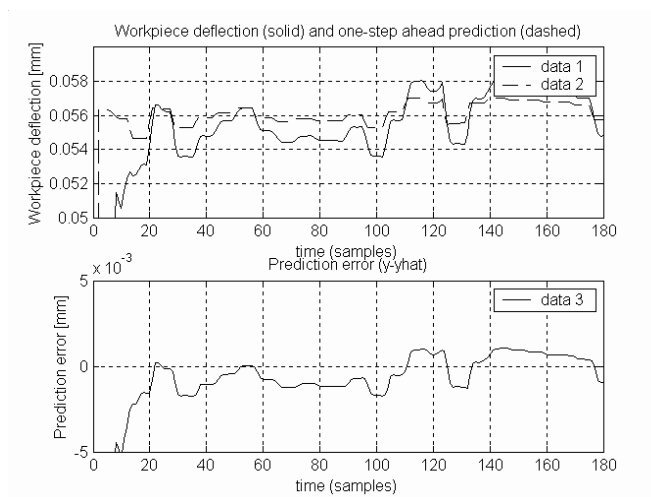
gdzie oznacza błąd predykcji, którego wartość wpływa na proces uczenia sieci. Rozwiązanie to jest określane w literaturze [2] jako *recurrent network*.

### 3. Przykład modelu procesu skrawania zbudowanego z zastosowaniem sieci neuronowej

Identyfikacja związków dynamicznych między siłami skrawania a względnym przemieszczeniem narzędzia i przedmiotu jest istotna z punktu widzenia możliwości sterowania procesem skrawania.

Do modelowania przyjęto strukturę NNARX oraz sieć neuronową o jednej warstwie ukrytej zawierającej 12 neuronów. Do uczenia sieci przygotowano zestaw uczący złożony z wektora siły odporowej oraz wektora przemieszczenia narzędzia i przedmiotu. Siłę rejestrowano podczas eksperymentu toczenia siłomierzem piezoelektrycznym umieszczonym w oprawce narzędzia, przemieszczenia mierzone bezdotykowo przy pomocy czujnika laserowego. Toczeniu poddano przedmiot o nieregularnej powierzchni co powodowało losową zmienność głębokości skrawania. Zadaniem nauczonej sieci była predykcja przemieszczenia na jeden okres próbkowania do przodu. Wyniki modelowania przedstawiono na rys.5.

Oznaczenie „data1” odnosi się do przemieszczeń rzeczywistych narzędzia i przedmiotu podczas gdy przebieg oznaczony „data2” ilustruje przewidywane wartości odkształceń przez sieć neuronową. Na dolnym wykresie pokazano błąd predykcji. Jak widać wartości błędu predykcji są rzędu tysięcznych części mm, co wskazuje na bardzo dobre dopasowanie modelu predykcyjnego. Opracowana metoda modelowania zostanie wykorzystana w dalszym etapie do sterowania procesem skrawania w trybie on-line.



Rys. 5. Wyniki modelowania procesu toczenia

#### 4. Literatura

- [1] Nørregaard M., Ravn O., Poulsen N.K., Hansen L.K.: *Neural networks for Modelling and Control of Dynamic Systems*, Springer-Verlag, London, UK, 2000.
- [2] Ljung L.: *System Identification Toolbox User's Guide*, The MathWorks Inc., 1995
- [3] Hunt K.J., Sbarbaro D., Zbikowski R., Gawthrop P.J.: *Neural Networks for Control Systems – A Survey*, Automatica, Vol. 28, No. 6, pp. 1083-1112.
- [4] Hunt K.J., Sbarbaro D.: *Neural Networks for Nonlinear Internal Model Control*, IEE Proceedings-D, Vol. 138, No. 5, pp. 431-438.
- [5] Billings S. A., Jamaluddin H. B., Chen, S.: *Properties of neural networks with applications to modelling non-linear dynamical systems*. International Journal of Control, 55(1), 193-224.
- [6] Billings S. A., Zhu Q. M.: *Nonlinear model validation using correlation tests*. International Journal of Control, 60(6), 1107-1120.