

ICSGCE 2011: 27–30 September 2011, Chengdu, China

Modeling of Electric Power Transformer Using Complex-Valued Neural Networks

Yury S. Chistyakov^{a*}, Elena V. Kholodova^b, Alexey S. Minin^a,
Hans-Georg Zimmermann^c, Alois Knoll^d

^a Corporate Technology Department, Siemens LLC, Volynskiy lane 3, 191186, St. Petersburg, Russia

^b Corporate Technology Department, Siemens LLC, Letnikovskaya str. 10/11, 115114, Moscow, Russia

^c Corporate Technology Department, Siemens AG, D-81730 Muenchen, Mch-P, R. 53.220

^d Technical University of Munich. Boltzmannstraße 3, 85748 Garching bei München, Germany

Abstract

Accurate simulation of a power grid requires use of detailed power equipment models in order to reflect maximum of complex dynamics occurs in the grid. Conventional approaches are not always sufficient to fulfill necessity of meticulous description of processes in power devices. Existence of physical difference even between devices of exactly the same type pulls the accuracy of the whole grid simulation using one model for each type of equipment down. A completely new approach of power equipment modeling – modeling based on Complex-Valued Neural Networks (CVNN) – gives an opportunity to build a high-quality models which are able to track dynamics of grid devices. The nature of the approach makes it relatively easy to build models of all electric network devices even individually taking into account the uniqueness of each one. Power transformer, being quite common and, generally, complicated nonlinear element of power grid, has been chosen for demonstration of CVNN method. Results obtained from this work show that application of CVNN in power engineering modeling appears as quite promising method.

© 2011 Published by Elsevier Ltd. Selection and/or peer-review under responsibility of University of Electronic Science and Technology of China (UESTC).

Keywords: Complex-valued neural network; CVNN; Transformer modeling; Power equipment modeling

1. Introduction

Detailed modeling of electrical power network's elements is necessary for receiving accurate model data. At the same time, element's model complication may lead to significant increase of calculation time, memory overrun and other computation problems. Proposed modeling with Complex-Valued Neural Networks (CVNN) makes it possible to easily model equipment nonlinearities and uniqueness keeping

* Corresponding author. Tel.: +7(812)324-8302.

E-mail address: yury.chistyakov@siemens.com.

model complexity on the appropriate level. As power grid element for CVNN-based modeling, power transformer has been chosen. Model was performed with use of two methods – conventional analytical model and CVNN-based model. Conventional method is described in the first part of the paper and these results of transformer simulation were used for CVNN training in the second part.

Distinctive feature and of introduced complex-valued neural network is its intrinsic capability to deal with the complex numbers instead of the real ones. This feature is quite useful in frame of power grid elements modeling. Presented paper shows promising results for further research in this direction.

2. Analytical Modeling of Transformer

2.1. Basic transformer principles and equations

In a basic transformer one of the windings, named a primary winding, is energized by an external voltage source. The alternating current flowing through a primary winding creates a variable magnetic flux in magnetic core. The variable magnetic flux in magnetic core creates electromotive force (EMF) in all windings, including primary. When current is sinusoidal absolute value of EMF is equal to the first derivative of a magnetic flux. EMF induces current in the secondary winding. Ideal transformer without losses is shown in Fig. 1.

Equivalent circuit of generic transformer is shown in Fig. 2. Power losses are represented as resistances R_1 (primary) and R_2 (secondary), flux leakage – as reactances X_1 (primary) and X_2 (secondary). Iron losses caused by hysteresis and eddy currents in the core are proportional to the core flux and thus to the applied voltage. Therefore they can be represented by resistance R_m . To maintain the mutual flux in the core magnetizing current I_μ is required. Magnetizing current is in phase with the flux. Since the supply is sinusoidal, the core flux lags the induced EMF by 90° can be modeled as a magnetizing reactance X_m in parallel with the resistance R_m . R_m together with X_m are called magnetizing branch of the model. In case of open-circuit, current I_0 represents the transformer's no load current [2], [3].

Analysis of circuit significantly simplifies if the circuit with magnetically connected windings will be replaced by an equivalent circuit, elements of which are electrically connected with each other (see Fig. 3). Here the number of turns in primary (N_1) and secondary (N_2) is equal, so the parameters of the transformer have to be changed in order to maintain all energy relations. The secondary winding is moved

(or "referred") to the primary side utilizing the scaling factor: $N^2 = \left(\frac{N_1}{N_2}\right)^2$

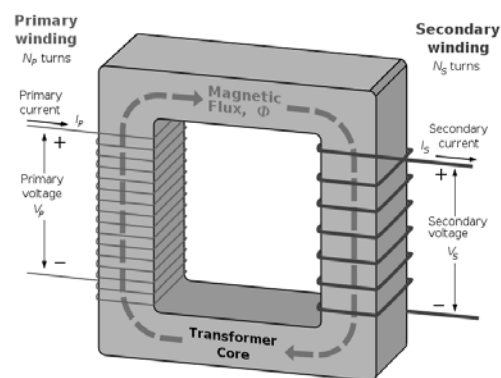


Fig. 1. Ideal transformer [1]. Electric and magnetic circuits are depicted. No losses in windings assumed on the picture.

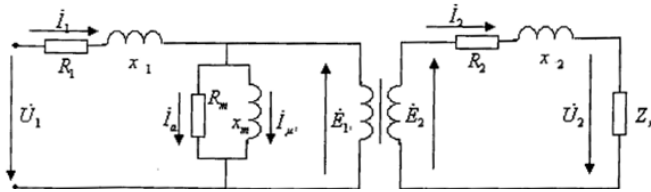


Fig. 2. Equivalent circuit of a transformer [2, p.53]

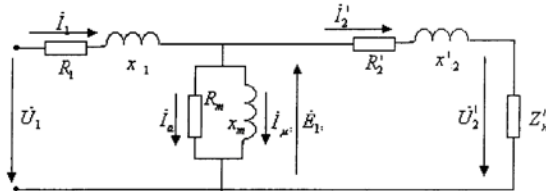


Fig. 3 Equivalent circuit of a transformer referred to the primary winding [2, p.54]

Finally, transformer equations can be written as follows [2]:

$$\begin{cases} U_1 = E_1 + I_1(R_1 + jX_1) = E_1 + I_1 Z_1 \\ E_1 = R_2' I_2' + jX_2' I_2' + U_2' \\ I_1 = I_0 + I_2' \end{cases} \quad (1)$$

where $U_1, E_1, I_1, R_1, X_1, Z_1$ – primary winding voltage, EMF, current, resistance, reactance and impedance, respectively. Secondary winding is described with similar values, but already referred to the primary winding:

$$\begin{aligned} U_2' &= U_2 N, & I_2' &= \frac{I_2}{N} \\ R_2' &= R_2 N^2, & X_2' &= X_2 N^2 \end{aligned} \quad (2)$$

2.2. Equivalent circuit parameters

Given transformer model is based on the real transformer data of Russian transformer OMP-10/10 [4].

Table 1: Transformer parameters

Parameter	Symbol	Value	Unit
Nominal power	S	10	kVA
Primary winding voltage	U_1, U_{hv}	10	kV
Secondary winding voltage	U_2, U_{lv}	400	V
No load current	I_{nl}	4.2	%
No load power	P_{nl}	60	W
Short circuit power	P_{sc}	280	W

Short circuit voltage	U_{sc}	3.8	%
-----------------------	----------	-----	---

Using factory data from short circuit and no load tests (see Table 1) other transformer parameters were calculated:

Primary winding:

$$Z_1 = \frac{U_{sc} \cdot U_1^2}{100 \cdot S} = \frac{3.8 \cdot 10000^2}{100 \cdot 10000} = 380 \text{ Ohm}$$

$$R_1 = \frac{P_{nl} \cdot U_1^2}{S^2} = \frac{60 \cdot 10000^2}{10000^2} = 60 \text{ Ohm}$$

$$X_l = \sqrt{(Z_1^2 - R_1^2)} = \sqrt{(380^2 - 60^2)} = 375 \text{ Ohm}$$

(3)

Secondary winding:

$$Z_2 = \frac{U_{sc} \cdot U_2^2}{100 \cdot S} = \frac{3.8 \cdot 230^2}{100 \cdot 10000} = 0.201 \text{ Ohm}$$

$$R_2 = \frac{(P_{nl} \cdot U_2^2)}{S^2} = \frac{60 \cdot 230^2}{10000^2} = 0.032 \text{ Ohm}$$

$$X_2 = \sqrt{(Z_2^2 - R_2^2)} = \sqrt{(0.201^2 - 0.032^2)} = 0.198 \text{ Ohm}$$

$$I_0 = I_a + I_\mu = \frac{S \cdot I_{nl}}{U_1 \cdot 100} = \frac{10000 \cdot 4.2}{10000 \cdot 100} = 0.042 \text{ A}$$

(4)

Other values:

$$\cos \phi_0 = \frac{P_{nl}}{U_1 \cdot I_0} = \frac{60}{10000 \cdot 0.042} = 0.143$$

$$R_0 = \frac{S}{U_1 \cdot I_0 \cdot \cos \phi_0} = \frac{10000}{10000 \cdot 0.042 \cdot 0.143} = 166.5$$

$$X_0 = \frac{S}{U_1 \cdot I_0 \cdot \sin \phi_0} = \frac{10000}{10000 \cdot 0.042 \cdot 0.143} = 14.06$$

(5)

2.3. Conventional model improvement

In order to improve the basic model, temperature dependences have been introduced in the equations:

1) Windings' resistance dependence from temperature:

$$R = R_{nom} (1 + \alpha(T - 20))$$

(6)

where R is calculated winding resistance,

R_{nom} – nominal winding resistance, α – temperature coefficient, T – temperature.

2) Load impedance depends from temperature.

Transformer windings are assumed to be made from copper and corresponding temperature coefficient $\alpha = 3.8 \cdot 10^{-3} \text{ K}^{-1}$ was used.

Implemented transformer model works on some specified RL load which should be treated as

equivalent impedance of some power system, supplied by the transformer. For the load it is assumed that depending on temperature fluctuations some changes in the power systems happen (e. g. switching climate control devices) which lead to the change of impedance.

These enhancements are believed to reflect some part of complicated dynamics in the real system more precisely.

2.4. Modeling results

Analytical modeling has been carried out in MATLAB. Consequence of modeling is the following: with given source voltage U_1 , primary winding current I_1 is calculated:

$$I_1 = \frac{U_1}{Z_1 + Z_m \cdot (Z'_{load} + Z'_2) / (Z_m + (Z'_{load} + Z'_2))} \quad (7)$$

Then, E_1 is found. After that, magnetizing current is computed and according to Kirchoff's law current in the secondary winding (I_2) is obtained. Then, with use of main transformer equations (1), U_2 is calculated.

Results, obtained from transformer simulation are presented in Fig. 4. As it can be seen, variation of temperature leads to adequate voltage and current response. Rise of temperature increases load impedance, which by-turn decrease primary current (I_1), secondary current (I_2) and secondary voltage (U_2).

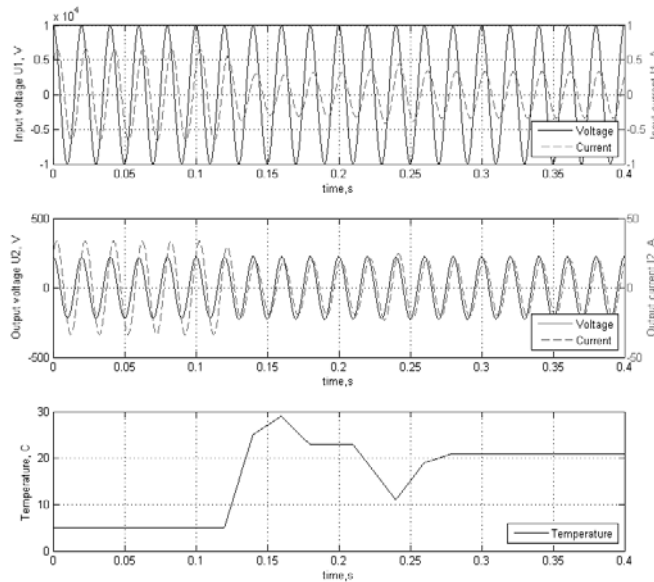


Fig. 4. Results of the simulation. Due introduced temperature dependencies changing temperature causes change of windings and load impedances. Finally, voltages and currents are affected.

One should note that the aim of the simulation was to generate the data which will show general possibility of CVNN to model such a device. Range of temperature change within the simulated time period (0.4 s) is not physical, but it does not matter in the neural network training business. See next sections for details.

3. Complex-Valued Neural Networks

Complex-Valued Neural Network (CVNN, see [5] and [6]) is a method for data approximation, based on traditional real-valued neural networks, where inputs of network, weights of network and transition functions are complex numbers. This is an essential extension of traditional neural networks to the complex plain. In the following work we consider the complex numbers in Euler notations, namely absolute part and phase. For more information on CVNN authors refer to the works [7] and [8].

In the paper we will briefly discuss the basics of CVNN to give the reader more information on this novel approach.

For the current research we will use the so-called multi layer perceptron, which inputs are complex numbers (current, voltage, etc.). These inputs propagate through the input layer ($netin_0$) of the network, then these inputs go to the first hidden layers input ($netout_0$). Then this input is to be multiplied with the weights matrix W_l . After this linear algebra operation one should apply transition function f . The procedure should be repeated iteratively. After the information goes out of the network ($netout_2$) it should be compared with the teacher signal *target* (see Fig. 5).

The quality measure is the root mean squared error (see [9]):

$$E = \frac{1}{T} \sum_{t=1}^T (y_t - y_t^d) \overline{(y_t - y_t^d)} \rightarrow \min_w \quad (8)$$

In order to adjust the network weights one should calculate the Taylor expansion of the error:

$$E(w + \Delta w) = E(w) - \eta g^T \bar{g} + \frac{\eta^2}{2} g^T G \bar{g} \quad (9)$$

In order to train the network one should minimize the approximation error. After the Taylor expansion is calculated one can extract the training rule out of (8) appears as $\Delta w = -\eta \cdot \bar{g}$, $g = \frac{dE}{dw}$. Using this simple rule one can do the gradient descent optimization and train the network. This algorithm is called error back propagation algorithm. Here, in order to calculate all partial derivatives of the error with respect to the weight one will have to propagate the derivative of the error back through the network till the input layer. Full procedure is presented in Fig. 5 (first proposed in [10]). In case of CVNN, the back propagation algorithm remains nearly the same as it is typically used in real-valued neural networks. There are only few changes which are widely discussed in [7].

Training of neural network is the presentation of patterns to the network, back-propagation of the error and weights adjustment. After each pattern from the training set of data is presented, the training epoch is finished. One can start the second training epoch and repeat the procedure once again. After the limit of epochs is reached the network can be considered as trained. After the network is trained, one can use it to map the inputs into the outputs.

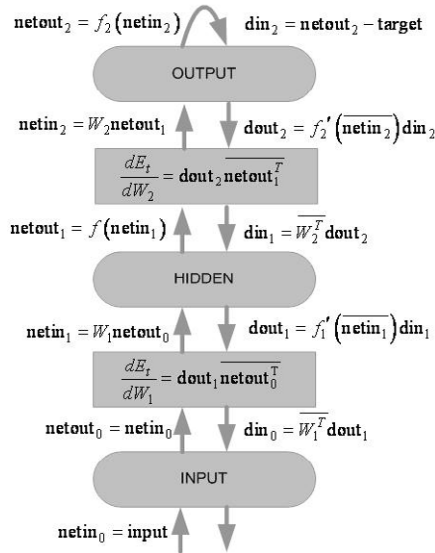


Fig 5. CVNN, multi layer perceptron. The figure shows information flow inside the network. The bar above values means complex conjugate, T means matrix transpose.

In order to simulate the transformer we have the set of input parameters (input voltage, current and temperature). Secondary transformer voltage and current are the outputs of the neural network. The task is to find the mapping from inputs into the outputs so that the selected inputs together with the neural network can lead to the set of expected outputs.

Using the transformer model described above one can generate as much training and test data as needed. In the current work we have generated 3000 data points (patterns). 2000 patterns were used for network training and the rest 1000 was used to test the network and to provide the results. For this experiment the network had 2 hidden layers with 20 neurons at each layer. Transition functions are selected to be *tanh*. Learning rate equals to $\eta = 0.002$. The amount of epochs for training is equal to 500. To provide better modeling quality the committee of 20 networks is used. After all networks are trained we use the mean output of all networks. See Fig. 6 to find out error convergence during the training.

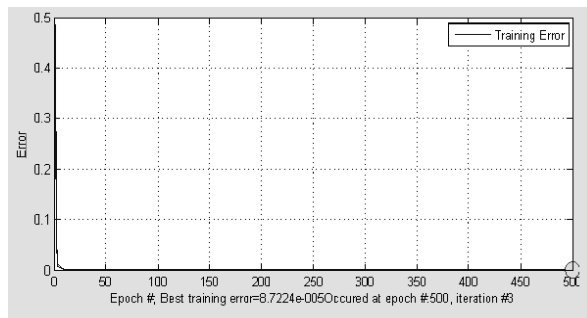


Fig. 6. Training error. The final error is 10-5 which occurred at the last training epoch.

The network decay for the particular problem turned out to be exponential, which means that this problem is rather simple for the neural network. More advanced analytical model should be used in order

to reproduce real device behavior, some noise can also be added in order to check the approximation. The best check of the approach is with real data measured from transformer in the grid.

Following statistics for the training set has been introduced in order to see how good the network is able to approximate the data of the training set:

- Root mean squared error (rms)
- Correlation coefficient (r)
- Determination coefficient (R2)

One can see the results of CVNN-based model in Fig. 7. and Fig 8. First part (a) of the Fig. 7 is the absolute part of the training set for the network. Second (b) shows absolute part of the network output for the test set. Phase part of the network output for the training set is depicted in Fig. 8 (a) and last part (b) is the phase part of the network output for the test set.

The information in which we are interested is mainly concentrated in the absolute part of the network output, but the phase part also contains important information, which is in our case the quality of the mapping. Looking at the phase we can say how good the network operates. Moreover, in case we should have phase distortions, we would see it also in the network output, which means it can also predict the phase distortions. This feature is not possible with the real-valued network. In the example with the transformer we do not have phase distortions and it behaves linearly. This behavior can be seen at all phase network outputs in Fig. 8.

The conclusion out of the modeling is that all statistical coefficients at the trainings set and at the test set are close to their best limit values, which gives us a possibility to say that transformer modeling is efficiently done using the CVNN.

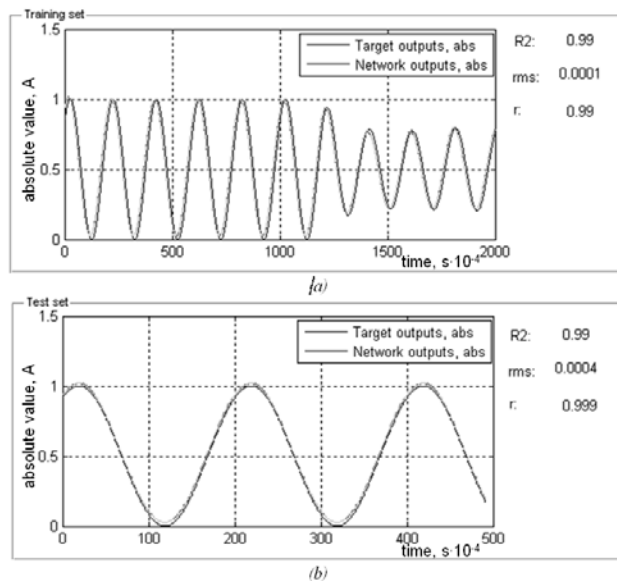


Fig. 7. Results of the transformer modeling. Absolute part of the network output for the training set (a) and for the test set (b).

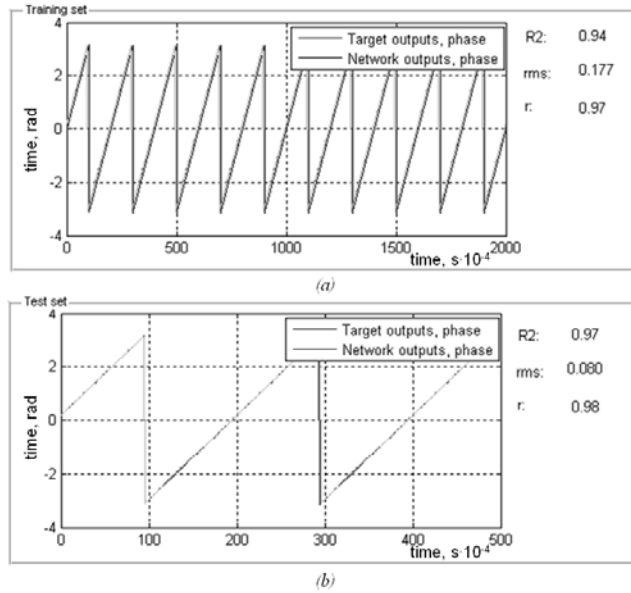


Fig. 8. Results of the transformer modeling. Phase part of the network output for the training set (a) and for the test set (b) respectively. The difference between two lines is very small therefore it looks like there is only one line at the figure.

4. Conclusions

The paper presented application of complex-valued neural networks for modeling of transformer. Significant end-use of the approach consists in integration of obtained CVNN-based transformer model in power engineering simulation software packages.

From obtained results the following conclusions can be formulated:

- General possibility of CVNN to model dynamics of advanced transformer model has been shown.
- Further tests with enhanced model have to be carried out in order to prove the preliminary simulation results. Injection of appropriate nonlinearities and adding noise in the analytical model for generating data will make the task more realistic.
- Tests with data from real devices have to be implemented. The attractive feature is that it is possible to model each grid device individually, just teaching the CVNN with measured data from particular device.
- CVNN could be applied for other power engineering equipment simulation.

Acknowledgment

Authors appreciate help of Andrei Szabo during paper preparation.

References

- [1] Power transformer: basic principles [Online]. Available: http://en.wikipedia.org/wiki/Power_transformer#Basic_principles
- [2] A. I. Voldek, V. V. Popov. *Vvedenie v elektromekhaniku. Mashini postoyannogo toka i transformatory: Uchebnik dlya vuzov.*, St. Petersburg, Piter, 2007

- [3] S. G. German-Galkin, G. A. Kardonov. *Electricheskie mashiny. Laboratornie raboti na PK.*, St. Petersburg, Korona print, 2003.
- [4] Orion OMP transformers catalog [Online]. Available: http://www.orion-nt.ru/cat/transf/tr-hmel/tr_omp1.htm
- [5] H. Leung, S. Haykin, The Complex Back Propagation, IEEE Transactions on Signal Processing., Vol.39, No.9, September 1991., pp. 2101 – 2104.
- [6] T. Kim, T. Adali, Fully Complex Multi-Layered Perceptron Network for Nonlinear Signal Processing, VLSI Signal Processing 32, pp. 29-43, 2002.
- [7] Zimmermann H.-G., Minin A., Kuserbaeva V., Historical Consistent Complex Valued Recurrent Neural Network, ICANN 2011, Part I, LNCS 6791, pp. 185–192, 2011.
- [8] Zimmermann H.-G., Minin A., Kuserbaeva V., Comparison of the Complex Valued and Real Valued Neural Networks Trained with Gradient Descent and Random Search Algorithms, Proc. of ESANN 2011, pp. 216-222, 2011.
- [9] D. H. Brandwood. A complex gradient operator and its application in adaptive array theory. IEE Proceedings, F: Communications, Radar and Signal Processing, 130(1):1116, 1983.
- [10] R. Neuneier, H.G. Zimmermann, How to Train Neural Networks, Neural Networks: Tricks of the Trade, Springer 1998, pp. 373-423.