dx.doi.org/10.24254/CNIB.16.27

Stimulus identification in the excitation of the corneal nerve in an aviar model by neural networks

Ivan Salgado[1], Mariel Alfaro-Ponce[2] and Isaac Chairez[3]

[1]*CIDETEC-Instituto Politécnico Nacional*
[2] *Escuela Superior de Tizayuca, Universidad Autónoma del Estado de Hidalgo*
[3]*Bioprocesses Department, UPIBI-Instituto Politécnico Nacional*

***Abstract*—This paper describes the application of differential neural networks (DNN) to classify the inverse response of the optical nerve in an aviar model. Generally speaking, the main objective in signal classification is to obtain the class of an obtained response given a characteristic stimulus. This work deals with the inverse procedure, that is, to recover the stimulus with the measure of the optical nerve response after a classification. A robust exact differentiator (RED) based on the super-twisting algorithm (STA), that is a second order sliding mode technique, reinforces the DNN classifier. The main idea is to combine the DNN and RED approaches to implement a so-called nonlinear observer for unknown inputs. The results show how the input stimulus in the eye of an aviar model are reproduced with the DNN-STA scheme based on a previous classification obtained measuring the response in the optical nerve with the unknown stimulus.**

***Index Terms*—Signal classification, optical nerve, differential neural network**

I. Introduction

NOWDAYS, neural networks (NN) based pattern recognition solutions have been frequently applied to classify biological signals, especially in the domain of function approximation [1], pattern recognition [2], automated medical diagnostic systems [3] and some others. The success of NN in pattern recognition is a consequence of their capability to approximate nonlinear relationships between the input and output pairs [4]. Therefore, the method selected to adjust the weights in the NN structure plays a key role on forcing a higher efficiency on the classification task. In [5], a DNN solved the EEG signal pattern recognition problem. A class of DNN [6] represented the relationship between the EEG signal and its particular pattern class given by a sigmoid type function. The DNN structure preserved the highly parallel structure that characterizes many of the usual pattern recognition forms. By virtue of its parallel distribution and feedback properties, the DNN is tolerant to the presence of faults and external noises, and it is able to generalize the input-output relationships in the approximation problems. The main problem solved by DNN signal classification was to identify the class of the EEG signal. This paper deals with the opposite task solved for many signal classifiers. Once the class is known, how to identify the stimulus input that produce such class. In this manuscript, the so-called unknown input observers [7] solve the main problem. The proposed methodology employs the super-twisting sliding mode differentiator [8]. The following section describes the complete methodology to reconstruct the input stimulus in an aviar model optical nerve.

Corresponding author: I. Chairez, UPIBI-IPN, Av. Acueducto de Guadalupe s/n, C.P 07340, G. A. Madero, Ciudad de México, México jchairezo@ipn.mx

II. Stimation of unknown input stimulus by DNN and Super-Twisting Algorithm algorithms

Fig. 1 describes the main result applying the DNN approach as an identifier and the Super-Twisting Algorithm (STA) as a differentiator to obtain the estimation of the unknown input stimulus taken in the optical nerve of an aviar model. The uncertain system provides the output information in a parallel way to the DNN identifier (the adaptation law) and to the RED based on the STA. The STA estimates the first order derivative of each component of the available output in finite-time. Then, the DNN identifier supplies the information of the states and the parametric reconstruction of the state vector. These two outputs (DNN identifier and STA differentiator) are subtracted in order to estimate the unknown input. In the estimation process, a first stage implies a DNN identifier trained with prior information. The main difference in the second stage is the information used to train the DNN. The first stage uses the measurement input to perform the training. Then, the weights obtained in this stage are submitted in the second stage. The second DNN, (as Fig. 1 describes) does not use the input information in the training process, that is, the term $W_2\phi(x)u$ does not appear in the DNN's training. By this reason, the approximation obtained by the second DNN has a degree of inaccuracy. The STA provides robustness against parametric uncertainties and finite-time convergence (classical sliding mode advantages) and makes an exact estimation of the derivative of the class to which the external stimulus belongs. These differences between the two estimating methods together with the weights W^{Id}, obtained in the off-line training are the tools to obtain the estimation of the input stimulus.

A. DNN identification

The classification method described in [5] employs a DNN to classify EEG signals and it is used in this paper. We refer the reader to Fig. 1 in the aforementioned paper to get used to the method employed to classify the signals in the optical nerve

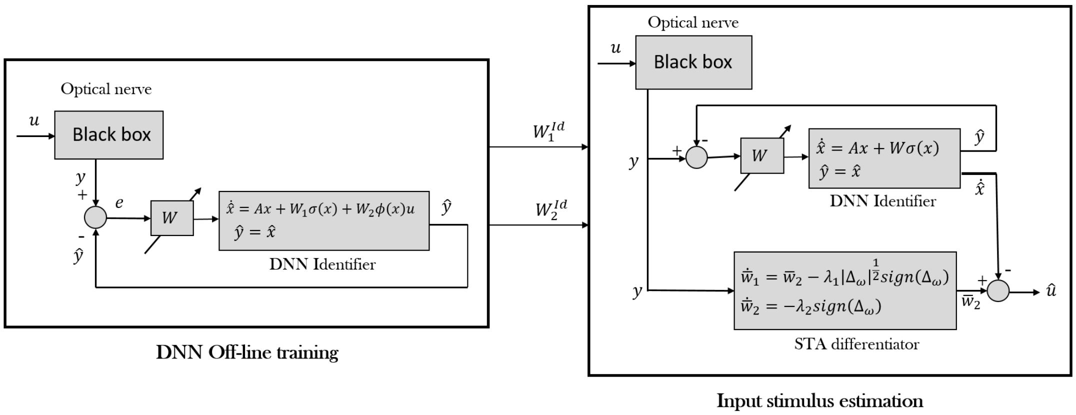

Figure 1. Complete procedure to obtain the estimation of unknown input stimulus u

of the aviar model. A sigmoid function represents a particular class of signal with an appropriate external stimulus, that is,

$$x^l(v) = \frac{a^l}{1+e^{-cv}} \tag{1}$$

where the variable x^l represents the target class and the variables l, a and c are constant parameters to characterize each class. By assumption, the next DNN represents the dynamics of function x^l in time

$$\dot{x} = Ax + W_1^*\sigma + W_2^*\phi u + f(x,u) \tag{2}$$

Where W_1^* and W_1^* represent the relationship between the optical nerve and the external stimulus u, and $f(x,u)$ is the error associated to the finite number of activation functions σ and ϕ. These activation functions are selected as sigmoid functions [9]. The next equation describes the dynamics of the DNN identifier

$$\dot{\hat{x}} = A\hat{x} + W_1^{Id}\sigma + W_2^{Id}\phi u \tag{3}$$

where the free parameters W_1^{Id} and W_2^{Id} are updated by the following differential equations

$$\begin{array}{ll} \dot{W}_1^{Id} = -k_1 P\sigma(\hat{x})\Delta^\top, & W_1^{Id}(0) = W_{1,0}^{Id} \\ \dot{W}_2^{Id} = -k_2 P\Delta u^\top\phi(\hat{x}), & W_2^{Id}(0) = W_{2,0}^{Id} \end{array} \tag{4}$$

in the last equation $W_1^{Id}(0), W_2^{Id}(0)$ are the known initial conditions of the weights. The term k is the learning coefficient of the DNN identifier and P is the solution of the following Riccati equation

$$\begin{array}{c} PA + A^\top P + PRP + Q = 0 \\ R = \bar{W} \in \mathbb{R}^{n\times n}, \quad Q = Q_0 + L_\sigma\Lambda_\sigma \\ Q_0 = Q_0^\top > 0 \quad \Lambda_\sigma = \Lambda_\sigma^\top > 0 \end{array} \tag{5}$$

Remark 1. *[6] The set of training algorithms presented in equation (8) are the result of the application of the Lyapunov based stability analysis with the following Lyapunov candidate function*

$$V = \Delta^\top P\Delta + k_1\mathbf{tr}\left[(W_1^{Id})^\top W_1^{Id}\right] + k_2\mathbf{tr}\left[(W_2^{Id})^\top W_2^{Id}\right] \tag{6}$$

Where $\mathtt{tr}(\cdot)$ *is the trace operator of a given matrix.*

B. DNN-STA estimation of input stimulus

1) DNN on-line identifier: Following the development proposed in [6], we define the DNN identifier as

$$\dot{\hat{x}} = \tilde{A}\hat{x} + W\tilde{\sigma}(\hat{x}) \tag{7}$$

supplied with a special *updating* (learning) law $\dot{W} = \Phi(W,\hat{x}|W^*)$. Notice that this DNN identifier has the same structure as the one in equation (2) but without the term regarding the input stimulus.

2) Learning law for the identifier: Let the identification error be defined by $\Delta = x - \hat{x}$, then, the adaptive learning law $\Phi(W,\hat{x}|W^*)$ for the free parameters W is given by

$$\begin{array}{l} \dot{W} = -k\tilde{P}\tilde{\sigma}(\hat{x})\Delta^\top \\ W(0) = W_0, \quad W_0 \in \mathbb{R}^{n\times n} \end{array} \tag{8}$$

in the last equation W_0 is the initial condition of the weights and it is known. The term k is the learning coefficient of the DNN identifier and $\tilde{P}$ is the solution of the following Riccati equation

$$\begin{array}{c} \tilde{P}\tilde{A} + \tilde{A}^\top\tilde{P} + \tilde{P}R\tilde{P} + \tilde{Q} = 0 \\ R = \bar{W} \\ \tilde{Q} = \tilde{Q}_0 + L_{\tilde{\sigma}}\Lambda_{\tilde{\sigma}} \\ \tilde{Q}_0 = \tilde{Q}_0^\top > 0 \quad \Lambda_{\tilde{\sigma}} = \Lambda_{\tilde{\sigma}}^\top > 0 \end{array} \tag{9}$$

3) Super-twisting sliding mode differentiator: The relative degree (for the definition of the relative degree, we refer the reader to [10], [11]) of the system (2) with respect to the unknown input ϕ, in the case of the identification problem, is equal to one. In order to recover the value of $\dot{x}$, the output signal $y = x$ must be differentiated once. For the case of the identification problem, the STA is applied as a robust exact differentiator (RED) [8] for each element of the output. In particular, the STA application is based on the following description. Define for the first element of the state vector x as $x_1 = r$, if $w_1 = r$ where $r \in \mathbb{R}$ is the signal to be differentiated, $w_2 = \dot{r}$ represents its derivative and under the assumption of $|\ddot{r}| \leq r^+$, the following auxiliary equation is gotten

$$\begin{array}{l} \dot{w}_1 = w_2 \\ \dot{w}_2 = \ddot{r} \end{array} \tag{10}$$

The previous differential equation is a state representation of the signal r. The STA algorithm to obtain the derivative of r looks like

$$\begin{aligned} \dot{\bar{w}}_1 &= \bar{w}_2 - \lambda_1 \left|\Delta_w\right|^{1/2} \operatorname{sign}\left(\Delta_w\right) \\ \dot{\bar{w}}_2 &= -\lambda_2 \operatorname{sign}\left(\Delta_w\right) \\ \Delta_w &= \bar{w}_1 - w_1 \\ d &= \bar{w}_2 \end{aligned} \tag{11}$$

where $\lambda_1, \lambda_2 > 0$ are the STA gains. Here d is the output of the differentiator [8]. In equation (11),

$$\operatorname{sign}(z) := \begin{cases} 1 & if \quad z > 0 \\ [-1, 1] & if \quad z = 0 \\ -1 & if \quad z < 0 \end{cases} \tag{12}$$

This procedure must be applied for each element of the state vector x in the sigmoid function $x(v)$. Define the vector D with the elements obtained by the parallel application of the STA to each component of the state vector x, that is, the state vector can be written as $x = [x_1, \cdots, x_n]^\top$ and $d_i \cong \dot{x}_i$ by means of the STA, then,

$$D = [d_1, \cdots, d_n]^\top \tag{13}$$

Theorem 1. *Consider the nonlinear system defined in equation (1), the DNN identifier defined in (7) and apply the sliding mode differentiator (11) to each output component of the state vector x, then, the identification error converges to a bounded region around the origin defined as*

$$\epsilon_{\tilde{x}} = \phi^+ + \tilde{f}^+ \tag{14}$$

and the unknown bounded perturbation can be obtained as

$$\hat{\phi} = D - A\hat{x} - W(t)\sigma(\hat{x}) \tag{15}$$

where $D(t)$ is defined in (13) Then, the perturbation estimation error is ultimately bounded around the origin as

$$\epsilon_{\tilde{\phi}} = \tilde{f}^+ \tag{16}$$

With $\tilde{f}^+ \in \mathbb{R}^+$ being the bound of the modelling error provided by the DNN. and $\phi^+ \in \mathbb{R}^+$ is the maximum value that the unknown stimulus can take along the time.

Proof. A similar Lyapunov stability analysis to the one presented in [6] and [12] can be followed to reach the result. □

III. Numerical results

A. DNN-STA approximation

Fig. 2 depicts the experimental setup that describes the position of the input and output electrodes. **All experiments followed the rules stated in the official norm NOM-062-ZOO-1999. Also the protocol proposed to execute the animal experiments were evaluated and approved by the ethical committee. The real images of the experimental setup are omitted in order to avoid a harmful impression to the reader.** The off-line training makes a signal classification into a classes that depends of input stimulus. Fig. 3 shows the identification of the class produced by the aviar model's optical nerve response by means of the DNN and the STA. Both algorithms reproduce the class of the nerve response. However,

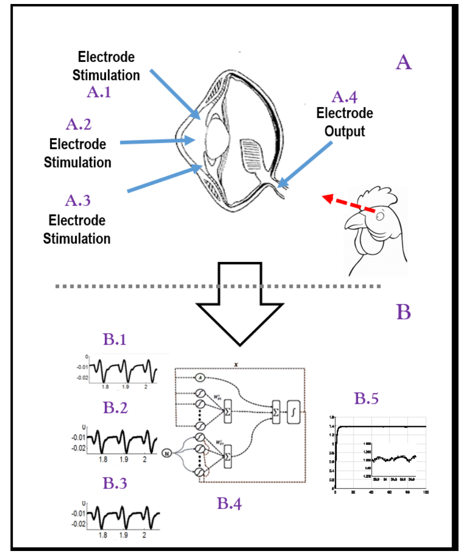

Figure 2. Experimental setup. Section A describes the position of each electrode inserted in the eye of an aviar model (A.1-A.3). Subsection (A.4) describes the position of the electrode inserted in the optical nerve. Section B describes the training in the DNN with the input stimulus (B.1-B.3). The DNN (B.4) made a signal clasiffication, the output is the class where the output stimulus (B.5) in the optical nerve belongs.

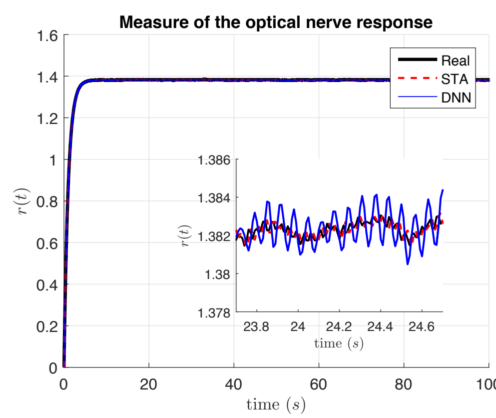

Figure 3. Off-line results provided by the DNN to classify the input stimulus in the aviar model's optical nerve

in order to see the differences between each algorithm, fig. 3 includes a closer view that shows a better approximation obtained with the STA. This difference is the key to reach the objective of the proposed method. The difference in the DNN approximation is a consequence of not use the input to train on-line the DNN. Then, the input stimulus in the optical nerve is reconstructed by means of these difference and the exact estimation that the STA provides. Fig. 5 depicts the evolution in time of DNN's weights. The DNN took less than 10 seconds to estimate the input stimulus.

Fig. 5 shows the on-line stimulus estimation by means of the DNN-STA algorithm and the Euler approximation implemented to recover the signal derivative. The Euler approximation exhibited a constant off-set in the estimation while the DNN-STA estimation obtained a better approximation. The results include a number of 10 different group of signals tested with the DNN-STA and using the Euler derivative approximation. The performance index chosen to compare them was the Euclidean norm. Fig. 6 shows this comparison. The Euclidean norm of DNN-STA approach remains below than the Euler approximation. The Euclidean norm of the DNN-Euler approach is twice bigger than the DNN-STA approximation.

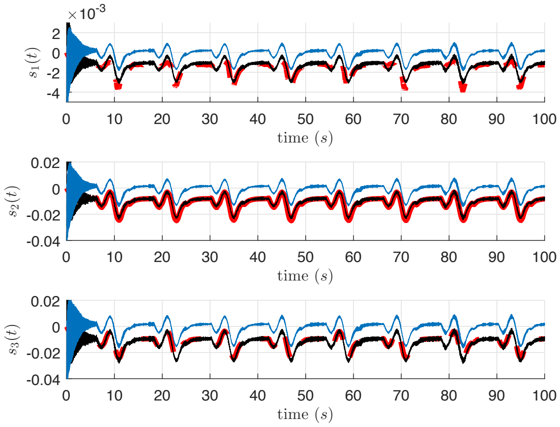

Figure 4. Estimation of unknown input stimulus u. The red line is the real input stimulus applied in the optical nerve. The black line is the input estimation obtained by the DNN-STA and the blue line is the input reconstruction provided by the DNN but with an Euler derivative approximation

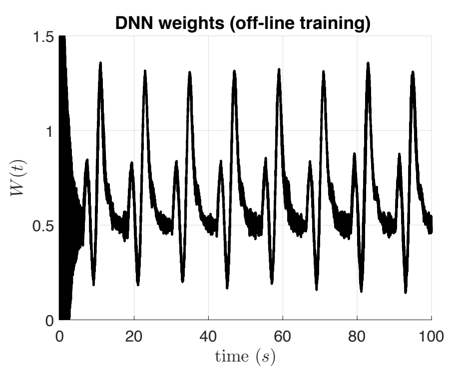

Figure 5. Behavior of the weights in the DNN. The weights in the DNN give the nonlinear relationship between the input and the output class in the signal classifier

B. Discussion

This new DNN approach to identify input stimulus (in this work applied into the optical nerve in the eye of an aviar model) seems suitable to estimate other kind of electrophisiologycal signals. In this first result, given a class (obtained with a previous classification) the DNN recovered the input stimulus. Recover the input stimulus with the measure of the electrophisiologycal signal seems to be the next step. The applicability of these results are quite important in orthosis and prosthesis devices, where sometimes, the electromyographic signal is the only available measurement, and the stimulus, that can be traduced in the orthosis/prosthesis movement is unknown.

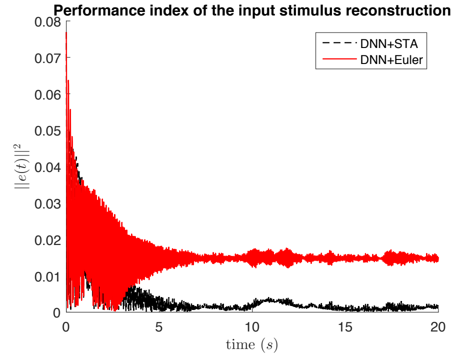

Figure 6. Comparison in the performance of the DNN input estimation with the STA and Euler differentiation techniques

With the optical signals presented in this manuscript, the idea is to recover an approximation of the image that the patient is observing just with a measure of the response in the optical nerve.

IV. Conclusion

This study constitutes a new application of the so-called unknown input observers in biomedical engineering, where the stimulus that produces a response is needed to identify. The main contribution is the the reproduction of the inverse function of a DNN applying the STA as a RED. This inverse function is equivalent to identify the stimulus injected to the optical nerve in the testing example in this manuscript. Comparing with the classical Euler approximation to obtain the derivative of the input stimulus, the results applying the STA presented better approximation capabilities.

References

[1] T. Alotaiby, S. Alshebeili, I. Alshawi, and F. El-Samie, “Eeg seizure detection and prediction algorithms: a survey,” *EURASIP Journal on Advances in Signal Processing*, pp. 1–21, 2014.

[2] N. Kasabova, K. Dhoblea, N. Nuntalida, and G. Indiverib, “Dynamic evolving spiking neural networks for on-line spatio- and spectro-temporal pattern recognition,” *Neural Networks*, vol. 41, pp. 188–201, 2013.

[3] F. Amato, E. Pea-Mendez, P. Vanhara, A. Hampl, and J. Havel, “Artificial neural networks in medical diagnosis,” *Journal of Applied Biomedecine*, vol. 11, pp. 47–58, 2013.

[4] G. Cybenko, “Approximation by superpositions of sigmoidal functions,” *Mathematics of Control Signals and Systems*, pp. 303–314, 1989.

[5] M. Alfaro, A. Arguelles, and I. Chairez, “Pattern recognition for electroencephalographic signals based on continuous neural networks,” *Neural Networks*, 2016.

[6] A. Poznyak, E. Snchez, and Y. Wen, *Differential Neural Networks for Robust Nonlinear Control (Identification, State Estimation and Trajectory Tracking)*. World Scientific Press, 2001.

[7] L. Fridman, Y. Shtessel, C. Edwards, and Y. X. G., “higher-order sliding mode observer for state estimation and input reconstrucction in nonlinaer systems,” *International Journal of Robust and Nonlinear Control*, vol. 18, pp. 299–412, 2008.

[8] A. Levant, “Robust exact differentiation via sliding mode tecnique,” *Automatica*, vol. 34, no. 3, pp. 379–384, 1998.

[9] S. Haykin, *Neural networks and learning machines*. Prentice Hall, 2009.

[10] H. Khalil, *Nonlinear Systems*. Upper Saddle River, NJ: Prentice Hall, 1996.

[11] A. Levant, “Sliding order and sliding accuracy in sliding mode control,” *International Journal of Control*, vol. 58, no. 6, pp. 1247–1263, 1993.

[12] A. Moreno and M. Osorio, “Strict lyapunov funtions for the super-twisting algorithm,” *IEEE Tran. Aut. Cont.*, vol. 57, no. 4, pp. 1035–1040, 2012.