1. Ines MAHMOUD, 2. Imen SAIDI, 3. Islem BEJAOUI

University of Tunis El Manar Tunis, Tunisia. Automatic Research Laboratory, LA.R. A, National Engineering School of Tunis.
ORCID: 1. 0000-0001-6596-7050; 2.0000-0001-7366-3022; 3. 0000-0003-2959-7181

doi:10.15199/48.2022.02.09

Internal Model Control using Artificial Neural Networks for Linear Minimum Phase Systems

***Abstract**. In this paper, we are interested in the internal model control using neural networks in the case of linear minimum phase systems. We propose, to use the neural internal model control to solve the inversion problem of a model M(z) in order to design the IMC controller. An example application is presented and the implementation of the proposed approach is discussed.*

Streszczenie.** W niniejszym artykule interesuje nas sterowanie modelem wewnętrznym za pomocą sieci neuronowych w przypadku liniowych układów o minimalnej fazie. Proponujemy wykorzystanie neuronowego sterowania modelem wewnętrznym do rozwiązania problemu inwersji modelu M(z) w celu zaprojektowania sterownika IMC. Przedstawiono przykładową aplikację oraz omówiono wdrożenie proponowanego podejścia.* (Sterowanie modelem wewnętrznym za pomocą sztucznych sieci neuronowych dla liniowych systemów o minimalnej fazie***)

Keywords: internal model control, artificial neural networks, linear minimum phase systems, stability.
Słowa kluczowe: sterowanie modelem wewnętrznym, sztuczne sieci neuronowe, liniowe układy fazowe minimalne, stabilność.

Introduction

Internal model control (IMC) using neural networks has gained increasing interest in recent decades due to the robustness inherent in its structure, it allows to represent dynamic systems from experimental data when a theoretical model is unavailable. This control structure has been the subject of several research works in the case of nonlinear systems based on the neural network. An artificial neural network is a computational model whose design is inspired by the functioning of biological neurons. Neural networks are strongly interconnected structures of elementary processors. Each of the processors calculates a single output based on the information it receives [1], [2], [3], [4].

The IMC control of linear processes by application of artificial neural networks (ANNs) are developed in various studies where the model is replaced by an ANNs and inverted on-line for the calculation of the IMC controller. Neural networks can be applied to regulators either directly or indirectly. In the case of the direct method, the learning of the artificial neural networks is carried out with input-output data of the process in order to produce the action of the command which leads to the desired state at the following sampling instant. knowing the current state of the dynamic system, the ANNs thus obtained can therefore be used as a controller [5], [6], [7].

In the case of the indirect method, learning is done with the input-output data of the process, the artificial neural network predicts the future state of the system, and can be used by an algorithm for the control action.

In this paper we propose an Internal model control using neural networks for linear minimum phase systems. In this sense, we deal with the case of neural control by internal model, its operating principle, the relative neural modeling and the implementation of the proposed control for a linear minimum phase system in order to evaluate the performance of the control proposed for this class of system.

Internal Model Control

Among the robust control strategies for dynamic systems with parameters that may be uncertain, we cite sliding mode control, predictive control and internal model control. The IMC makes it possible to solve certain difficulties for the realization of a robust structure, in the presence of modeling errors, and has the advantage of ensuring desired properties including stability, precision and speed [8], [9], [10].

Internal Model Control was introduced by Garcia and Morari, as a robust control structure to solve some difficulties in modeling the system to be controlled such as errors [11]. The IMC is applied simultaneously to the process G(z) to be controlled and to its model. The specific controller of this structure is assumed to be the implicit inverse of the process model *M(z)*, Fig. 1.

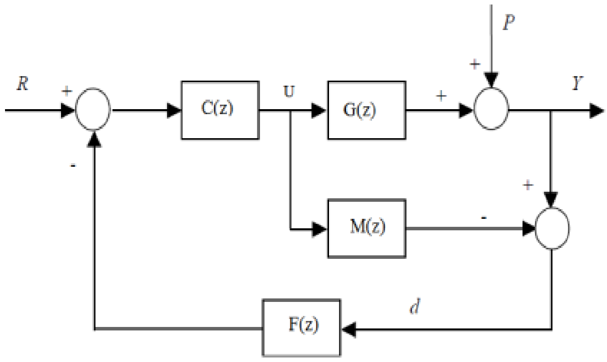

Fig.1. Internal model control scheme

The error signal includes the influence of external disturbances P as well as modeling errors. This considered structure is similar in the case of continuous time and discrete time systems. Among the necessary requirements of the IMC is the open-loop stability of its different blocks [12], [13].

The command *U* is expressed as a function of the setpoint *R* and the disturbance by the following expression, knowing that $F(z)=1$ [14], [15]:

(1) $$U(z)=\frac{C(z)}{1+C(z)\left(G(z)-M(z)\right)}R(z)-\frac{C(z)}{1+C(z)\left(G(z)-M(z)\right)}P$$

The expression of the closed-loop *Y(z)* response is then presented by the following equation:

(2) $$Y(z)=\frac{C(z)G(z)}{1+C(z)\left(G(z)-M(z)\right)}R(z)+\frac{1-C(z)M(z)}{1+C(z)\left(G(z)-M(z)\right)}P$$

Neural network in internal model control strategy

The difficulty of designing an implicit IMC controller unlike the model for a certain linear system with minimum phase [12], [14], this can be overcome thanks to the neural control by internal model. This type of command makes it possible to find online a perfect inverse model at each moment by a neural network without any problem with the dimension of the system and its model, Fig. 2.

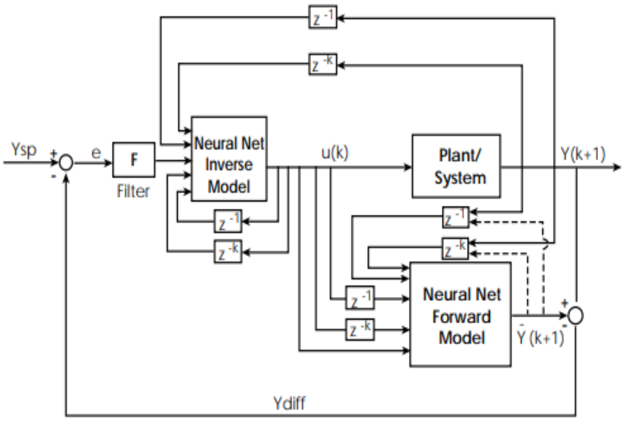

Fig.2. Neural network in internal model control strategy

According to Fig.2, the Internal model control using neural networks consists of three blocks. The first block "System" represents the system to be controlled, the second block "Neural net inverse model" represents the neural controller by internal model *C(z)*, the third block "Neural net forward model" represents the model *M(z)*.

The difference *e(k)*, between the output of the system to be controlled *y(k)* and the output of the Neural net forward model $\hat{y}(k)$ is injected as input to the neural net inverse put in series with the system to be controlled.

The neural internal model controller ensures perfect tracking of the reference trajectory in the perfect case. In the case where there are deviations between the model *M(z)* and the system *G(z)* to be controlled which can be expressed in terms of parametric uncertainties, a robustness filter *F(z)* can be added to guarantee stability [12], [13].

From Fig. 2, the neural internal model control strategy is based on the use of two different neural models. The first characterizes the dynamic behavior of the system to be controlled from the measurements of its inputs / outputs, this network represents the direct model of the system. The second model is the inverse neural model of the system [1], [2].

Neural Forward Model

To determine the direct neural model, we use a look-ahead network that estimates the output of the system from the old values of its input and output. The number of inputs, hidden layers as well as the number of neurons per layer have a great influence on the behavior of the neural model. The learning pattern of the direct neural model is given in Fig. 3.

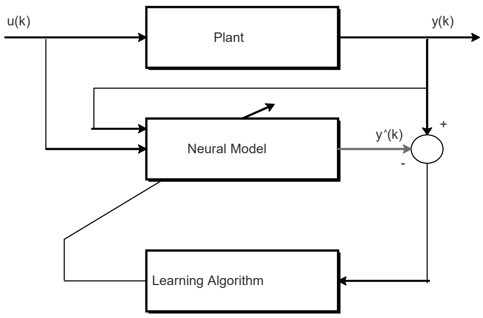

Fig.3. Forward dynamic modelling with neural networks

where: *u(k),* y(k) and $\hat{y}(k)$ represent the control input, the desired output and the output of the neural model, respectively.

The dynamics of nonlinear systems can be described by the following relation [16]:

(3) $$y_i(k) = h(y_i(k-1),...,y_i(k-n),u_i(k-1),u_i(k-m))$$

where:

(4) $$y(k) = \left[y_1(k), y_2(k),..., y_n(k)\right]^T$$

(5) $$u(k) = \left[u_1(k), u_2(k),..., u_n(k)\right]^T$$

where:
h(.) is an unknown nonlinear function, *n* and *m* represent, respectively, the number of delayed measurements of the output and the input.
$y_i(k)$ is the output of the system at sampling time *k*.

The expression of the output of the neural model is as follows [16]:

(6) $$\hat{y}_i(k) = NN(y_i(k-1),...,y_i(k-n)u_i(k-1),u_i(k-m))$$

with *NN* designates the neural network that forms the model.

In the case of point-by-point learning, the criterion to be minimized is expressed by the following equation:

(7) $$J_i = \frac{1}{2}(y_i(k) - \hat{y}_i(k))^2$$

In order to determine the parameters of the neural model, we apply to the real system a sequence of inputs, u (k), for *k = 1, ..., N*, *N* denotes the number of measurements. For each value of *u(k)*, we record the value of the output *y(k)*. The duration between two successive iterations is equal to the sampling period *Te*.

The learning procedure consists of presenting the measurements to the model to calculate the prediction error as shown in Fig. 3. During each iteration, the weights of the connections are modified until a very low modeling error is obtained [17]. The neural model is exploited using the parameters found at the end of the learning phase. The error between the output of the system and that of the model will be used to judge the quality of the model found.

❖ *Neural inverse model*

In the case of the inverse method, the training is done for the determination of the inverse model of the process. Based on the input-output data of the process, the artificial neural network replaces the model with a neural network and back online for the development of the controller. The dynamic system can be described at iteration *(k+1)* by the following equation:

(8) $$\begin{aligned} y_i(k+1) = h(&y_i(k), y_i(k-1),..., y_i(k-n+1), \\ &u_i(k), u_i(k-1),..., u_i(k-m+1)) \end{aligned}$$

The inverse model of the system given by equation (8) can be presented by equation (9).

(9) $$\begin{aligned} u_i(k) = h^{-1}(&y_i(k+1),..., y_i(k), y_i(k-1), \\ &y_i(k-n+1), u_i(k-1),.., u_i(k-m+1)) \end{aligned}$$

In order to approximate the function *h(.)* as an inverse function of *h* by a neural network, the inverse model receives as input:
- The future value of the setpoint *y(k+1)*.
-The current value and the old values of the output *y(k)*, *y(k-1),..., y (k-n+1)*.
- The old values of the input *u(k-1)*, ..., *u(k-m+1).*

In the literature, the identification of the inverse neural model can be ensured in the first step by determining the input vector, while knowing the number of delayed inputs and outputs and in the second step by determining the architecture of the network [5], [6].

Simulations Results

To validate the proposed algorithm and highlight the properties of the command proposed for the control of a linear minimum phase systems, we have developed a neural network with a fixed learning step $\mu = 0.03$. The developed control law depends on the parameters. of the neural controller such as the synaptic weight matrix, the input vector, the numbers of *n* and *m* neurons of the hidden layer and the input layer respectively.

we consider a linear minimum phase system to validate our proposed ordering approach. this system is given by the following equation.

$$\text{(10)} \qquad H\left(z^{-1}\right) = z^{-1}\frac{N\left(z^{-1}\right)}{D\left(z^{-1}\right)} = \frac{Y\left(z\right)}{U\left(z\right)}$$

where :

$$\text{(11)} \qquad N\left(z^{-1}\right) = 0.1 + 0.09\ z^{-1}\ ,\ D\left(z^{-1}\right) = 1 - z^{-1} + 0.3\,z^{-2}$$

The input vector is defined by:

$$\text{(12)} \qquad X(k) = \left[\varepsilon_i(k), \varepsilon_i(k-1), ..., \hat{y}_i(k-1), \hat{y}_i(k-2), ...\right]^T$$

$$\text{(13)} \qquad \varepsilon_i(k) = y_{ci}(k) - e_i(k)$$

$$\text{(14)} \qquad e_i(k) = \hat{y}_i(k) - y_i(k)$$

yc(k) is the reference signal, $\hat{y}(k)$ is the output estimated by the internal neural model and *y(k)* is the actual output of the system to be controlled.

The performance of the model is determined by the mean squared error (MSE) criterion given by the following equation [5]:

$$\text{(15)} \qquad MSE = \frac{1}{N}\sum_{i=1}^{N}\left(y(i) - y_c(i)\right)^2$$

The reference input signal *yc(k)* is a square amplitude signal that varies between 1 and -1, its expression is given as follows:

$$\text{(16)} \qquad y_c(k) = \begin{cases} 1, & 0 \langle k \langle T \\ -1, & T \langle k \langle 2T \end{cases}$$

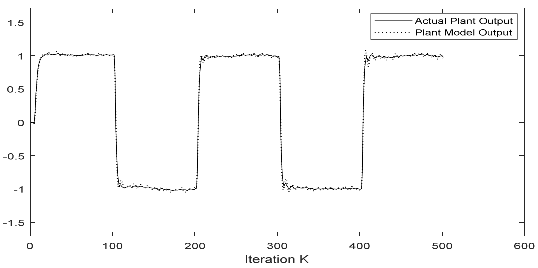

Fig.4 : Variations in actual plant output *y(k)* and plant model output estimated by the neural internal model $\hat{y}(k)$

Fig. 4 shows the evolution of the actual plant output y(k) and plant model output estimated by the neural internal model $\hat{y}(k)$. We notice that the model perfectly mimics the behavior of the system. This will lead to good closed loop performance.

Fig. 5 illustrates the evolution of the output of the system y(k), by applying the internal model control using neural networks, the output *y(k)* perfectly follows the reference setpoint with good precision without overshoot and a response of the system in term speed.

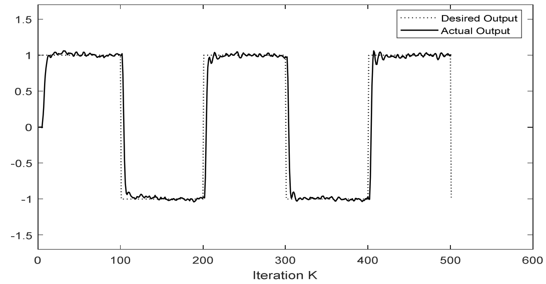

Fig.5: Desired output Vs actual output after neural internal model control

Fig. 6 presents the evolution of neural internal model control. This control law is able to make the system described by equation (10) to correctly follow the trajectory of the reference model, Fig. 5, with good precision as shown in Fig. 7.

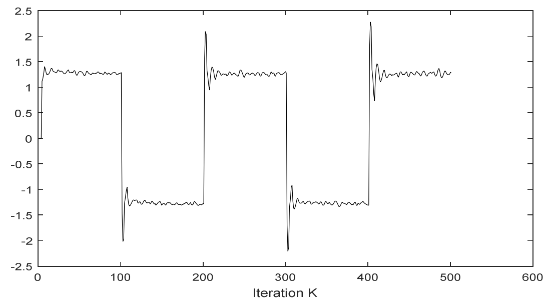

Fig.6: Evolution of the controlled input by neural internal model control law

Fig. 7 shows the difference between the setpoint *yc(k)* and the output *y(k)* by the neural internal model control. We notice that the gap tends towards zero.

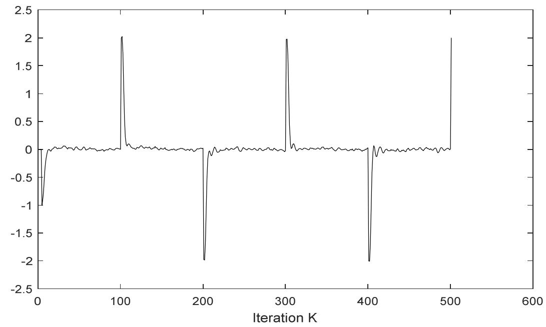

Fig.7: Error in Plant Model

Fig. 8 illustrates the difference between the desired setpoint and the direct neural model. We notice the perfect correspondence between *yc(k)* and $\hat{y}_i(k)$.

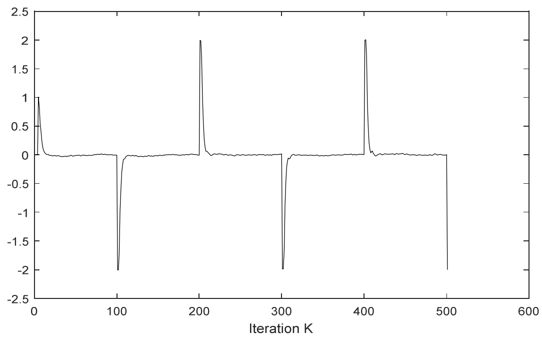

Fig.8: Error in Inverse Model

Conclusion

In this paper, we have developed, tested and discussed the different efficiencies of internal model control law using neural networks, modeled by a direct neural network and inverse neural model based controller.

The results of simulations for the linear minimum phase system have shown that the neural control by internal model is efficient by giving a good tolerance to errors in terms of robustness, stability and response time. The latter also exhibits good tolerance to modeling faults and to disturbances.

This proposed control approach will be applied for other classes of systems in a future work.

***Authors: Ines MAHMOUD**, Assistant Professor at the Ministry of National Defense, Borj Amri Aviation School, E-mail: ines.abdellatif.mahmoud@gmail.com.*

***Imen SAIDI**,* Professor at the Faculty of Sciences of Bizerte*, E-mail: imen.saidi@gmail.com.*

***Islem BEJAOUI**, has received the PhD in Electrical Engineering, from Ecole Nationale d'Ingenieurs de Tunis (ENIT) in 2021., E-mail: islem.bejaoui@enit.utm.tn.*

REFERENCES

[1] Rivals I., Personnaz L. Nonlinear internal model control using neural networks: Application to processes with delay and design issues, IEEE transactions on neural networks, 11 (2000), no.1, 80-90.

[2] Rivals I., Personnaz, L. Internal model control using neural networks, IEEE International Symposium on Industrial Electronics, Poland, (1996), 109-114.

[3] Bassil Y. Neural network model for path-planning of robotic rover systems, International Journal of Science and Technology, 2(2012), no. 2,1-6.

[4] Wang Q. G., Zhang Y., Zhang, Y. A modiffied internal model control scheme with simplified design and implementation. Chemical Engineering Communications, 184(2001), no. 1, 35-47.

[5] Bejaoui I., Saidi I., Soudani D., New Internal Model Controller design for discrete over-actuated multivariable system, 4th International Conference on Control Engineering & Information Technology (CEIT), Hamamet, 2016.

[6] Dimitris C. P., Ungar L.H. Direct and indirect model based control using artificial neural networks, Industrial & Engineering Chemistry Research, 30(1991), no. 12, 2564–2573.

[7] Yildirim S. A proposed neural internal model control for robot manipulators, Journal of Scientific and Industrial Research, 65(2006), 713–720.

[8] Saidi I., Touati N., Nonlinear predictive control for trajectory tracking of underactuated mechanical systems, Przegląd Elektrotechniczny, 97 (2021), nr. 6, 30-33.

[9] Saidi I., N. Touati N., Sliding mode control to stabilization of nonlinear Underactuated mechanical systems, Przegląd Elektrotechniczny, 97 (2021), nr. 7, 106-109.

[10] Bejaoui I., Saidi I., Xibilia M. G., Soudani D., Internal model control of discrete non-minimum phase over-Actuated systems with multiple time delays and uncertain parameters, Journal of Engineering Science and Technology Review,12(2019), no. 2, 111-118.

[11] Garcia C. G., Morari M., Internal model control. 1.a unifying review and some results, Industrial Engineering Chemistry Process Design and Development, 2 (1982), 403–411.

[12] Saidi I., N. Touati, Dhahri A., Soudani D., A comparative study on existing and new methods to design internal model controllers for non-square systems, Transactions of the Institute of Measurement and Control, 41 (2019), no.13, 3637-3650.

[13] Dhahri A., Saidi I., Soudani D., Internal model control for multivariable over-actuated systems with multiple time delay, 4th International Conference on control Engineering & Information Technology, Tunisia, (2016), 623-626.

[14] Touati N., Saidi I., Dhahri A., Soudani D., Internal multimodel control for nonlinear overactuated systems, Arabian Journal for Science and Engineering, 44(2019), no.3, 2369-2377.

[15] Touati N., Saidi I., Internal model control for underactuated systems based on novel virtual inputs method, Przegląd Elektrotechniczny, 2021, 2021(9), 95-99.

[16] Bloch G., Thomas P., Theilliol D. Accomodation to outliers in identification of non-linear SISO systems with neural networks, Neurocomputing, 14(1997), 85-99.

[17] Nelles O. Nonlinear system identification: from classical approaches to neural networks and fuzzy models. *Applied Therapeutics*, 6(2001), no.7, 21–717.