

# Parameter Estimation of Double Exponential Pulse Based on Artificial Neural Network

Zhu Zhizhen<sup>1, a</sup>, Shi Yuewu<sup>1</sup>, Yang Jing<sup>1</sup>, Nie Xin<sup>1</sup>, Zhou Hui<sup>1</sup>

<sup>1</sup>State Key Laboratory of Intense Pulsed Radiation and Effect, Northwest Institution of Nuclear Technology, Xi'an, 710024, China

<sup>a</sup>email: zhuzhizhen@nint.ac.cn

**Keywords:** Double Exponential Function; High-altitude Electromagnetic Pulse (HEMP); Artificial Neural Network (ANN); Parameter Estimation; Estimation Error

**Abstract.** In the study of high-power electromagnetic environments, double exponential function is widely used. The physical parameters of the pulse are the rise time  $t_r$  and the pulse width  $t_w$ , which have a strong relation with  $\alpha$  and  $\beta$ , the key parameters of the double exponential function. In actual use, the transformation of  $t_r$ ,  $t_w$  and  $\alpha$ ,  $\beta$  is very necessary, but difficult. In this paper, a method based on artificial neural network is proposed. The parameter  $\alpha$  can be given out directly by the solution, and  $\beta$  is easy to calculate. Simulation is carried out for verification. The estimation errors are less than 2.10% for both  $\alpha$  and  $\beta$ .

## Introduction

Nowadays, electronic devices and systems face the threats of high-power electromagnetic environments, which include the high-altitude electromagnetic pulse (HEMP) and ultra-wide band (UWB) pulses. Many experiments are carried out to verify the survivability of systems, and harden methods are studied. Both in simulation and tests, the pulse shapes of these environments are very important.

The pulse shapes of the high-power electromagnetic environments [1][2] are mainly described by the rise time  $t_r$  and the pulse width  $t_w$ . The double exponential function is often used to approximate the pulse shapes, and  $\alpha$ ,  $\beta$  are the key parameters of the function. The relation between  $t_r$ ,  $t_w$  and  $\alpha$ ,  $\beta$  is very important. The transformation of the two groups of parameters is often needed. Different methods are used to estimate the relation [3][4]. However, the solutions are often complex for using. And too many linear functions are required for expressing the correlation.

The method based on Artificial Neural Network (ANN) is a very useful mathematical method for solving the problems which are nonlinear [5]. From supervised learning of the input-output relationship with some real data, ANN is trained. The trained model can then be used to give out the output from arbitrary input.

In this paper, we introduce ANN to solve the parameter-transformation problem of double exponential function. The parameters can be given out directly by the model. And the method is easy to understand and operate. Finally, the estimation errors are presented and analyzed.

## Shape of Double Exponential Pulse

The double exponential pulse is shown in equation (1).

$$f(t) = K(e^{-\alpha t} - e^{-\beta t}) \quad (1)$$

$$(t \geq 0)$$

where  $K$ ,  $\alpha$  and  $\beta$  are parameters which are independent of  $t$  (denotes time).  $\alpha$  and  $\beta$  should satisfy  $0 < \alpha < \beta$  to keep the function value positive. The shape of the pulse is shown in Fig.1.

In Fig.1, the rise time  $t_r$  denotes the time interval between the time at which the function value are 10% and 90% of the peak value on the rising edge. The pulse width  $t_w$  is the time interval between the time at which the function value are 50% of the peak value.

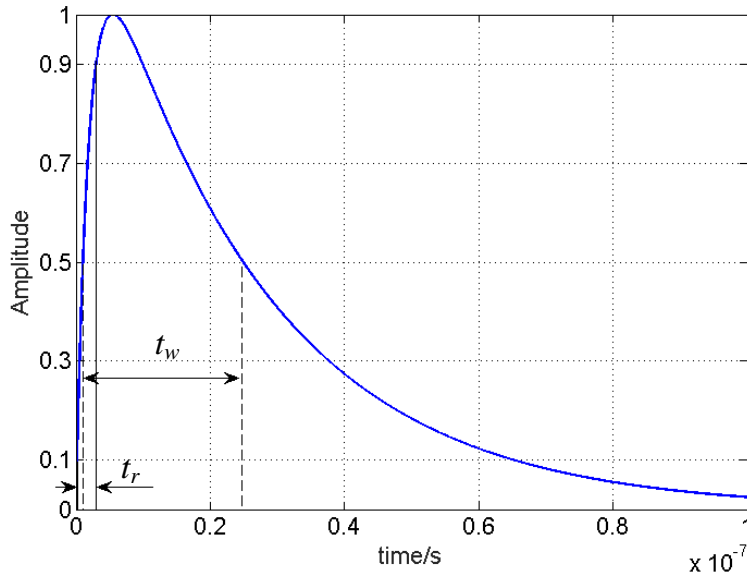


Fig.1. The shape of double exponential pulse

### Artificial Neural Network

The ANN based method is a useful method for non-linear problem modeling. With the knowledge of the input and output of a nonlinear system, ANN can be trained for representing the input-output relationship of the system. Figure 2 shows how ANN based method works.

The input and output of the real system are imported into the method. The real input is set as the input of ANN. The training error of ANN is got from the difference of the real output and the output of ANN. The training error is then used to revise the model, according to learning method.

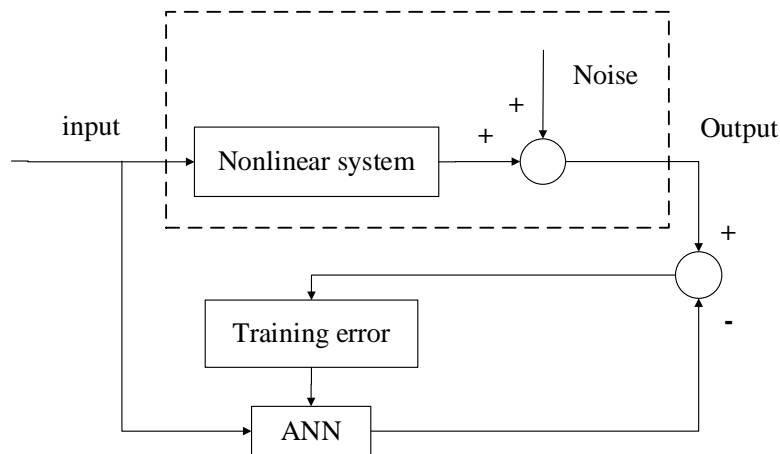


Fig.2. The modeling of ANN

As shown in Figure 3, ANN are comprised of a lot of nodes which are called artificial neurons. The nodes line up to form the layers. The nodes receive the signal from the nodes in the front layer through weighted connections ( $\omega_{ij}$ ). After create their outputs from all the inputs (using threshold  $d_j$  and transfer function  $\varphi_j(\cdot)$ ), the nodes then transmit the outputs to the nodes in the next layer. Following the way, the output signals are finally generated from the input signals. The hidden layers are the layers between the input layer and the output layer.  $\omega_{ij}$  are the key parameters of ANN. The learning rules modify  $\omega_{ij}$  from the difference between the outputs of ANN and the real outputs. And the procedure is so called learning.

The correction of  $\omega_{ij}$  is shown in equation (2).

$$\Delta\omega_{ij}(n) = \eta f[y_j - y'_j(n)] \tag{2}$$

where  $\eta$  is the learning rate,  $y_j$  is the real output from the training data, and  $y'_j(n)$  is the output data

from ANN. For different learning method,  $f(\cdot)$  usually differ from each other.

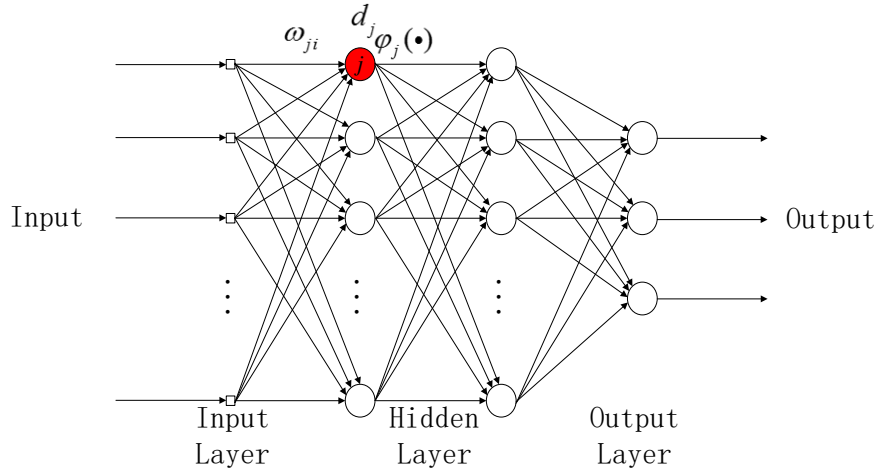


Fig.3. Back propagation model

### Modeling of the Parameter Estimation

ANN is introduced here to model the relation between  $t_r$ ,  $t_w$  and  $\alpha$ ,  $\beta$ . Back-propagation (BP) method is chosen as the learning rules. The architecture of back-propagation model used in this paper is shown in Figure 3.

The transfer function, shown in equation (3), is logistic sigmoid function, which is a common transfer function used in Back-Propagation model. The Levenberg-Marquardt (LM) algorithm [5] was employed for training.

$$y_j = \varphi\left(\sum_i \omega_{ij} a_i, d_j\right) = \frac{1}{1 + \exp[-(\sum_i \omega_{ij} a_i - d_j)]} \quad (3)$$

The training data of  $\alpha$  and  $\beta$  are sampled as follows:

(1) In order to decrease the training error and reduce the amount of training data, the whole range of the value of  $\beta$  is split into two parts with different intervals.

- ①  $\beta \in [1.1 \times 10^8, 10 \times 10^8]$ , the interval is  $1 \times 10^7$ .
- ②  $\beta \in [11 \times 10^8, 600 \times 10^8]$ , the interval is  $5 \times 10^8$ .

(2)  $\alpha \in [10^7, 10^8]$ , the interval is  $1 \times 10^7$ .

The ANN for  $\alpha$  is shown in Figure 4. It has three layers. They are one input layer, one hidden layer and one output layer. The amount of nodes in each layer are shown in Table 1.

Table 1. The amount of nodes in each layer of ANN $_{\alpha}$

	Input Layer	Hidden Layer	Output Layer
Nodes	2	30	1

The input training data of ANN $_{\alpha}$  is  $t_r \times 10^9$  and  $t_w \times 10^9$ . The processing of the data is similar to the input normalization. And the output training data of ANN $_{\alpha}$  is  $\alpha/10^7$ , as shown in Table 2.

Table 2. The training data of ANN $_{\alpha}$

Input signal 1	Input signal 2	Output signal
$t_r \times 10^9$	$t_w \times 10^9$	$\alpha/10^7$

The ANN for  $\beta/\alpha$  is shown in Figure 5. It also has three layers, one input layer, one hidden layer and one output layer. The amount of nodes in each layer are shown in Table 3. The input training data of ANN $_{\beta/\alpha}$  is  $t_w/t_r$ . And the output training data is  $0.1\beta/\alpha$ , as shown in Table 4.

Table 3. The amount of nodes in each layer of ANN <sub>$\beta/\alpha$</sub>

	Input Layer	Hidden Layer	Output Layer
Nodes	1	25	1

Table 4. The training data of ANN <sub>$\beta/\alpha$</sub>

Input signal	Output signal
$t_w/t_r$	$0.1\beta/\alpha$

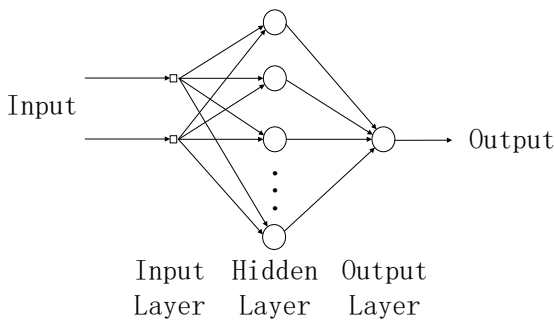


Fig.4. The ANN <sub>$\alpha$</sub>  for  $\alpha$

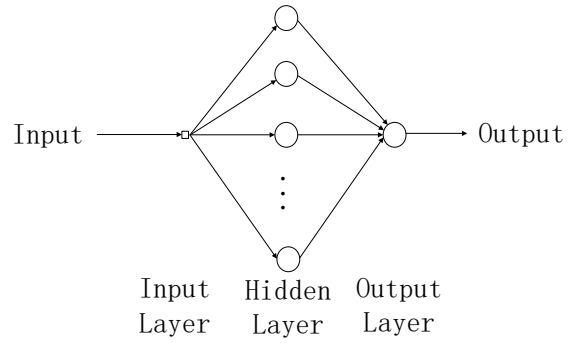


Fig.5. The ANN <sub>$\beta/\alpha$</sub>  for  $\beta/\alpha$

### Simulation and Results

With the training data of  $\alpha$  and  $\beta$  chosen follow the way above, the training data of  $t_w$  and  $t_r$  can be obtained from the analytic method. When the training is finished, random data is used to verify the performance of the ANN.

A sub-set of 10000 randomly chosen data, following uniform distribution, is used to test the fitting error. The ranges of the data are also:  $\alpha \in [10^7, 10^8]$  and  $\beta \in [1.1 \times 10^8, 600 \times 10^8]$ . With the chosen  $\alpha$  and  $\beta$ , the input of the two ANNs are given out from the analytic method.

The fitting errors can be calculated as follow:

$$e = \left| \frac{y - y'}{y} \right| \times 100\% \quad (4)$$

where  $y$  is the real value, and  $y'$  is the estimation value got from the ANN model.

The fitting errors are shown in Figure 6 and Figure 7. The maximum fitting error of  $\alpha'$  is 2.10%. And for  $(\beta/\alpha)'$ , it is 0.39%.

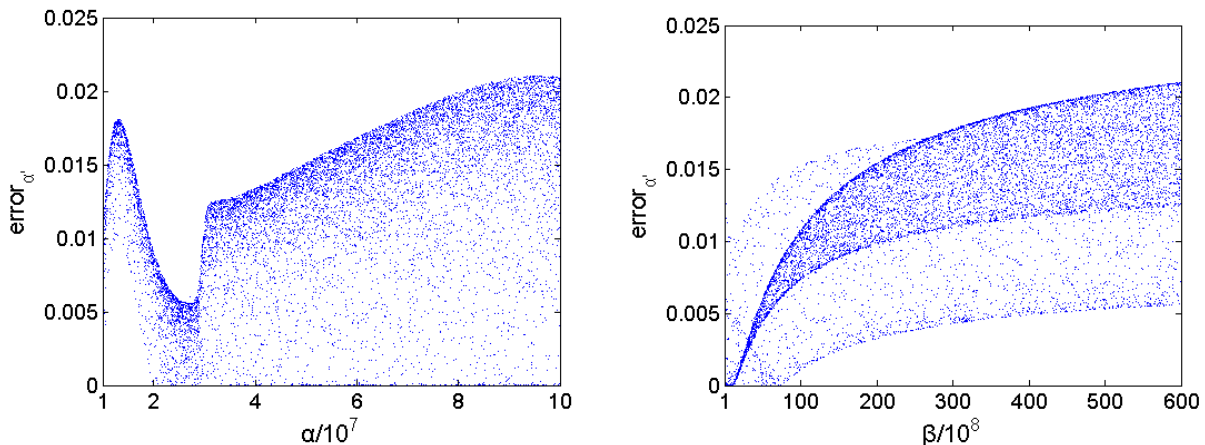


Fig.6. The fitting error of  $\alpha'$

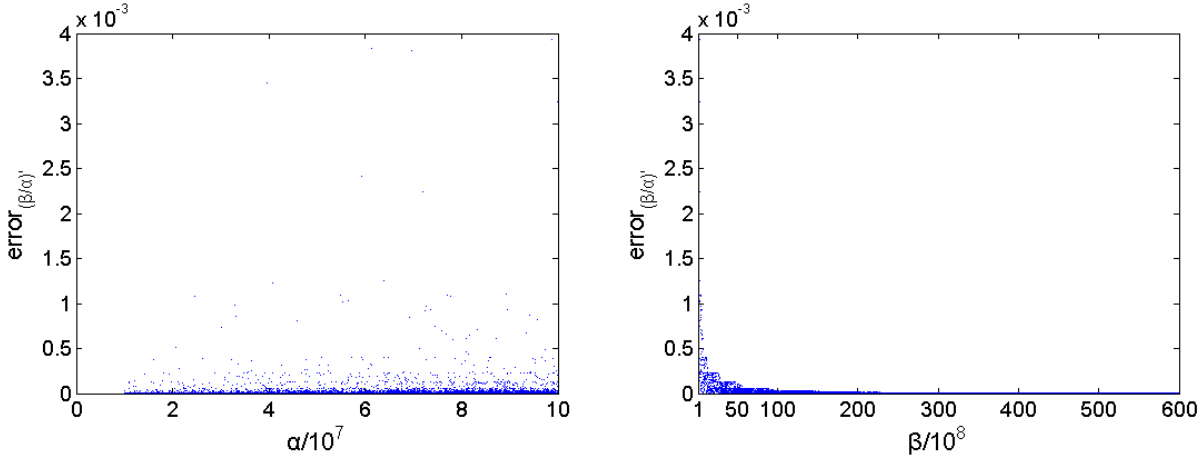


Fig.7. The fitting error of  $(\beta/\alpha)'$

$\beta'$  can then be calculated from  $\alpha'$  and  $(\beta/\alpha)'$ . As shown in (5) to (7), the fitting error of  $\beta'$  is below 2.50%.

$$\beta' = \left(\frac{\beta}{\alpha}\right)' \times \alpha'$$

$$\beta + \Delta_{\beta} \times \beta = \left(\frac{\beta}{\alpha} + \Delta_{\beta/\alpha} \times \frac{\beta}{\alpha}\right) \times (\alpha + \Delta_{\alpha} \times \alpha) \quad (5)$$

$$= \beta + \Delta_{\alpha} \times \beta + \Delta_{\beta/\alpha} \times \beta + \Delta_{\beta/\alpha} \times \Delta_{\alpha} \times \beta$$

So,

$$\Delta_{\beta, \max} \leq |\Delta_{\alpha, \max}| + |\Delta_{\beta/\alpha, \max}| + |\Delta_{\beta/\alpha, \max} \times \Delta_{\alpha, \max}| \quad (6)$$

Finally,

$$\Delta_{\beta, \max} \leq |\Delta_{\alpha, \max}| + |\Delta_{\beta/\alpha, \max}| + |\Delta_{\beta/\alpha, \max} \times \Delta_{\alpha, \max}|$$

$$= 2.10\% + 0.39\% + 0.39\% \times 2.10\%$$

$$\approx 2.50\% \quad (7)$$

In the simulation, the actual fitting error of  $\beta'$  is shown in Figure 8, and the maximum value is 2.10%. Because the fitting error is mainly determined by  $\alpha'$ , Figure 8 is very similar to Figure 6.

Figure 9 shows the fitting error in 2D distribution.

## Conclusion

In this paper, a new solution of the parameter estimation of the transformation of  $t_r$ ,  $t_w$  and  $\alpha$ ,  $\beta$ , which is based on ANN, is given out. With this method,  $\alpha$  and  $\beta/\alpha$  can be given out directly from  $t_r$  and  $t_w$ .  $\beta$  can then be calculated easily. The actual fitting errors are all less than 2.10%.

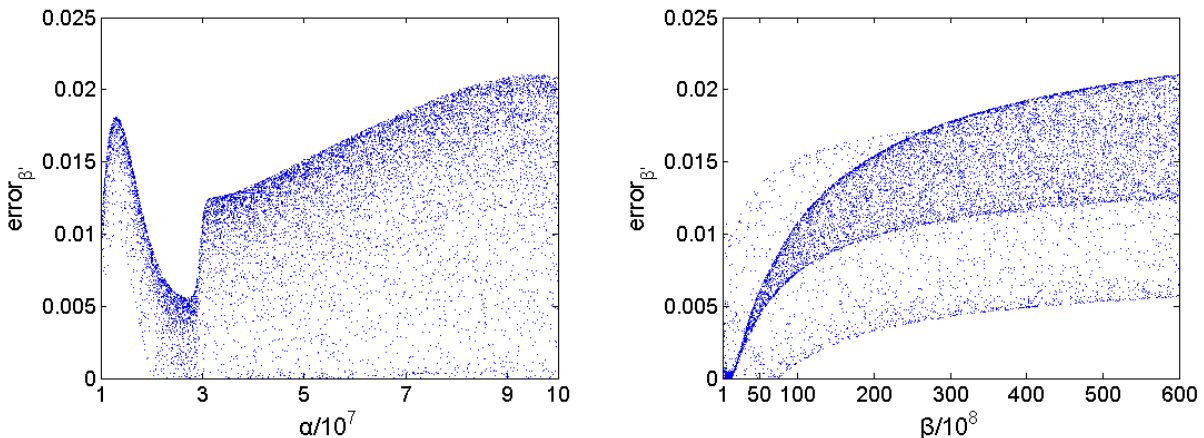


Fig.8. The fitting error of  $\beta'$

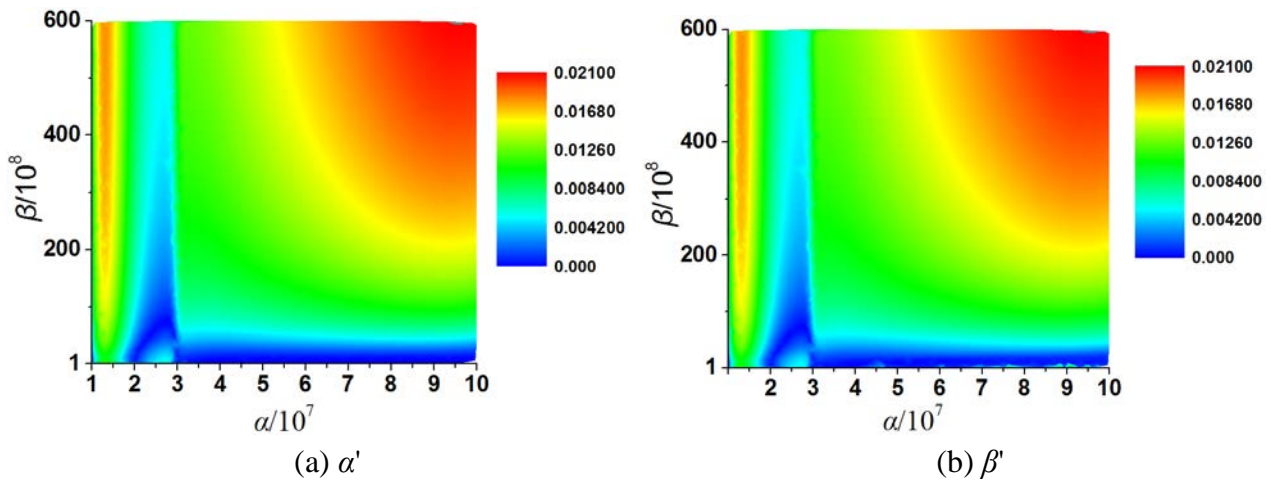


Fig.9. The fitting errors of  $\alpha'$  and  $\beta'$  in two-dimensional distribution

### Acknowledgement

In this paper, the research was sponsored by the Fundamental Research Project of State Key Laboratory of Intense Pulsed Radiation and Effect (Project No. SKLIPR1404).

### References

- [1] Electromagnetic Compatibility (EMC)-Part 2: Environment—Section 9: Description of HEMP Environment—Radiated Disturbance, IEC 61000-2-9 Ed. 1.0, 1996–2002.
- [2] Yanzhao Xie. A modified description of early time high-altitude electromagnetic pulse waveform (E1), Sensor and Simulation Note 562, Univ. New Mexico, Albuquerque, NM, USA, Jan. 2013.
- [3] Gang Wu. Shape Properties of Pulses Described by Double Exponential Function and Its Modified Forms [J], IEEE Trans. Electromagn. Compat., vol. 56, no. 4, pp. 923–931, Feb. 2014.
- [4] Congguang Mao, Hui Zhou. Novel parameter estimation of double exponential pulse (EMP, UWB) by statistical means [J], IEEE Trans. Electromagn. Compat., vol. 50, no. 1, pp. 97–100, Feb. 2008.
- [5] Zhongzhi Shi. Neural Networks [M], Beijing: Higher Education Press, 2009.