Computing the dynamic AC of an electrical network via a fuzzy adaptive recurrent neural network

Spyridon D. Mourtas[1,2*]

[1]Department of Economics, Division of Mathematics-Informatics and Statistics-Econometrics, National and Kapodistrian University of Athens, Sofokleous 1 Street, 10559 Athens, Greece
[2]Laboratory "Hybrid Methods of Modelling and Optimization in Complex Systems", Siberian Federal University, Prosp. Svobodny 79, 660041 Krasnoyarsk, Russia

Abstract. The convergence and durability of zeroing neural networks (ZNN), a special family of recurrent neural networks, have been the subject of much recent research. Numerous time-varying problems in science and engineering have been successfully solved by ZNN dynamics. An improvement of the ZNN design for calculating the dynamic alternating current (AC) of an electrical network, which is a specific time-varying linear matrix equation problem, is proposed in this paper by utilizing a suitable defined neutrosophic-logic system (NS). In particular, the gain parameter in the ZNN architecture can be dynamically adjusted over time to accelerate the convergence of the ZNN model using an appropriate value that is acquired as the outcome of an adequately built NS. The results of the application demonstrate that the NS-based ZNN model defines the varying-gain parameter more effectively than the corresponding standard ZNN model.

1 Introduction

Electrical distribution networks are currently crucial components of global economic growth. An essential first step in verifying the effectiveness of electrical distribution networks with respect to energy losses, conductor chargeability and voltage profiles is to analyze them from the perspective of power flow [1]. The dynamic alternating current (AC) of an electrical network was calculated in this work using neural networks (NNs), which are widely used in engineering and science domains to tackle time-varying linear matrix-equation (TLM) problems, such as robot control and optimization [2], signal-processing, and statistics [3]. One crucial and effective parallel processing method for solving matrix-equation problems is the dynamic system approach. As a result of substantial research in NNs, several dynamic and analog solutions that rely on recurrent NNs (RNNs) have been created and studied [3]. The RNN technique is becoming more and more recognized as a potent substitute for online computing because of its parallel distributed nature and ease of circuit building [4].

* Corresponding author: spirmour@econ.uoa.gr

The zeroing NN (ZNN) is regarded as a contemporary online computation technique and is a specific type of RNN [4]. In a broad range of time-varying problems, such as matrix-equations systems [5], quadratic optimization [2], linear equations systems, and generalized inversion, ZNN has been extensively studied and utilized primarily as a zeroing equation tool to produce online solutions. Declaring an error matrix-equation (ERM) $E(t) \in \mathbb{R}^n$ is the first step in solving the liable issue in the formation of ZNN dynamical development. The following dynamical development [4] has to be solved next:

$$\dot{E}(t) = -\lambda \cdot \mathrm{E(t)}, \tag{1}$$

where the derivative of time is denoted by $\dot{(\,)}$ and the varying-gain parameter applied to scale the convergence is denoted by $\lambda > 0$.

The current direction in ZNN development is the development of fuzzy adaptive ZNN models. Recent research has focused on enhancing the ZNN design by integrating fuzzy control, given that fuzzy logic systems (FS), which has been founded on fuzzy and linguistic principles [6], can handle imprecision, vagueness, ambiguity, and uncertainty. But the evolution of intuitionistic fuzzy set theory gives rise to the neutrosophic set theory, that enhances the FS into a neutrosophic-logic system by adding the indeterminacy-membership function [7]. The ZNN dynamic system (1) is converted into the fuzzy adaptive ZNN dynamic system in the following manner in order to improve the ZNN method using FS:

$$\dot{E}(t) = -\lambda^{\nu} \cdot \mathrm{E(t)}, \tag{2}$$

where the fuzzy parameter is denoted by ν.

In this paper, the next specific TLM is tackled utilizing the ZNN approach:

$$AX(t) = B(t), \tag{3}$$

where t denotes the time, $A \in \mathbb{R}^{n\times n}$ and $B(t) \in \mathbb{R}^n$ are the coefficient matrices and $X(t) \in \mathbb{R}^n$ is the desired solution. Notice that A is a time-invariant matrix.

A neutrosophic-logic controller (NC) is used to acquire the needed neutrosophic parameter, ν, in the correctly determined neutrosophic adaptive ZNN (NAZNN) dynamic system structure of (2), which is used in this study to tackle the TLM problem. The convergence speed of NAZNN is faster than that of the corresponding classic ZNN model, according to the application's results when calculating the dynamic AC of an electrical network. To put it another way, the NAZNN outperforms the comparable conventional ZNN model in resolving the TLM issue. The following are the study's main outcomes.:

- An suitable parameter ν in (2), which is defined as the output of an NC, is used to offer a novel NAZNN model for solving a particular TLM problem.
- The NAZNN model is more successful than the corresponding classic ZNN model at defining the value of the varying-gain parameter, according to applications on calculating the dynamic AC of an electrical network.

2 Methods

This study introduces NAZNN, a neutrosophic adaptive ZNN model for solving the TLM issue, in line with the current trend in ZNN research. Therefore, by developing and implementing an NC, the study seeks to improve the ZNN design's performance. In order to solve (3), we want to use the flow of the dynamic system (2) and a suitable neutrosophic parameter ν. This section explains the ZNN and NAZNN models for resolving the TLM issue (3) and presents the NC that can improve the performance of the ZNN design.

2.1 The NC structure

A neutrosophic set $\mathcal{K}$'s entries are neutrosophic numbers with the following structure [7]:

$$\mathcal{K} = \{\langle x\colon T_{\mathcal{K}}(x), I_{\mathcal{K}}(x), F_{\mathcal{K}}(x)\rangle \mid\ x \in \mathcal{U}\}, \tag{4}$$

where $T_{\mathcal{K}}(x), I_{\mathcal{K}}(x)$, and $F_{\mathcal{K}}(x)$ are the truth, indeterminacy and falsity membership functions. Note that the outcomes of those functions are independent and within the range $[0,1]$. The $T_{\mathcal{K}}(x), I_{\mathcal{K}}(x)$, and $F_{\mathcal{K}}(x)$ values will be used to anticipate the varying-gain parameter λ^{ν} values in the NAZNN design. It seems reasonable to use $\eta := \| E(t) \|_F$ as a metric for building NC in the NAZNN design since the primary goal in the ZNN design is $E(t) = \mathbf{0}$. Keep in mind that $\mathbf{0} \in \mathbb{R}^n$ is the zero matrix and $\|\cdot\|_F$ is the matrix Frobenius norm.

Three phases make to the development of the NC according to our methodology. The procedure of neutrosophication, the neutrosophic inference engine, and the procedure of de-neutrosophication are these phases.

The input set becomes the fuzzy input set $\mathcal{J}$ during the neutrosophication process, and the output set becomes the fuzzy output set $\mathcal{G} = \{T_{\mathcal{K}}, I_{\mathcal{K}}, F_{\mathcal{K}}\}$, which is in the neutrosophic format. Notably, the membership functions for truth, indeterminacy, and falsehood are determined as Gaussian, Z-shaped, and sigmoid functions, respectively. We specifically take into account the truth sigmoid-membership function that follows:

$$\mathrm{T}(\eta) = 1/(1 + \mathrm{e}^{-a_1(\eta - a_2)}). \tag{5}$$

where a_1 and a_2 are parameters accountable for the function's slope at the intersection $\eta = a_2$. We also look at the following falsity Z-shaped-membership function:

$$\mathrm{F}(\eta) = \begin{cases} 1, & \eta \leq s_1 \\ 1 - 2\left(\frac{\eta - s_1}{s_2 - s_1}\right)^2, & s_1 \leq \eta \leq \frac{s_1 + s_2}{2} \\ 2\left(\frac{\eta - s_2}{s_2 - s_1}\right)^2, & \frac{s_1 + s_2}{2} \leq \eta \leq s_2 \\ 0, & \eta \geq s_2 \end{cases}, \tag{6}$$

where s_1 and s_2 are parameters accountable for the function's shoulder and the foot. Finally, we look at the following indeterminacy Gaussian-membership function:

$$\mathrm{I}(\eta) = \mathrm{e}^{-\frac{(\eta - d_2)^2}{2d_1^2}}, \tag{7}$$

where d_1 and d_2 are parameters for the standard deviation and mean.

The following rule of "IF-THEN" is regarded as the rule of neutrosophic among $\mathcal{J}$ and $\mathcal{G}$ in the neutrosophic inference engine:

$$R\text{: If } \mathcal{J} = \text{SE then } \mathcal{G} = \{T_{\mathcal{K}}, I_{\mathcal{K}}, F_{\mathcal{K}}\},$$

where a substantial inaccuracy is precisely indicated by SE, which is a fuzzy set.

The conversion $\langle \eta : T(\eta), I(\eta), F(\eta) \rangle \to \nu(\eta, t) \in \mathbb{R}$ yields a crisp value $\nu(\eta, t)$ throughout the neutrosophication process. Specifically, the de-neutrosophication rule that follows is suggested to determine $\nu(\eta, t)$:

$$\nu(\eta, t) = r_1 + r_2 \frac{T(\eta) + I(\eta) + F(\eta)}{t + 1}, \tag{8}$$

where $t \in \mathbb{R}_0^+$, and the lower and upper limit parameters of (8) are denoted by $r_1 \geq 1$ and $r_2 \geq 0$.

2.2 Addressing the TLM issue via ZNN and NAZNN

The following ERM is declared in line with the first stage of the ZNN development, taking into account the TLM problem (3):

$$E(t) = AX(t) - B(t) \in \mathbb{R}^n. \tag{9}$$

Its first time derivative is the following:

$$\dot{E}(t) = A\dot{X}(t) - \dot{B}(t). \tag{10}$$

According to the ZNN design's subsequent phase, (9) and (10) are extended in (1) as follows:

$$A\dot{X}(t) - \dot{B}(t) = -\lambda(AX(t) - B(t)), \tag{11}$$

or equivalent,

$$A(t)\dot{X}(t) = -\lambda\big(AX(t) - B(t)\big) + \dot{B}(t). \tag{12}$$

The following ZNN model is then produced by applying the vectorization vec(·) and Kronecker product ⊗:

$$(I \otimes A(t))\mathrm{vec}(\dot{X}(t)) = \mathrm{vec}(-\lambda\big(AX(t) - B(t)\big) + \dot{B}(t)). \tag{13}$$

Notice that $I \in \mathbb{R}^{n\times n}$ is the identity matrix. As a result, the suggested ZNN model for resolving the TLM issue is (13). Utilizing the exact process as for the ZNN model of (13), the NAZNN model that follows is produced:

$$(I \otimes A(t))\mathrm{vec}(\dot{X}(t)) = \mathrm{vec}(-\lambda^{v}\big(AX(t) - B(t)\big) + \dot{B}(t)). \tag{14}$$

Therefore, the suggested NAZNN model for resolving the TLM issue is (14) as a result. Notice that (13) and (14) can efficiently be tackled by applying an ode solver of MATLAB. Furthermore, [4] indicates that the ZNN model of (13) converges to the theoretical solution (TSOL), whereas [8] indicates that the NAZNN model (14) converges to the TSOL.

3 Results and discussion

In an engineering application, the NAZNN model is contrasted with the corresponding conventional ZNN model in this section. Calculating an electrical network's dynamic AC is part of the application. In both models, the initial state is set to $X(0) = \mathbf{0}$, the varying-gain parameter is set λ=10, and the solver ode15s of MATLAB is used with at time range [0,10]. Also, the NC's parameters are set to $a_1 = r_1 = s_1 = 1, a_2 = 3, d_1 = 2, s_2 = 7$and $r_2 = d_2 = 6$.

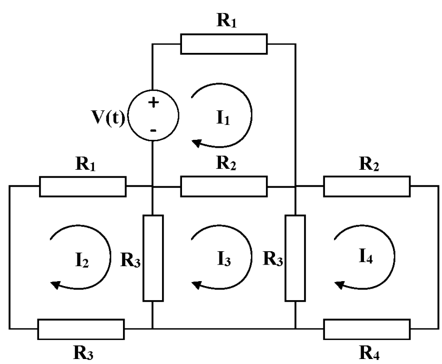


Fig. 1. Electrical network.

Fig. 1 shows the electronic circuit, and $V(t)$ is a source of dynamic AC voltage. The circuit's dynamic AC currents are computed using the ZNN and NAZNN models. Kirchhoff's voltage law and the loop-current approach yield the following form for the linear circuit equations:

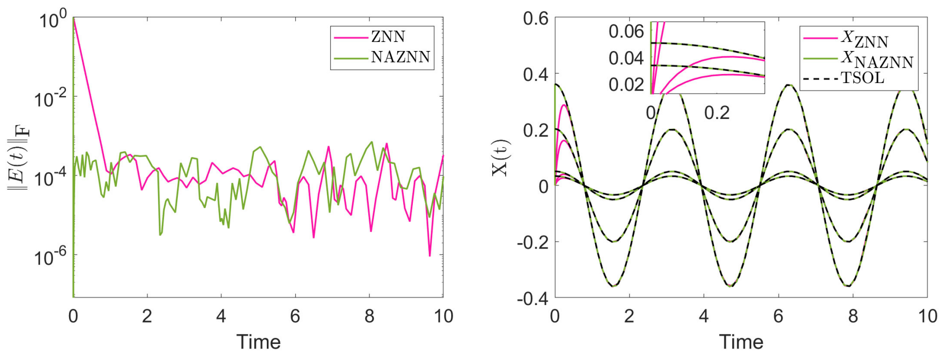
$$\begin{cases}(R_1 + R_2)I_1 - R_2 I_3 = V(t)\\ (R_2 + 2R_3)I_2 - R_3 I_3 = \mathbf{0}\\ (R_2 + 2R_3)I_3 - R_2 I_1 - R_3 I_2 - R_3 I_4 = \mathbf{0}\\ (R_2 + R_3 + R_4)I_4 - R_3 I_3 = \mathbf{0}\end{cases}$$

where the electrical resistance are denoted by R_1, R_2, R_3, R_4, the dynamic AC voltage source is denoted by $V(t)$ and the desired electric currents of each loop are denoted by I_1, I_2, I_3, I_4.

The TLM of (3) can be used to express the above electronic circuit equation as follows:

$$A = \begin{bmatrix} R_1 + R_2 & 0 & -R_2 & 0 \\ 0 & R_2 + 2R_3 & -R_3 & 0 \\ -R_2 & -R_3 & R_2 + 2R_3 & -R_3 \\ 0 & 0 & -R_3 & R_2 + R_3 + R_4 \end{bmatrix}, B(t) = \begin{bmatrix} V(t) \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad X(t) = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix},$$

where $R_1 = 1\Omega, R_2 = 4\Omega, R_3 = 2\Omega, R_4 = 6\Omega$ and $V(t) = \cos(2t)$. Observe that the present matrix $X(t)$ is the unknown matrix that needs to be obtained.



(a) ERMs values. **(b)** Error of TLM.

Fig. 2. Errors and solutions trajectories.

The findings of the application are presented in Fig. 2. The convergence of ERMs is shown in the subfigure 2a. The ERM convergence for both models begins at $t = 0$ in a value over 10, as this figure illustrates, but for each model, it terminates at alterative times t in the range $[10^{-6}, 10^{-3}]$. In the case of ERMs, the ZNN model's convergence ends at $t = 0.8$, while the the NAZNN model's convergence ends at $t = 10^{-7}$. The $X(t)$ trajectories of the application are shown in the subfigure 2b. The $X(t)$ trajectories of the models respond similarly to the ERMs convergence of their model, as this figure illustrates. In other words, the NAZNN model's solution trajectories converge to TSOL at $t = 10^{-7}$, whereas the ZNN model's solutions trajectories converge to TSOL at $t = 0.8$. Therefore, when it comes to addressing the TLM problem, the NAZNN model outperforms the corresponding standard ZNN model. To sum up, the NAZNN method is useful for solving the TLM and may also be used to calculate the dynamic AC of an electrical network.

4 Conclusion

In order to solve the TLM problem, this research presented a unique neutrosophic adaptive ZNN model, NAZNN. The goal of the study was to develop and integrate an NC in order to improve the performance of the conventional ZNN design. The NAZNN model outperforms the corresponding classic ZNN model in an engineering application that involves calculating the dynamic AC of an electrical network. Nonlinear activation functions may be used in future studies to speed up further the convergence rate of ZNN dynamics.

This work was supported by the Ministry of Science and Higher Education of the Russian Federation (Grant No. 075-15-2022-1121).

References

1. O.D. Montoya, F.M. Serra, C.H. De Angelo, Electronics, **9**, 1352 (2020)
2. N. Zhong, Q. Huang, S. Yang, F. Ouyang, Zh. Zhang, IEEE Access, **9**, 50810-50818 (2021)
3. A Cochocki, R. Unbehauen, *Neural networks for optimization and signal processing* (John Wiley & Sons, Inc., New Jersey, USA, 1993)
4. Y. Zhang, S. S. Ge, IEEE Transactions on Neural Networks, **16**, 1477-1490 (2005)
5. P.S. Stanimirović, S.D. Mourtas, D. Mosić, V.N. Katsikis, X. Cao, S. Li, IEEE Transactions on Neural Networks and Learning Systems, **2024** 1-10 (2024)
6. L. A. Zadeh, Information and Control, **8**, 338-353 (1965)
7. F. Smarandache, *A unifying field in logics: Neutrosophic Logic, Neutrosophy, Neutrosophic Set, Neutrosophic Probability* (American Research Press, Rehoboth, 2003)
8. J. Dai, Yu. Chen, L. Xiao, L. Jia, Y. He, IEEE Transactions on Industrial Informatics, **18**, 2434-2442 (2022)