

FedCSIS_2023_paper_3874.pdf

Use of Dynamic Neural Networks for Modeling Nonlinear Objects with Significant Nonlinearity

² Oleksandr Fomin, Sergii Polozhako, Valentyn Krykun, Andrii Orlov, Andrii Prokofiev
Odessa Polytechnic National University,
Odessa, Ukraine,
fomin@op.edu.ua

Abstract— The work is devoted to ⁹ the problem of nonlinear modeling of objects based on dynamic neural networks. The aim of the work is to improve the accuracy of modeling dynamic objects with significant nonlinearities using neural network models, and identify the scope ¹⁰ of their effective application. This aim is achieved by applying the apparatus of time delay ¹¹ neural networks for building dynamic nonlinear models. The most significant results consist in obtaining the method of building the neural network models of objects with significant nonlinearities with preservation of nonlinear and dynamic properties of the research object. Significance of the obtained results: the application of the proposed method allows both high accuracy and efficiency of the construction nonlinear dynamic models, allows providing modelling of the objects that are in a functional mode. The proposed method verified using the data of the ¹² dynamical objects with significant nonlinearities. The verification results demonstrates high accuracy and efficiency of the models building.

Keywords— Identification, nonlinear dynamic objects, significant nonlinearities, time delay neural networks.

I. INTRODUCTION

Today, as a result of the development of technology and science, practical applications increasingly consider dynamic control objects characterized by significantly nonlinear properties. Due to these characteristics, objects can function in more complex modes than objects with characteristics in the form of linear or insignificant nonlinear functions [1]. A non-significantly nonlinear ¹³ relation should be understood as the case when the nonlinear function and its first- and second-order derivatives are continuous over the entire range of input signal changes. Nonlinear links that do not satisfy this condition are considered to be significant nonlinearities.

For successful interaction with such objects (solving control, management, and diagnostic tasks), it is first of all necessary to ensure their adequate mathematical support and effective modeling tools. However, the presence of significantly nonlinear characteristics makes the ¹⁴ use of existing analytical methods ineffective. Such models do not reflect all the properties of a real object, so they cannot provide high identification accuracy [2]. An up-to-date approach to modeling nonlinear dynamic objects is the artificial neural network apparatus ¹⁵ [3, 4].

The aim of the work is to improve the accuracy of modeling dynamic objects with significant nonlinearities using neural network models, and identify the scope of their effective application.

This goal can be achieved by examining the existing neural network TDNN model architectures and identifying their advantages, disadvantages and identifying areas for their effective use:

1. Study of modeling accuracy of nonlinear objects with smooth nonlinearity.
2. The study of modeling accuracy of nonlinear objects with piecewise linear nonlinearity.
3. Investigation of modeling accuracy of discrete objects.

II. RELEVANT WORKS

Today several methods of modeling nonlinear ¹⁶ dynamic objects based on ANN are known: Dynamic Neuro-SM [5], Time-Delay Neural Networks (TDNN) [5, 6] and Wiener-type DNNs (Wiener-type Dynamic Neural Networks). [6, 7]

Dynamic Neuro-SM type models are improvements over the well-known Static Neuro-SM models [5-9], which aim to map a given approximate model of an object to an exact ¹⁷ model. Dynamic Neuro-SM models use neural networks to automatically map and change an existing equivalent model, called a coarse model, into a desired (exact) model through a learning process [10]. Such models provide improved accuracy compared to static neurospatial mapping models, but assume some a priori information about the laws of functioning of the object under study [11].

Wiener-type DNNs are based on the principle of building ¹⁸ a nonlinear dynamic Winner system, which consists of a simplified linear dynamic model followed by a nonlinear static model ¹⁹ [6, 12]. The dynamic performance is mainly related to the linear subsystem, while the complexity of the nonlinearity is contained only in the static nonlinear subsystem, which is implemented as an artificial neural networks [13-15]. This structure can significantly increase the reliability of the dynamic neural model, but has a complex (hybrid) structure, which imposes additional requirements on the network learning algorithms and narrows the scope of the model.

Among the considered variants of models TDNN are the most general structures consisting of several layers with direct connection (direct signal propagation) [16]. Such models are capable of learning from the input-output experimental data ²⁰ of nonlinear dynamic objects [17, 18] and have good convergence ²¹ properties, which are advantages over the mentioned Dynamic Neuro-SM and Wiener-type DNN methods.

²² Among the existing types of dynamic neural network models, time-delay neural networks (TDNNs) are the most general structures capable of analyzing time-shifted data [5, 19]. Such models are able to learn from the experimental ²³ input-output data of nonlinear dynamic objects and have good convergence properties, which are advantages over other neural network models.

III. TIME-DELAYED NEURAL NETS

Dynamic models based on three-layer neural networks are a common and effective direction in the modeling of nonlinear dynamic objects with continuous characteristics.

The structure of neural network used for modeling of nonlinear dynamic objects consists of three layers. The input layer connects the features of the training set with the input neurons. The input layer consists of J neurons with nonlinear activation functions. The output layer consists of one neuron, which has linear activation function.

The output signal of an TDNN at time t_k depends not only on the input signal $x(t)$ at this time, but also on the values of the input signals $x(t-1)$, $x(t-2)$, ..., $x(t-M)$ at times $t-1$, $t-2$, ..., $t-M$, respectively, where M is the length of the delay (system memory) [5, 20]. The output signal $y(t)$ of such a model is determined by the formula:

$$y(n) = b_0 + s_0 \sum_{i=0}^K r_i^2 S_i \left(b_i + \sum_{j=0}^M r_{i,j}^1 x(n-j) \right) \quad (1)$$

where b_0 , b_i – bias of the output layer neuron and input layer neurons accordingly; S_0 , S_i – activation functions of the output layer neuron and input layer neurons accordingly; r_i^2 , $r_{i,j}^1$ – weighing coefficients of the output layer neuron and input layer neurons accordingly; K – number of neurons in the input layer.

If we represent the activation function as a polynomial, the input-output relation (3) for a three-layer TDNN with M inputs and one output can be expressed as follows [21, 22].

$$y(n) = b_0 + s_0 \sum_{i=0}^K r_i^2 S_i \left(b_i + \sum_{j=0}^M r_{i,j}^1 x(n-j) \right)^n \quad (2)$$

In publications of researchers, there are many different structures of neural networks: with several hidden layers, non-signature activation functions and so on.

However, they lead to a much more complex relationship with discrete multidimensional weight function models, which does not allow us to put them on a par with neural networks as fundamental and general approaches to modeling nonlinear dynamic models built on the base of the input-output experiment.

TDNNs are networks that can efficiently learn dynamic nonlinear behavior with high order nonlinearity [23–25] if they are trained on samples of input-output data with time intervals at different levels of shift at the same time.

The use of TDNN in information systems provides unique opportunities for modeling complex nonlinear dynamic processes.

Despite a sufficient number of architectural solutions, in practice TDNN uses the standard three-layer perceptron topology [26–28]. In this case the output signal is generated by a single neuron, most often with a linear activation function. The input signal comes by means of successive sets of input data, each of which will differ from the previous one by one value from the input set.

A neural network with $M+1$ input neurons, one hidden layer with J neurons in it, and one output neuron is shown in Fig. 1.

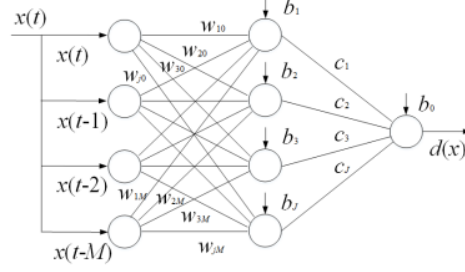


Fig. 1. TDNN in the form of a three-layer perceptron

In this case, the output signal $y(t)$ is determined by formula (3). The output of the network at time t_k depends not only on the input signal $x(t)$ at this moment, but also on the values of the input data $x(t-1)$, $x(t-2)$, ..., $x(t-M)$ at times $t-1$, $t-2$, ..., $t-M$, where M is the length of delay (system memory). The input memory M represent the memory effect of the behavioral model of the dynamic object. The number of neurons J is chosen to best fit the training sample.

IV. EXPERIMENTAL SETUP

The effectiveness of the proposed method of interpreting neural network models is investigated on the example of a test nonlinear dynamic object of the first order with quadratic nonlinearity in the feedback [29] (Fig. 1).

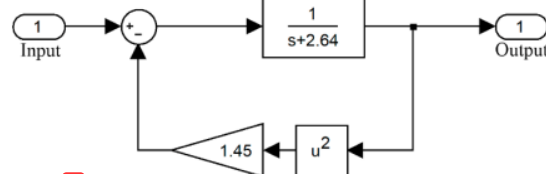


Fig. 2. Structure diagram of the test nonlinear dynamical object

The analytical expression for the model of the first order $w_1(t)$:

$$w_1(\tau_1) = e^{-a\tau_1}, \quad (3)$$

The analytical expression for the model of the second order $w_2(t, t)$:

$$w_2(t, t) = \frac{\beta}{\alpha} (e^{-2at} - e^{-at}), \quad (4)$$

To get the training sample for the learning TDNN the test object exposed by means of n test pulses with different amplitude at different time moments.

As a result of the experiment, TDNN was trained. To create a neural network, we use the Keras software module (keras.io), one of the key Python libraries for efficiently organizing APIs when modeling neural networks of arbitrary complexity. The library is most effective when building small networks with a sequential structure, where layers follow each other, there is only one input layer and one output, although it

is possible to model more complex ¹³ neural network structures with multiple inputs and outputs.

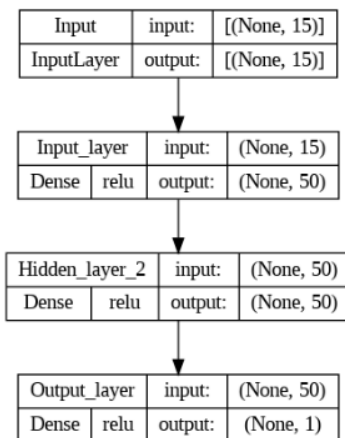
To create a neural network, we use Keras software tools (keras.io), one of the key Python libraries for efficient organization of APIs when modeling neural networks of arbitrary complexity. The library is most effective when building small networks with a sequential structure, where layers follow each other, there is only one input layer and one output, and more complex neural network structures with multiple inputs and outputs.

To build feedforward networks, the Keras library uses the Sequential model class. You can add an arbitrary number of sequential layers of the predefined types to the created model: Input, Dense, and Activation, where each layer is a function that takes the result of the previous layer as input.

The library has a ready-made set of loss functions and optimization algorithms that allow you to quickly train the model and avoid local minima whenever possible.

As a result of the experiment, a three-layer neural network was created ¹ and trained. The input signal is fed to the M+1 neurons ¹⁰ the input layer. The hidden layer consists of J neurons. The output layer consists of one neuron with a linear activation function.

Comparison of the output signals - the model's response to test input excitations demonstrates an increase in modeling accuracy using the resulting neural network by 3-6% compared ¹ linear models, and by 8-15% compared to the reference model of a nonlinear dynamic object.

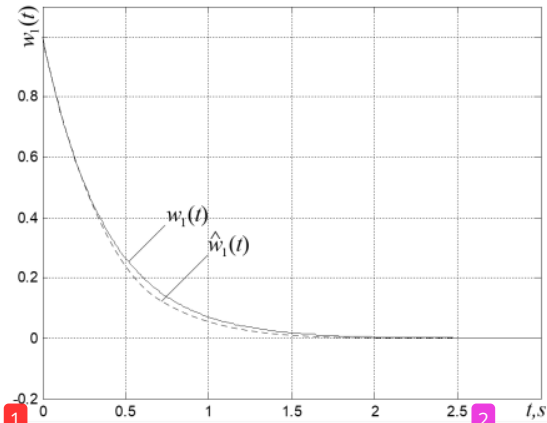


¹ Fig. 3. Structure diagram of the TDNN

¹ The calculations of the percent normalized root-mean-square error used for accuracy estimation criterion for comparison analytical model of the test object and interpretation model based on multidimensional weight functions series. The results of the comparison of the both models ² are shown in Fig 3 and Fig 4.

The obtained simulation results allow getting nonlinear dynamic model in the form of TDNN with an expected accuracy of 5-9% as compared to the simulation model.

Figure 3 shows ¹ the plots of the first-order analytical and simulation model. Figure 4 shows the plots of the second order analytical and simulation model.



¹ Fig. 4. The first-order multidimensional weight function $w_1(t)$ and it's estimation $\hat{w}_1(t)$ ²

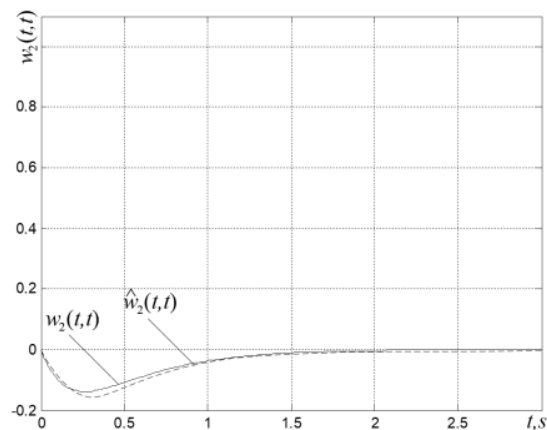


Fig. 5. The diagonal section of second order multidimensional weight function $w_2(t,t)$ and it's estimation $\hat{w}_2(t,t)$

CONCLUSION

The results of this research are as follows. ³

The scientific novelty lies in the use of integral-step series in the form of multidimensional weight functions to build analytical models that interpret neural networks with time ³ layers. This method allows obtaining interpretations for neural network models while preserving their nonlinear and dynamic properties.

The practical benefit of the developed approach is ³ improve the accuracy of building interpretive models, and to provide modeling of nonlinear dynamic objects in both test and functional modes of operation.

The proposed approach is tested on the data of a test nonlinear dynamic object. The experimental results demonstrate an increase in the accuracy of identification of interpretive models compared ² to linear models by 3-6%.

The speed of building an interpretive model is limited by the training time of the neural network. Therefore, the direction of further research may be methods to increase the training speed of neural networks with time delays.

- [1] C. Rudin and J. Radin, "Why are we using black box models in AI when we don't need to? A lesson from an explainable AI competition", *Harvard Data Science Review*, vol. 2, no. 1, 2019.
- [2] J. Sen, "Machine Learning – Algorithms, Models and Applications", London, United Kingdom, IntechOpen, 2021.
- [3] W. Liu, Y. Su and L. Zhu, "Nonlinear Device Modeling Based on Dynamic Neural Networks: A Review of Methods," 2021 IEEE 4th International Conference on Electronic Information and Communication Technology (ICEICT), Xi'an, China, 2021, pp. 662–665.
- [4] W. Samek, G. Montavon, S. Lapuschkin, C. J. Anders and K.-R. Müller, "Explaining Deep Neural Networks and Beyond: A Review of Methods and Applications", *Proceedings of the IEEE*, vol. 109, no. 3, pp. 247–278.
- [5] M. Sugiyama, H. Sawai and A. H. Waibel, "Review of TDNN (time delay neural network) architectures for speech recognition", *IEEE International Symposium on Circuits and Systems*, vol. 1, 1991, pp. 582–585.
- [6] L. Wenyuan, L. Zhu, F. Feng, W. Zhang, Q.-J. Zhang, L. Qian and G. Liu, "A time delay neural network based technique for nonlinear microwave device modelling, in: *Micromachines*", Basel, vol. 11, no. 9, 2020, p. 831.
- [7] G. Favier, A. Y. Kibangu and T. Bouilloc, "Nonlinear system modelling and identification using Volterra-PARAFAC models", *International journal of adaptive control and signal processing*, Wiley, vol. 26, no. 1, 2012, pp. 30–53.
- [8] O. O. Fomin, O. D. Ruban and O. V. Rudkovskiy, "Method for construction the diagnostic features space of switched reluctance motors based on integral dynamic models", *Problemele energeticii ionale*, no. 4, 2020, pp. 35–44.
- [9] J. Petch, Sh. Di, W. Nelson, "Opening the Black Box: The Promise and Limitations of Explainable Machine Learning in Cardiology", *Canadian Journal of Cardiology*, vol. 38, no. 2, 2022, pp. 204–213.
- [10] S. Mullainathan and J. Spiess, "Machine Learning: An Applied Econometric Approach", vol. 31, no. 2, Spring, 2017, pp. 87–106.
- [11] S. Marton, S. Lüdtke and C. Bartelt, "Explanations for Neural Networks by Neural Networks", *Appl. Sci*, vol. 12, 2022, p. 980.
- [12] W. Samek, G. Montavon, S. Lapuschkin, C. J. Anders and K.-R. Müller, "Explaining Deep Neural Networks and Beyond: A Review of Methods and Applications", *Proceedings of the IEEE*, vol. 109, no. 3, 2021, pp. 247–278.
- [13] J. Sen, "Machine Learning – Algorithms, Models and Applications", London, United Kingdom, IntechOpen, 2021. (<https://www.intechopen.com/books/10651>)
- [14] J. Schoukens and L. Ljung, "Nonlinear System Identification: A User-Oriented Road Map", *IEEE Control Systems Magazine*, vol. 39, no. 6, 2019, pp. 28–99.
- [15] A. C. Meruelo, D. M. Simpson, S. M. Veres and Ph. L. Newland, "Improved system identification using artificial neural networks and analysis of individual differences in responses of an identified neuron", *Neural Networks*, vol. 75, 2016, pp. 56–65.
- [16] G. Sarkar, J. Saha and Md. Rokonzaman, "Development of regression equation for optimizing the materials requirements of lime and sand stabilizing adobe based on consistency and linear shrinkage", *International Journal of Applied Science and Engineering Research*, vol. 1(3), 2012, pp. 499–511.
- [17] A. Agresti, "Foundations of linear and generalized linear models", Wiley series in probability and statistics, 2017.
- [18] I. Hachchane, A. Badri, A. Sahel and Y. Ruichek, "Large-scale image-to-video face retrieval with convolutional neural network features", *IAES International Journal of Artificial Intelligence*, vol. 9, no. 1, 2020, pp. 40–45.
- [19] A. Amanat, M. Rizwan, A.R. Javed, M. Abdelhaq, R. Alsaqour, S. Pandya and M. Uddin, "Deep Learning for Depression Detection from Textual Data", *Electronics*, vol. 11, 2022, p. 676.
- [20] H. Purwins, B. Li, T. Virtanen, J. Schlüter, Sh. Chang and T. Sainath, "Deep Learning for Audio Signal Processing, Journal of Selected Topics of Signal Processing", vol. 13, no. 2, 2019, pp. 206–219.
- [21] A. R. Rao and M. Reimherr, "Non-linear functional modelling using neural networks", 2021. (<https://arxiv.org/abs/2104.09371>)
- [22] G. Govind and P. A. Ramamoorthy, "Multilayered neural networks and Volterra series: The missing link", *Proceedings of International Conference on Systems Engineering*, 1990, pp. 633–636.
- [23] V. Z. Marmarelis and Zh. Xiao, "Volterra models and three-layer perceptrons", *IEEE Transactions on neural networks*, vol. 8, no. 6, 1997, pp. 1421–1433.
- [24] H. Eivazi, H. Veisi, M. H. Naderi and V. Esfahanian, "Deep neural networks for nonlinear model order reduction of unsteady flows", *Physics of Fluids*, vol. 32, no. 105104, 2020.
- [25] C. A. Mitrea, C. K. M. Lee and Z. Wu, "A comparison between neural networks and traditional forecasting methods: case study", *International journal of engineering business management*, vol. 1, no. 2, 2009, pp. 19–24.
- [26] G. Stegmayer, M. Pirola, G. Orenco and O. Chiotti, "Towards a Volterra series representation from a neural network model", *WSEAS Transactions on Circuits and Systems*, archive 1, 2004, pp. 55–61.
- [27] M. Sugiyama, H. Sawai and A. H. Waibel, "Review of TDNN (time delay neural network) architectures for speech recognition", *IEEE International Symposium on Circuits and Systems*, vol. 1, 1991, pp. 582–585.
- [28] L. Wenyuan, L. Zhu, F. Feng, W. Zhang, Q.-J. Zhang, L. Qian and G. Liu, "A time delay neural network based technique for nonlinear microwave device modelling, in: *Micromachines*", Basel, vol. 11, no. 9, 2020, p. 831.
- [29] V. Pavlenko, and V. Speranskyy, "Analysis of Identification Accuracy of Nonlinear System Based on Volterra Model in Frequency Domain", *American Journal of Modeling and Optimization*, vol. 1, no. 2, 2013, pp. 11–18.

66%

SIMILARITY INDEX

PRIMARY SOURCES

- 1

Sergii Polozhaenko, Oleksandr Fomin, Andrii Orlov, Valentyn Krykun, Daria Lys. "Interpretation Method for Dynamic States Neural Network Models", 2022 IEEE 3rd International Conference on System Analysis & Intelligent Computing (SAIC), 2022
Crossref

1650 words — 54%
- 2

Oleksandr Fomin, Sergii Polozhaenko, Valentyn Krykun, Andrii Orlov, Daria Lys. "Chapter 24 Interpretation of Dynamic Models Based on Neural Networks in the Form of Integral-Power Series", Springer Science and Business Media LLC, 2023
Crossref

96 words — 3%
- 3

www.researchgate.net
Internet

74 words — 2%
- 4

Wenyuan Liu, Weicong Na, Lin Zhu, Jianguo Ma, Qi-Jun Zhang. "A Wiener-Type Dynamic Neural Network Approach to the Modeling of Nonlinear Microwave Devices", IEEE Transactions on Microwave Theory and Techniques, 2017
Crossref

49 words — 2%
- 5

Wenyuan Liu, Yi Su, Lin Zhu. "Nonlinear Device Modeling Based on Dynamic Neural Networks: A Review of Methods", 2021 IEEE 4th International Conference on Electronic Information and Communication Technology (ICEICT), 2021

43 words — 1%

6	deepbci.korea.ac.kr Internet	37 words — 1%
7	Lin Zhu, Jian Zhao, Zhuo Li, Wenyan Liu, Lei Pan, Haifeng Wu, Deliang Liu. "A general neuro-space mapping technique for microwave device modeling", EURASIP Journal on Wireless Communications and Networking, 2018 Crossref	17 words — 1%
8	Dmitry Maevsky, Andriy Bojko, Oleksandr Besarab, Elena Maevskaya. "Theory of the Electric Circuits with Coupled Capacitors", 2022 IEEE 16th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), 2022 Crossref	16 words — 1%
9	er.chdtu.edu.ua Internet	14 words — < 1%
10	hepex.irstea.fr Internet	13 words — < 1%
11	repository.tudelft.nl Internet	11 words — < 1%
12	id.scribd.com Internet	8 words — < 1%
13	nozdr.ru Internet	8 words — < 1%

