Electric Power Systems Research 147 (2017) 73–80

Contents lists available at ScienceDirect

Electric Power Systems Research

journal homepage: www.elsevier.com/locate/epsr

A continually online trained impedance estimation algorithm for transmission line distance protection tolerant to system frequency deviation

Chrystian Dalla Lana da Silva [a,*], Ghendy Cardoso Junior [a], Adriano Peres de Morais [a], Gustavo Marchesan [b], Fernando Guilherme Kaehler Guarda [a]

[a] *Universidade Federal de Santa Maria, Roraima Av., 1000, Santa Maria, RS, Brazil*
[b] *Universidade Federal do Pampa, General Osório Av., 900, Bagé, RS, Brazil*

ARTICLE INFO

Article history:
Received 28 November 2016
Received in revised form 20 February 2017
Accepted 21 February 2017

Keywords:
Artificial Neural Networks
Distance relaying
Impedance estimation
Adaptive phasor estimation
Power system protection

ABSTRACT

Distance relays are protective devices which main goal is the protection of transmission lines. However, the presence of harmonics and the exponentially decaying DC offset in the system voltage and current signals negatively affects the relay performance. In this paper, an adaptive phasor estimation method based on Artificial Neural Networks is presented to reduce these components' effects, focusing on impedance estimation for distance relaying. The method uses the multilayer perceptron architecture to estimate the current and voltage signals, and then proceeds to calculate the complex apparent impedance during a continually online training process. This online training allows for adaptability regarding the system frequency, providing tolerance against its deviation. Graphical results of the test cases are presented, comparing the functionality and performance of the proposed algorithm with a Fourier-based method.

1. Introduction

The voltage and current signal amplitude and phase angle estimation in power systems has always received considerable attention due to the effect that noisy and distorted data have on protection devices. Specially for distance relaying, the importance of providing a clean signal input is made clear due to zones 2 and 3 being usually very wide, despite having a time delay. Whether the relay misinterprets a fault or operates in a zone it was not supposed to, the error generated affects the whole system; hence the importance of estimation techniques.

Regarding the techniques for impedance or phasor estimation, there is a wide array of methods available in the literature. The differential equations algorithm [1] calculates the transmission line parameters by solving the voltage and current differential equations. The Walsh filter [2] is a phasor estimation method which determines the fundamental component through the calculation of coefficients similar to the Fourier filter. The Kalman filter [3] is a predictive recursive filter that estimates the variables on the system in order to calculate its phasors. However, out of all methods available, the most used for phasor estimation in commercial relays are the Fourier and cosine filters.

This paper will focus on the Fourier algorithm for comparison purposes. Even though largely used on protection relays due to requiring low processing time and satisfactory results, it has a poor performance in off-nominal frequencies situations [4]. Several works published in the literature aim to improve the algorithm [5,6], but none address the off-nominal frequency issue. During a power oscillation or faults the system frequency usually deviates from its nominal value, and the distance relay may cause a decision error in such critical moment.

The application of Artificial Neural Networks (ANNs) in phasor estimation is not anything new or groundbreaking. As shown in several works [7–10], they can be very effective in this area. These papers, however, focus on presenting a new way of estimating phasors for current and voltage sinusoidal signals. The algorithm and simulations presented in this paper will be focused towards impedance estimation for numerical distance protection.

Regarding distance protection, one application of ANNs that stands out due to being extensively studied is estimating the fault distance in a line [11–13]. In this case, the ANN is usually applied to model the system and calculate the approximate distance of a fault

* Corresponding author.
E-mail addresses: chrystiands@gmail.com (C.D. Lana da Silva), adriano@ctism.ufsm.br (A.P. de Morais), gutomarchesan@gmail.com (G. Marchesan), fernandokg@gmail.com (F.G. Kaehler Guarda).

http://dx.doi.org/10.1016/j.epsr.2017.02.023

for several fault resistances. Some methods rely on more than just ANNs, applying other mathematical concepts such as the Wavelet transform [14] to enhance its performance.

Another application of ANNs on distance protection is as a pattern classifying algorithm [15,16]. This essentially transforms the ANN into a safety measure, indicating whether the fault is located inside or outside the protection zones. This kind of application is very common due to the adaptive and learning capabilities of ANNs.

It is also worth mentioning that impedance estimation using ANNs has been explored before [17] using a different approach. In this reference, the author proposed an offline trained ANN with training sets modelled via several data windows. This method, however, needs to be adjusted according to the power grid topology and parameters, which the author called pre-ANN step, in order to estimate the apparent impedance.

The method presented in this paper offers an alternative way to estimate the apparent impedance seen by the distance relay to rival the commonly used Fourier filter. It consists in decomposing the current and voltage signals, and using two separate continually online trained parallel ANNs to calculate impedance. This structure enables the method to work properly under off-nominal frequencies, a feature most commonly used algorithms lack, such as the Discrete Fourier Transform (DFT), which will be used for comparison.

2. The proposed method

In order to create the ANN's inputs and outputs, modelling of the system current and voltage is required. This modelling is accomplished considering initially a basic equation of a generic sampled sinusoidal signal $A(k)$, as shown in (1). The current and voltage signals $I(k)$ and $V(k)$ are identical to (1), but in the voltage equation the exponentially decaying DC offset is set to zero ($A_0 = 0$).

$$A(k) = A_1 \cos(\omega k\Delta t + \phi_1) + A_0 e^{-\frac{k\Delta t}{\tau}} \tag{1}$$

where k is the sample number, $A(k)$ is the instantaneous value at sample k, A_1 is the fundamental component amplitude, ω is the angular frequency, Δt is the time difference between samples, ϕ_1 is the fundamental component phase angle, and A_0 and τ are the exponentially decaying DC offset amplitude and time constant, respectively.

From (1), the cosine term is expanded and the exponential term is substituted by its Taylor series first-order equivalent. These modifications are shown in (2).

$$A(k) = A_1 \cos(\phi_1)\cos(\omega k\Delta t) - A_1 \sin(\phi_1)\sin(\omega k\Delta t) + A_0 - \frac{A_0 k\Delta t}{\tau} \tag{2}$$

Let us consider two constants, C_1 and C_2, shown in (3) and (4), respectively. By replacing these constants in (2), (5) can be obtained. Also, for simplicity, $\gamma = -A_0/\tau$.

$$C_1 = A_1 \cos(\phi_1) \tag{3}$$

$$C_2 = -A_1 \sin(\phi_1) \tag{4}$$

$$A(k) = C_1 \cos(\omega k\Delta t) + C_2 \sin(\omega k\Delta t) + A_0 + \gamma k\Delta t \tag{5}$$

Since the original equation (1) does not consider harmonic components, an additional step is required in order to include these components in the ANN. More coefficients are added to (5), up to the 5th harmonic, resulting in (6).

$$\begin{aligned} A(k) = {} & C_1 \cos(\omega k\Delta t) + C_2 \sin(\omega k\Delta t) + \cdots + C_9 \cos(5\omega k\Delta t) \\ & + C_{10} \sin(5\omega k\Delta t) + A_0 + \gamma k\Delta t \end{aligned} \tag{6}$$

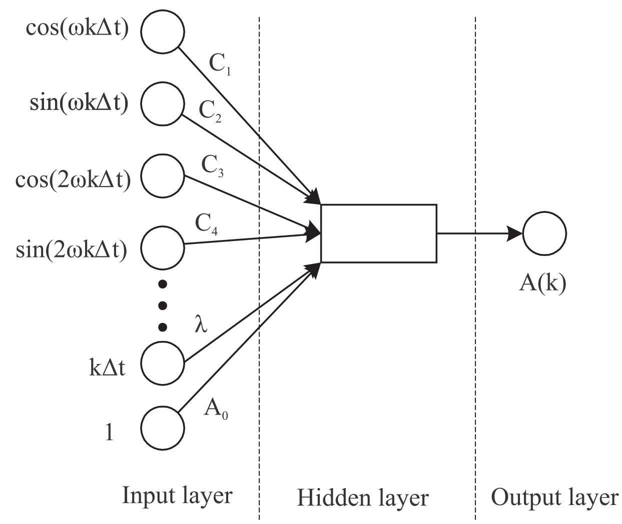

Fig. 1. ANN structure used in the simulations.

With this equation, it is possible to make sure any harmonic components are effectively removed, since higher order harmonics are naturally removed by the relay's lowpass filter. The filter used in this method is a second-order Butterworth filter set to 200 Hz. Eq. (6) can also be rewritten in its matrix form, as shown in (7).

$$\begin{bmatrix} \cos(\omega k\Delta t) & \sin(\omega k\Delta t) & \cdots & k\Delta t & 1 \end{bmatrix} \times \begin{bmatrix} C_1 \\ C_2 \\ \vdots \\ \gamma \\ A_0 \end{bmatrix} = A(k) \tag{7}$$

Eq. (7) can be written simply as shown in (8), where M, P and A are the input, weight, and output matrices, respectively.

$$M \times P = A \tag{8}$$

Matrix P carries all the values referencing amplitude and phase angle of each component present in the signal. Knowing that M carries the input information and A carries the output information, shown in (7), those matrices can be easily determined. Then, to calculate matrix P, the ANN training algorithm is used.

Fig. 1 shows the ANN structure created based on (7). Since current and voltage information are needed for impedance estimation, two identical structures are used, one for each variable, substituting $A(k)$ for $I(k)$ and $V(k)$ accordingly. The structure has 12 neurons in the input layer, 1 neuron in the output layer, and one hidden layer with 1 neuron. The number of neurons on each layer were chosen in order to make the ANN reproduce (7).

Since the inputs and outputs in (1), represented by M and A in (8), are known, it is possible to obtain the current and voltage fundamental phasors by determining the weight matrix P. The training is online, and there is no need for an offline step. Because of the continually online training, the ANN uses real time inputs and outputs, allowing it to constantly adapt as the system topology and frequency change. With the backpropagation training method, the ANN adjusts the input weights at each training step, so the fundamental phasors are also calculated in real time.

With this online training, differently from the traditional ANN usage, the goal here is not the output, but the weights between layers, represented by P in (8). These weights will be used later to calculate the amplitude and phase angle of the fundamental component. Since the only weights that concern the fundamental