



# Recovery of saturated signal waveform acquired from high-energy particles with artificial neural networks

Yu Liu<sup>1</sup> · Jing-Jun Zhu<sup>2</sup> · Neil Roberts<sup>3</sup> · Ke-Ming Chen<sup>1</sup> · Yu-Lu Yan<sup>1</sup> · Shuang-Rong Mo<sup>2</sup> · Peng Gu<sup>1</sup> · Hao-Yang Xing<sup>1</sup>

Received: 1 February 2019 / Revised: 2 May 2019 / Accepted: 9 May 2019 / Published online: 30 September 2019  
© China Science Publishing & Media Ltd. (Science Press), Shanghai Institute of Applied Physics, the Chinese Academy of Sciences, Chinese Nuclear Society and Springer Nature Singapore Pte Ltd. 2019

**Abstract** Artificial neural networks (ANNs) are a core component of artificial intelligence and are frequently used in machine learning. In this report, we investigate the use of ANNs to recover the saturated signals acquired in high-energy particle and nuclear physics experiments. The inherent properties of the detector and hardware imply that particles with relatively high energies probably often generate saturated signals. Usually, these saturated signals are discarded during data processing, and therefore, some useful information is lost. Thus, it is worth restoring the saturated signals to their normal form. The mapping from a saturated signal waveform to a normal signal waveform constitutes a regression problem. Given that the scintillator and collection usually do not form a linear system, typical regression methods such as multi-parameter fitting are not immediately applicable. One important advantage of ANNs is their capability to process nonlinear regression problems. To recover the saturated signal, three typical ANNs were tested including backpropagation (BP), simple recurrent

(Elman), and generalized radial basis function (GRBF) neural networks (NNs). They represent a basic network structure, a network structure with feedback, and a network structure with a kernel function, respectively. The saturated waveforms were produced mainly by the environmental gamma in a liquid scintillation detector for the China Dark Matter Detection Experiment (CDEX). The training and test data sets consisted of 6000 and 3000 recordings of background radiation, respectively, in which saturation was simulated by truncating each waveform at 40% of the maximum signal. The results show that the GRBF-NN performed best as measured using a Chi-squared test to compare the original and reconstructed signals in the region in which saturation was simulated. A comparison of the original and reconstructed signals in this region shows that the GRBF neural network produced the best performance. This ANN demonstrates a powerful efficacy in terms of solving the saturation recovery problem. The proposed method outlines new ideas and possibilities for the recovery of saturated signals in high-energy particle and nuclear physics experiments. This study also illustrates an innovative application of machine learning in the analysis of experimental data in particle physics.

---

This work is supported by the “Detection of very low-flux background neutrons in China Jinping Underground Laboratory” project of the National Natural Science Foundation of China (No. 11275134).

---

✉ Hao-Yang Xing  
xhy@scu.edu.cn

<sup>1</sup> School of Physical Science and Technology, Sichuan University, Chengdu 610064, China

<sup>2</sup> Key Laboratory of Radiation Physics and Technology of Ministry of Education, Institute of Nuclear Science and Technology, Sichuan University, Chengdu 610065, China

<sup>3</sup> Edinburgh Imaging, School of Clinical Sciences, University of Edinburgh, Edinburgh, UK

**Keywords** Saturated signals · Artificial neural networks (ANNs) · Recovery of signal waveform · Generalized radial basis function · Backpropagation neural network · Elman neural network

## 1 Introduction

In particle physics experiments, various inherent properties of the detection medium, photomultiplier tube (PMT), range of analog-to-digital conversion, and gain circuitry sometimes result in the saturation of the signals associated with high-energy particles. These saturated signals are often discarded during the processing of the experimental data, and therefore, potentially useful information is lost. There has been notable research activity with respect to the restoration of saturated signals including the application of statistical techniques based on maximum likelihood analysis or minimum Chi-squared fitting, and pulse analysis based on the identification of measurable values that are related to the appropriate physical variables [1]. However, the efficiency of the former approach is relatively low, especially when the saturation is severe. The latter technique is adequate in terms of energy recovery, but it is not ideal for waveform recovery. Examples of related studies include the work of Xie et al. [2] that involved the recovery of spurious signals via linear extrapolation in the frequency domain and that of Huang et al. [3] on the recovery of unknown weak signals via polynomial fitting and the variance of the time difference sequence. Among these approaches, the most typical and most likely to be implemented in a practical setting is the use of a signal model or normalization to fit the saturated signals [4, 5]. However, the nonlinearity of the system waveforms and the conversion of the deposited energy may result in differences between the signals recovered via the aforementioned method and the actual unsaturated signals. These differences will affect waveform discrimination and the application of accurate energy information. However, there is an alternative method that can avoid the saturation of waveforms based on a direct measurement of the amplitude using the time-over-threshold technique [6]. However, this results in a loss of waveform information.

In terms of mathematical theory, the recovery of saturated signals is essentially a regression problem of predicting global data from partial data. To solve such a regression problem, apart from model fitting and statistical regression, machine learning in the form of artificial neural networks (ANNs) can be utilized. To date, some researches have been conducted on the application of artificial neural networks to perform multivariate analysis to address problems in particle physics [7]. A neural network consists of simple processing nodes or neurons that are interconnected to each other in a specific order. After training and the determination of the weight of each node, the neural network can classify and fit complex functions. Moreover, compared with traditional fitting methods, the artificial neural network has advantages that include the following:

it requires less formal statistical training, can implicitly detect complex nonlinear relationships between dependent and independent variables, and can detect all possible interactions between predictor variables. Moreover, multiple training algorithms are available. There have been several studies in which ANNs have been employed to conduct multivariate analysis in particle physics [7]. For nonlinear regression problems, ANNs can model the inherent nonlinear relationships and utilize nonlinear data [8].

In previous particle physics experiments, ANNs have been used to discriminate between neutrons and  $\gamma$ -rays in a liquid scintillation detector [9], spectrometry analysis [10], reconstruction of the neutron spectrum [11, 12], discrimination of signals from the background [13], optimization of the concrete mixture for a thermal neutron shield [14], classification of gravitational-wave signals [15], treatment of intricate quantum many-body problems [16], energy-dispersive X-ray fluorescence analysis [17], and the prediction of the intensity of solar radiation using neuro-fuzzy inference systems [18]. In this study, we investigate the possibility of using ANNs to recover saturated particle detection signals. Several different neural networks were used to test the performance of saturated signal recovery including backpropagation (BP), simple recurrent (Elman), and generalized radial basis function (GRBF) neural networks (NNs). Backpropagation neural networks represent a basic network structure and are the most widely used and the most mature training algorithm. The Elman neural network represents a network structure with feedback and has a memory function in addition to dynamic expression capability. As such, the results of each learning step contain all the previously learned contents. The GRBF neural network is a network structure with a kernel function and can be used to map low-dimensional data to high-dimensional spaces for further analysis. Although other neural networks may be feasible for saturated signal recovery, the aforementioned three network types were considered as representative models. To the best of our knowledge, no attempt has been made to recover complete signals from saturated signals using machine learning. This work is the first study to apply ANNs to the problem of signal waveform processing.

## 2 Data sources

The data in this research were obtained from a liquid scintillation detector consisting of an organic liquid scintillator (EJ-335, produced by Eljen Technology [19]) and 8-inch-diameter PMTs (HamamatsuR5912-02). The data acquisition system (DAQ) consisted of an analog-to-digital converter (FADC,MOD.V1721) operating at a sampling

frequency of 500 MHz and an accuracy of 8 bits. The data acquisition site was a deep underground laboratory named the China Jinping Underground Laboratory (CJPL). It is the world's deepest underground laboratory at present and is covered by 2400 m of rock [20]. The detected signals include background radiation. The sampled waveforms are transmitted via an optical fiber and stored on a computer. Few saturated signals were observed during the data acquisition period of this study. In addition, waveforms with different energies have a nonlinear relation with the signal amplitude [21]. In this case, the waveforms of different neutrons or gamma rays were added to the data set without any discrimination. This report focuses on the algorithm for the reconstruction of saturated signals using ANNs. As such, the reason for the saturation of the signals and the nonlinearity of the waveforms is not discussed.

### 3 Construction of training set and test set for the ANNs

The original data from the detectors were processed via a preliminary exclusion step. Given that signals with two or more peaks can cause the method to fail, these signals were removed. Signals in the low-energy range cannot be properly distinguished from noise, and therefore, they were also eliminated. The inclusion of too many sampling points will increase the computational load of the neural network and may lead to local convergence. Therefore, under the condition that the integrity of the signal is ensured, the range from 60 ns before the maximum sampling point of the signal to 120 ns after the maximum sampling point for the entire 480 ns sampling range of the ADC was selected as the sampling range of the neural network.

After the initial processing of the original data, the training and test sets were constructed. ANNs are suitable for approximating complex relationships between input and output variables, including nonlinear optimization [22]. Therefore, saturated signals with continuous energy distributions and corresponding complete waveforms were used as input and output, respectively, to train the ANN to automatically recover saturated signals waveform from the test set. The raw data were divided into two parts according to the energy of the signals. Thirty percent of the data with high energy values were assigned to the test set, and the remaining seventy percent 70% were placed into the training set. Given that the amplitude and the actual shape of the saturated signals are not known, artificial saturated signals were generated so that the complete waveforms recovered by the ANN could be compared with the original waveforms. Prior to the generation of the saturated signals, the cutoff value of the saturation was determined. The largest signal in the

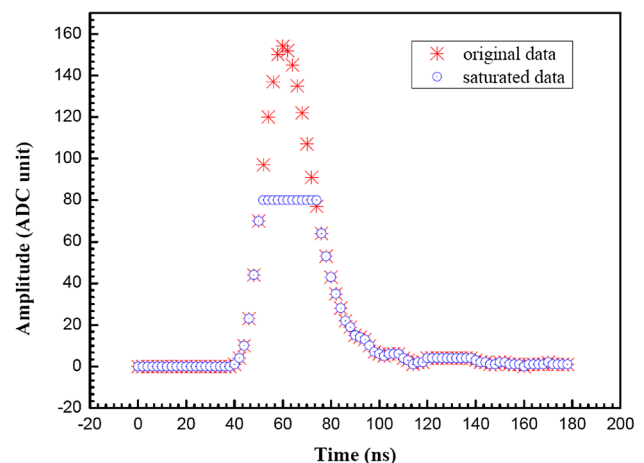
training set was selected, and the cutoff value was set to 40% of the amplitude of this signal. In this study, saturated signals were generated by truncating the signal in the peak region of standard waveforms and replacing the omitted region with a horizontal linear trace. An example is shown in Fig. 1.

The choice of 40% of the amplitudes cutoff value for high cutoff values reduces the number of saturated signals in the training set, whereas the utilization of low cutoff values reduces the number of effective sample points in a waveform. The use of both high and low cutoff values affects the training of the neural network.

According to this process, the original signals were truncated to generate the saturated signals including in the training set. The same procedure was repeated for the test set.

### 4 Neural networks considered in this work

The performance of different ANNs was investigated. For practical use, the ANNs must satisfy two important criteria: the output (i.e., recovered) waveform must be as close as possible to the original waveform in terms of both the shape and magnitude. Given that the presented work is similar to signal fitting, it is important to examine the degree of compliance between the original signal and the recovered signal. Thus, a deformation of the Chi-squared test (the evaluation benchmark for signal fitting) was used to evaluate the performance of the ANNs considered in this work. A comparison based on this benchmark can identify the method with the best reliability and accuracy. In particular, the value of the corrected Chi-squared statistic (CCS) was computed as follows:



**Fig. 1** Complete signal and the corresponding cutoff signal. The red points represent the original data for a gamma signal, and the blue points represent the corresponding saturated signal after cutoff

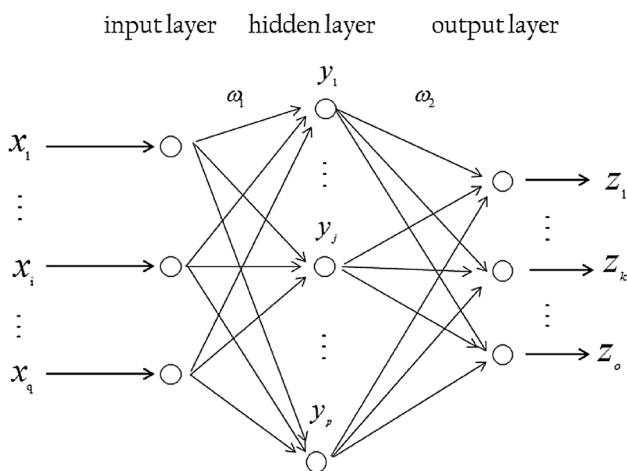
$$ccs = \frac{\sum_{i=b}^{N-f-b} (m_i - m'_i)^2 / m_i}{f - b}, \tag{1}$$

where  $m_i$  represents the value of the  $i$ th sampling point in the original signal,  $m'_i$  represents the value of the  $i$ th sampling point in the recovered signal,  $N$  represents the total number of sampled points,  $b$  represents the number of the first sampling point in the cutoff region, and  $f$  represents the number of the last sampling point of the cutoff region. This equation differs from the standard Chi-squared test in that an extra factor of  $\frac{1}{f-b}$  is added to consider the variation in the cutoff range with the signal energy.

### 4.1 Backpropagation neural network

Backpropagation neural networks (BP-NNs) are the most basic and widely used class of ANNs. They have a hierarchical feed-forward structure in which the outputs of each layer are sent directly to every neuron of the next layer. There are at least three layers and sometimes several more. The structure of a typical three-layer BP-NN is shown in Fig. 2.

The core idea of BP-NN is to distribute the output error to all units of each layer using some form of back-transmission from the hidden layer to the input layer. In this way, it is possible to correct the weight of each unit. Therefore, the learning process of BP neural network includes two sub-processes: the forward propagation of the signal and the backpropagation of the error. The process continues until the network output error has decreased to an acceptable level or a preset number of learning steps have been reached.



**Fig. 2** Topology of a three-layer BP-NN. The nodes of the input layer ( $x_i$ ) and the hidden layer ( $y_j$ ) are connected one by one with a weight of  $\omega_1$  and a transfer function, and the nodes of the hidden layer ( $y_j$ ) and the output layer ( $z_k$ ) are connected one by one with a weight of  $\omega_2$  and a different transfer function

If the input layer has  $p$  neurons, the hidden layer has  $q$  neurons, and the output layer has  $o$  neurons, the input vector and expected output vector can be expressed as follows:

$$x(m) = x_1(m) + x_2(m) + \dots + x_i(m) + \dots + x_q(m), \tag{2}$$

$$t(m) = t_1(m) + t_2(m) + \dots + t_i(m) + \dots + t_o(m). \tag{3}$$

The relationship between the input layer and the input vector of the hidden layer can be expressed as Eq. (4):

$$y_{in}(m) = \sum_{i=1}^p \omega_{1i}x_i(m) - b_y, \tag{4}$$

where  $\omega_{1i}$  is the weighting factor between the input layer and the hidden layer and  $b_y$  is the threshold of the hidden layer neurons. The output of the hidden layer after calculating the activation function can be expressed as Eq. (5):

$$y_{out}(m) = f_1(y_{in}(m)), \tag{5}$$

where  $f_1(\cdot)$  is the activation function selected for the hidden layer. The activation function needs to be differentiable everywhere; the ‘‘sigmoid’’ function is often used.

The input and output vectors of the output layer are the same as those of the hidden layer.

The error between the expected output vector and the actual output vector of the network can be defined as follows:

$$E(m) = \frac{1}{2} \sum_{i=1}^o (t(m) - z_{out}(m))^2. \tag{6}$$

After the forward propagation of the network, the connection weights between the hidden layer and the output layer ( $\omega_2$ ) need to be corrected according to the error between the expected output and the actual output. This process typically uses some form of the gradient descent method, such as Eq. (7):

$$\begin{aligned} \Delta\omega_2 &= -\eta \frac{\partial E}{\partial \omega_2} = -\eta \frac{\partial E}{\partial z_{in}} \frac{\partial z_{in}}{\partial \omega_2} = \eta(t - z_{out})f'(z'_{in}) \frac{\partial z_{in}}{\partial \omega_2} \\ &= \eta(t - z_{out})f'(z'_{in})y_{out}, \end{aligned} \tag{7}$$

where  $\eta$  is the preset learning efficiency, which is set in advance and used to adjust the speed of learning.

The same approach is then applied to correct the connection weights between the hidden layer and the output layer ( $\omega_1$ ) as follows:

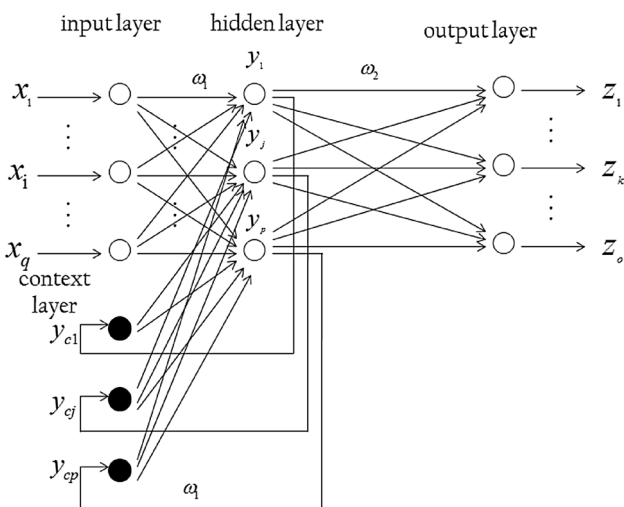
$$\begin{aligned} \Delta\omega_1 &= -\eta \frac{\partial E}{\partial \omega_1} = -\eta \sum_q \sum_o \frac{\partial E}{\partial z_{in}} \frac{\partial z_{in}}{\partial y_{out}} \frac{\partial y_{out}}{\partial z_{\omega_1}} \\ &= \eta \sum_o (t - z_{out})f'(y'_{in})\omega_{2f}(z'_{in})x. \end{aligned} \tag{8}$$

The forward and backpropagation processes are repeated until the output error has decreased to an acceptable level or a preset number of learning steps have been reached.

### 4.2 Elman neural network

The Elman NN is based on BP-NN, but it includes a feedback loop. The version of the Elman NN used in the present study is a partially recurrent network and lies somewhere between a classic feed-forward perception network and a purely recurrent network. The feed-forward loop consists of an input layer, a hidden layer, and an output layer. In addition, the weights that connect two neighboring layers can be varied. In contrast to the traditional feed-forward loop, the back-forward loop includes a context layer that is sensitive to the history of the input data and has fixed connections between the context layer and the hidden layer [23]. The context layer receives feedback signals from the hidden layer, whereby it can “remember” the output value of hidden layer neurons from previous time steps. The output of the context layer neurons is delayed and stored before being inputted to the hidden layer. This causes the neural network to be sensitive to historical data and increases its ability to process dynamic information. The structure of a typical three-layer Elman NN is shown in Fig. 3.

Apart from the addition of a connection layer, the Elman network operates similar to the BP network. Thus, in the



**Fig. 3** Topology of a three-layer Elman NN. The nodes of the input layer ( $x_i$ ) and the context layer ( $y_{ci}$ ) are connected with the nodes of the hidden layer ( $y_j$ ) one by one by with a weight of  $\omega_1$  and a transfer function, and the nodes of the hidden layer ( $y_j$ ) and the output layer ( $z_k$ ) are connected one by one with a weight of  $\omega_2$  and another transfer function

forward propagation step, only the input and output of the hidden layer are different from that of BP-NN.

$$y_{in}(m) = \sum_{i=1}^p (\omega_{1i}x_i(m) + \omega_{1ci}y_{ci}(m)), \tag{9}$$

where  $y_{ci}(m) = y_{out}(m - 1) = f_1 \left( \sum_{i=1}^p (\omega_{1i}x_i(m - 1) + \omega_{1ci}y_{ci}(m - 1)) \right)$  is the output of the context layer.

The values of  $\omega_2$  and  $\omega_1$  are modified in the same manner as for BP-NN, and the modified value of  $\omega_{1c}$  can be expressed as Eq. (10):

$$\Delta\omega_{1c}(m) = -\eta \frac{\partial E}{\partial \omega_1} = \eta \sum_0 (t - z_{out}) f(y'_{in}) \omega_2 \frac{\partial y_{out}(m)}{\partial \omega_2}, \tag{10}$$

where  $\frac{\partial y_{out}(m)}{\partial \omega_2} = f(z'_{in})x(m - 1) + \alpha \frac{\partial y_{out}(m-1)}{\partial \omega_2}$ . This demonstrates that each adjustment of the Elman NN relies on all previous training.

These modifications to the Elman NN improve the accuracy of the training process and make it easier to achieve the training purpose. However, as the size of the context layer increases, the number of calculations and the training time also increases.

### 4.3 Generalized radial basis function neural network

The generalized radial basis function neural network (GRBF-NN) is derived from the radial basis function neural network (RBF-NN). The RBF-NN usually consists of a single feed-forward input layer and a single hidden layer, and the output takes the form of an RBF. As the kernel function, the RBF maps low-dimensional data to a high-dimensional space. This projection of complex pattern classification problems to higher-dimensional space via the kernel function makes them more likely to be linearly separable. GRBF-NN also can map low-dimensional data to high-dimensional space. Therefore, certain rules that cannot be found in low-dimensional data may appear once the data are mapped to more dimensions.

The RBF-NN has the major advantages of being simple to design, generalizable, robust, and tolerant to input noise [24]. Unlike BP-NN and Elman NN, it is not necessary to construct the structure of the network: this is automatically determined based on the input and output data. As a kind of RBF-NN, GRBF-NN is relatively computationally simple, provides high model accuracy, avoids local minima during training, exhibits fast convergence and robustness [25, 26], and is predicted to be best suited to solve the problem of reconstructing saturated signals.

For GRBF-NN, the input layer and the hidden layer perform the same operation as in a normal RBF-NN. However, in the GRBF-NN, the output is produced by weighting and averaging of the hidden layer. The structure of a typical three-layer GRBF-NN is shown in Fig. 4.

These RBF units are usually probability density functions, such as the Gaussian function expressed in Eq. (11):

$$\varphi(r) = e^{-r^2/2\sigma^2}, \tag{11}$$

where  $\sigma$  is referred to as the extension or width of the basis function. The smaller this width, the more selective is the RBF. In a GRBF-NN, the independent variable is the Euclidean distance between the input vector and the center vector. Thus,  $r$  can be replaced by  $\|dist\|$ , and Eq. (11) is modified to give Eq. (12):

$$\varphi_i(X) = e^{-\|X-c_i\|^2/2\sigma_i^2}, \tag{12}$$

where  $c_i$  is one of the centers of the radial function in the input space,  $\|X - c_i\|$  is the Euclidean norm of the centered vector  $X$ , and  $\sigma_i$  is the parameter that controls the dispersion of  $\varphi_i(X)$  around its center.

When a GRBF-NN has a multi-dimensional input, a generalization of Eq. (12) is used that includes different dispersion parameters for each dimension of the input space. This is expressed as Eq. (13):

$$\varphi_i(X) = e^{-\sum_i^{-1} \cdot (X-c_i)}, \tag{13}$$

where  $\sum_i^{-1} = \text{diag}(\sigma_{i,1}^2, \sigma_{i,2}^2, \dots, \sigma_{i,n}^2)$ . Radial symmetry is lost in this case, but this is preferable in problems where the values of the input vector cover different dynamic ranges.

Finally, the output result of an RBF-NN can be expressed as Eq. (14):

$$Y(X_i) = \sum_{j=0:k} \omega_{i,j} e^{-\|X-c_i\|^2/2\sigma_i^2}, \tag{14}$$

where  $Y(X_i)$  is the output result of the RBF-NN and  $\omega_{i,j}$  is the connection weight between the hidden layer and the output layer of the RBF-NN.

Using Eq. (13) to replace the kernel function in Eq. (14), we obtain the result for the GRBF-NN, which is expressed as Eq. (15):

$$Y(X_i) = \sum_{j=0:k} \omega_{i,j} e^{-\sum_i^{-1} \cdot (X-c_i)}. \tag{15}$$

By definition, the regression of a dependent variable  $Y$  on an independent variable  $X$  estimates the most probable value for  $Y$ , given  $X$  and a training dataset. The regression method will produce the estimated value of  $Y$  that minimizes the mean squared error (MSE). GRBF-NN is a method for estimating the joint probability density function (PDF) of  $X$  and  $Y$  given only a training dataset. The system is perfectly general because the PDF is derived from the data with no preconceptions about its form.

In this work,  $X$  refers to the sampled points of the saturated waveform and  $Y$  refers to the sampled points of the recovered waveform. The basis function corresponds to the Euclidean distance between the sampled point of the input saturated waveform and the data center as the independent variable of the function. Using the “newgrnn” function in MATLAB 2017 [27] and inputting the training and test datasets, the function automatically calculates the hidden nodes of the network, the data center and extension constant of the RBF, and the modified output weights of the GRBF-NN.

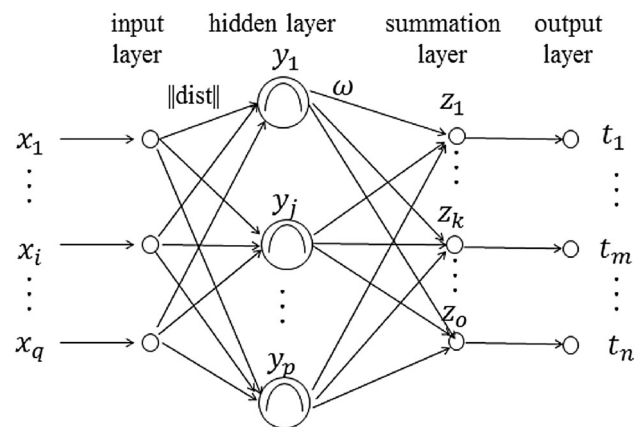
## 5 Results and Discussion

The performance of the various ANNs in recovering the saturated signal waveforms is described in the following section. All neural network algorithms were executed using MATLAB.

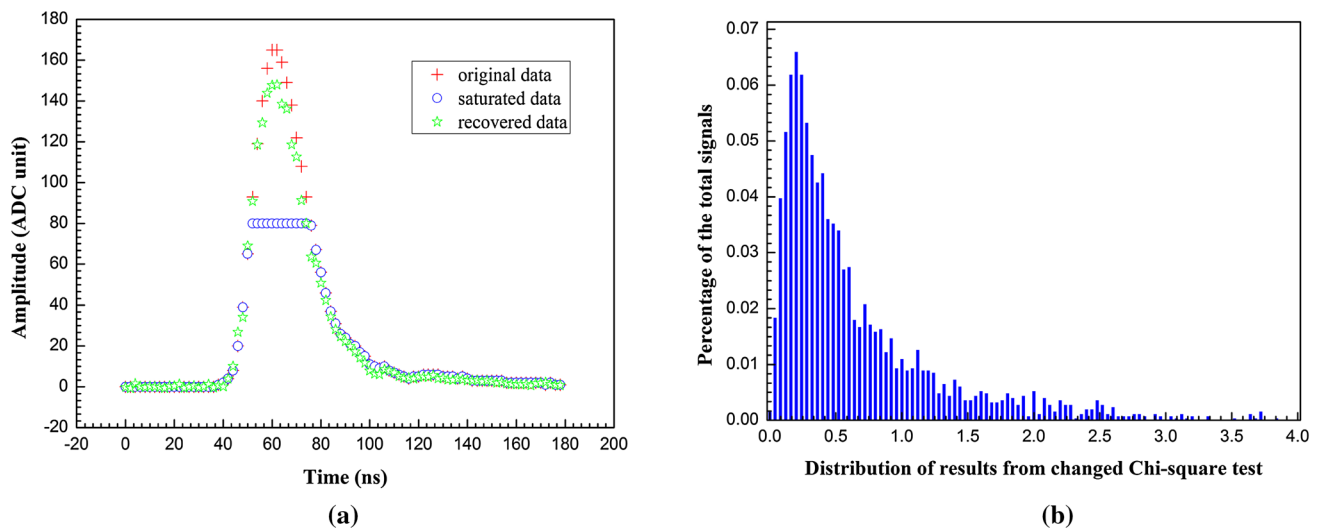
### 5.1 Results for backpropagation neural network

The results obtained by applying BP-NN (with two hidden layers of  $400 \times 400$  nodes, “tansig” as the transfer function, and 5000 training epochs performed in 565 s) to recover the saturated signal waveforms are shown in Fig. 5.

Apart from the output from the BP-NN, the corresponding input that was arbitrarily selected from the test dataset and the original signals before cutoff are shown for comparison. It is evident that the amplitude of the recovered signal waveform is different from that of the original



**Fig. 4** Neuronal structure of GRBF-NN. The nodes of the input layer ( $x_i$ ) and the hidden layer ( $y_j$ ) are connected one by one according to the Euclidean distance between the input vector and the center vector, and the nodes of the hidden layer and the output layer ( $t_m$ ) are connected one by one with a weight of  $\omega$  and a summation process ( $z$ )



**Fig. 5** Illustration of the performance of BP-NN. **a** Red points refer to the original waveform of the gamma signal, blue points refer to the corresponding saturated signal after application of the cutoff, and the

waveform. For other typical recovered waveforms, there are discontinuities and random fluctuations near the maximum value, and the connection between the sampling points is not sufficiently smooth. Generally, the output from BP-NN does not have the shape of the corresponding original signal. Although the weighting of the BP-NN can be adjusted during training, it is still difficult to determine the most appropriate value.

## 5.2 Results for Elman neural network

The results obtained by applying the Elman NN (with two hidden layers of  $400 \times 400$  nodes, “tansig” as the transfer function, and 5000 training epochs in 3712 s) to recover the saturated signal waveforms in a dataset arbitrarily selected from the test dataset are shown in Fig. 6.

The Elman NN performed better than BP-NN, but the results are still not consistent with the shape of the original signal, especially near the maximum value. Some improvements in performance could be obtained by modifying the network parameter, but the difference in peak position remains. Further improvements may be possible, but there is no theory to guide the adjustments and the convergence speed of the Elman NN is unfortunately slow. The learning efficiency of the Elman NN is another limitation on the recovery of saturated signals.

## 5.3 Results for generalized radial basis function neural network

The results for the application of the GRBF-NN (with the spread of the RBF set to 0.14) for recovery of the

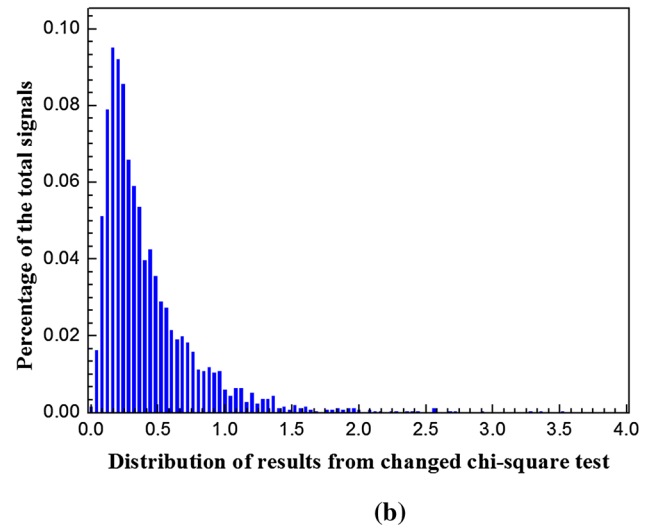
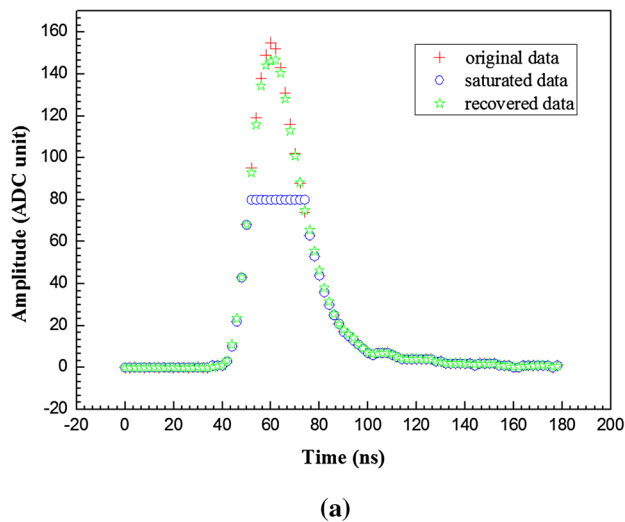
green points refer to the waveform recovered using a four-layer BP-NN. **b** Distribution of the results of the CCS test between the original and recovered waveforms for BP-NN. (Color figure online)

saturated signal waveforms in a dataset arbitrarily selected from the test dataset are shown in Fig. 7.

By adjusting the parameters of these three neural networks to optimize their performance and comparing the results from the CCS test, it is apparent that GRBF-NN is superior to Elman NN and BP-NN in terms of reliability and accuracy. The fitting curve is essentially identical to the original signal waveform, except that the maximum value is slightly and negligibly different. In most cases, GRBF-NN is successful and suitable for restoration of the saturated signal waveforms.

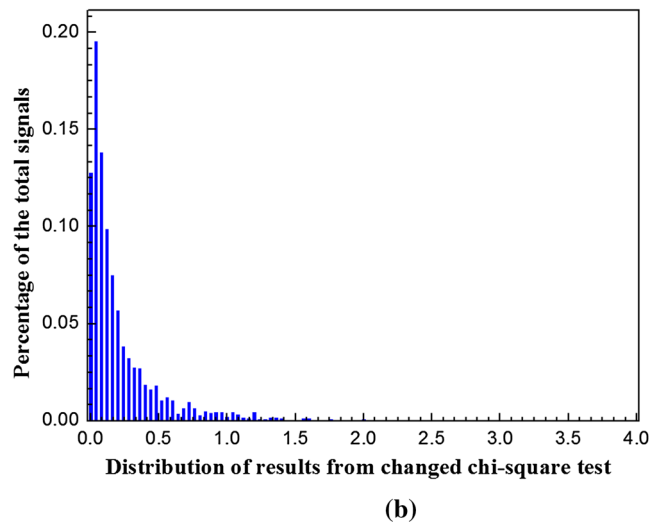
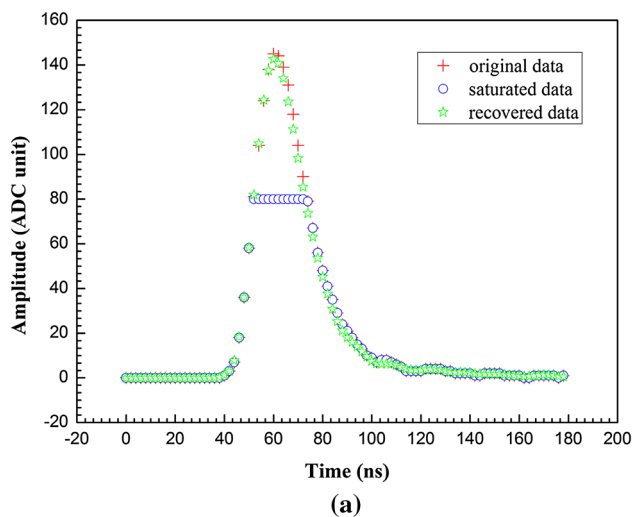
A comparison of the previous three figures indicates that the signal waveforms recovered by GRBF-NN are much better than those obtained using the other two NNs. The CCS test results are also better. GRBF-NN not only performed well but also has a simple structure and fast training speed. It has a solid theoretical basis, a concise mathematical form, and intuitive geometry and can solve small sample, nonlinear problems without becoming stuck near local minima. Thus, it is widely used in data approximation and regression analysis. Moreover, GRBF-NN has proved to be a universal approximate, and it is considered the most powerful type of network for highly nonlinear systems [28]. In this work, the investigated problem is highly nonlinear. Therefore, it is reasonable that GRBF-NN can solve this problem and produce the best results.

In recent years, there have been rapid developments in deep learning. As a class of deep neural networks, convolutional neural networks (CNNs) are widely used and have demonstrated their ability to solve many complex problems with high dimensionality and large-scale data such as image segmentation, object detection, and object verification. Solving these problems typically requires a



**Fig. 6** Illustration of the performance of Elman NN. **a** Red points refer to the original waveform of the gamma signal, blue points refer to the corresponding saturated signal after application of the cutoff, and the green points refer to the waveform recovered using a four-

layer Elman NN. **b** Distribution of the results of the CCS test between the original and recovered waveforms for the Elman NN. (Color figure online)



**Fig. 7** Illustration of the performance of GRBF-NN. **a** Red points refer to the original waveform of the gamma signal, blue points refer to the corresponding saturated signal after application of the cutoff, and the green points refer to the waveform recovered by GRBF-NN. **b** Distribution of the results of the CCS test between the original and recovered waveforms for GRBF-NN. (Color figure online)

complex network structure and a large dataset. For example, Google's Inception is a 22-layer network trained using the Image Net Large-Scale Visual Recognition Challenge 2014, which contains approximately 1.2 million images [29]. However, in this study, the main problem is the accurate prediction of a nonlinear function based on relatively few samples. CNNs are not well suited to this task, as they require large-scale data, have a complex network structure, and are difficult to debug. Therefore, a CNN was not considered in the present work.

## 6 Conclusion

In this study, three ANNs were applied to the restoration of saturated signal waveforms recorded by liquid scintillation detectors. Based on the results presented in the previous section, it is evident that GRBF-NN produced the best CCS test results, followed by Elman NN and then BP-NN. GRBF-NN provides efficient, reliable, and accurate recovery of saturated signals. Of the three ANNs, BP-NN could not successfully restore some saturated signal waveforms, whereas Elman NN required substantial debugging and was relatively time-consuming. GRBF-NN



was determined to produce excellent performance. With the rise in artificial intelligence, neural networks and their applications have developed rapidly. There are many kinds of structures and methods to choose from, and other network structures may also achieve good results in the recovery of saturated signals. For instance, neuro-fuzzy networks have powerful predictive [30] and function approximation [31] abilities and may be suitable for recovery of saturated signals. This will be examined in a future work.

In summary, by choosing the appropriate ANN structure, incomplete signals can be successfully reconstructed. This restoration of saturated signal waveforms recorded by liquid scintillation detectors allows information to be extracted from high-energy signal waveforms, thereby providing more helpful information for further analysis. In future work, the proposed method will be used to recover and identify the waveforms of very few saturated signals obtained from the long-term data of CJPL and investigate the sources of these saturated high-energy signals. This study offers a new perspective on the combination of ANNs with particle physics experiments. The widespread application of this method in high-energy particle and nuclear physics experiments is worthy of further investigation.

## References

1. Q. Yue, W.P. Lai, W.C. Chang et al., Effective dynamic range in measurements with flash analog-to-digital convertor. *Nucl. Instrum. Methods Phys. Res. Sect. A* **511**, 408–416 (2003). [https://doi.org/10.1016/s0168-9002\(03\)02020-5](https://doi.org/10.1016/s0168-9002(03)02020-5)
2. X.B. Xie, Recover certain low-frequency information for full waveform inversion. *SEG Tech. Prog. Expand. Abstr.* **2013**, 1053–1057 (2019). <https://doi.org/10.1190/segam2013-0451.1>
3. D.W. Huang, J.H. Yang, D.J. Zhou et al., Recovering an unknown signal completely submerged in strong noise by a new stochastic resonance method. *Commun. Nonlinear Sci.* **66**, 156–166 (2019). <https://doi.org/10.1016/j.cnsns.2018.06.011>
4. S. Marrone, D. Cano-Ott, N. Colonna et al., Pulse shape analysis of liquid scintillators for neutron studies. *Nucl. Instrum. Methods Phys. Res. Sect. A* **490**, 299–307 (2002). [https://doi.org/10.1016/s0168-9002\(02\)01063-x](https://doi.org/10.1016/s0168-9002(02)01063-x)
5. M. Cavallaro, S. Tropea, C. Agodi et al., Pulse-shape discrimination in NE213 liquid scintillator detectors. *Nucl. Instrum. Methods Phys. Res. Sect. A* **700**, 65–69 (2013). <https://doi.org/10.1016/j.nima.2012.10.056>
6. X.Z. Liu, S.B. Liu, Q. An, A time-over-threshold technique for PMT signals processing. *Nucl. Sci. Tech.* **18**, 164–171 (2007). [https://doi.org/10.1016/S1001-8042\(07\)60040-2](https://doi.org/10.1016/S1001-8042(07)60040-2)
7. C.B. Pushpalatha, Multivariate analysis methods in particle physics. *Annu. Rev. Nucl. Part. Sci.* **61**, 281–309 (2011). <https://doi.org/10.1146/annurev.nucl.012809.104427>
8. M.G. Bonelli, M. Ferrini, A. Manni, Artificial neural networks to evaluate organic and inorganic contamination in agricultural soils. *Chemosphere* **186**, 124–131 (2017). <https://doi.org/10.1016/j.chemosphere.2017.07.116>
9. G. Liu, M.D. Aspinall, X. Ma et al., An investigation of the digital discrimination of neutrons and gamma rays with organic scintillation detectors using an artificial neural network. *Nucl. Instrum. Methods A* **607**(3), 620–628 (2009). <https://doi.org/10.1016/j.nima.2009.06.027>
10. J.P. He, X.B. Tang, P. Gong et al., Spectrometry analysis based on approximation coefficients and deep belief networks. *Nucl. Sci. Tech.* **29**, 69 (2018). <https://doi.org/10.1007/s41365-018-0402-4>
11. Q.J. Zhu, L.C. Tian, X.H. Yang et al., Advantages of artificial neural network in neutron spectra unfolding. *Chin. Phys. Lett.* **31**(7), 256–307 (2014). <https://doi.org/10.1088/0256-307X/31/7/072901>
12. S.A. Hosseini, Neutron spectrum unfolding using artificial neural network and modified least square method. *Radiat. Phys. Chem.* **126**, 75–84 (2016). <https://doi.org/10.1016/j.radphyschem.2016.05.010>
13. H. Qiao, C.Y. Lu, X. Chen et al., Signal-background discrimination with convolutional neural networks in the Panda X-III experiment using MC simulation. *Sci. China Phys. Mech.* **61**, 101007 (2018). <https://doi.org/10.1007/s11433-018-9233-5>
14. A. Yadollahi, E. Nazemi, A. Zolfaghari et al., Optimization of thermal neutron shield concrete mixture using artificial neural network. *Nucl. Eng. Des.* **305**, 146–155 (2016). <https://doi.org/10.1016/j.nucengdes.2016.05.012>
15. H. Gabbard, M. Williams, F. Hayes et al., Matching matched filtering with deep networks for gravitational-wave astronomy. *Phys. Rev. Lett.* **120**, 141103 (2018). <https://doi.org/10.1103/PhysRevLett.120.141103>
16. D.L. Deng, Machine learning detection of bell nonlocality in quantum many-body systems. *Phys. Rev. Lett.* **120**, 240402 (2018). <https://doi.org/10.1103/PhysRevLett.120.240402>
17. X.G. Tuo, B. Cheng, K.L. Mu et al., Neural network-based matrix effect correction in EDXRF analysis. *Nucl. Sci. Tech.* **19**(5), 278–281 (2008). [https://doi.org/10.1016/S1001-8042\(09\)60004-X](https://doi.org/10.1016/S1001-8042(09)60004-X)
18. L. Zou, L.C. Wang, L. Xia et al., Prediction and comparison of solar radiation using improved empirical models and adaptive neuro-fuzzy inference systems. *Renew. Energy* **106**, 343–353 (2017). <https://doi.org/10.1016/j.renene.2017.01.042>
19. Eljen Technology, Sweetwater TX79556 USA. <http://www.ggg-tech.co.jp/maker/eljen/ej-331.html>
20. S.T. Lin, Q. Yue, Status and prospects of CJPL and the CDEX experiment. *Phys. Procedia* **61**, 201–204 (2015). <https://doi.org/10.1016/j.phpro.2014.12.032>
21. H.Y. Xing, X.Z. Yu, J.J. Zhu et al., Simulation study of the neutron–gamma discrimination capability of a liquid scintillator neutron detector. *Nucl. Instrum. Methods Phys. Res. Sect. A* **768**, 1–8 (2014). <https://doi.org/10.1016/j.nima.2014.08.049>
22. M.G. Bonelli, M. Ferrini, A. Manni, Artificial neural networks to evaluate organic and inorganic contamination in agricultural soils. *Chemosphere* **186**, 124–131 (2017). <https://doi.org/10.1016/j.chemosphere.2017.07.116>
23. P.S. Sastry, G. Santharam, K.P. Unnikrishnan, Memory neuron networks for identification and control of dynamic systems. *IEEE Trans. Neural Netw.* **5**, 306–319 (1994). <https://doi.org/10.1109/72.279193>
24. H. Yu, T. Xie, S. Paszczynski et al., Advantages of radial basis function networks for dynamic system design. *IEEE Trans. Ind. Electron.* **58**, 5438–5450 (2011). <https://doi.org/10.1109/TIE.2011.2164773>
25. Y.M. Wu, Y.Q. Wang, L. Li, Application study on BP network and generalized RBF network in estimating distribution model of mechanical products. *Chin. J. Mech. Eng.* **20**, 2140–2144 (2006). [https://doi.org/10.1007/11816157\\_5](https://doi.org/10.1007/11816157_5) (in Chinese)

26. B.M. Vaziri, A. Shahsavand, Analysis of supersonic separators geometry using generalized radial basis function (GRBF) artificial neural networks. *J. Nat. Gas Sci. Eng.* **13**, 30–41 (2013). <https://doi.org/10.1016/j.jngse.2013.03.004>
27. Matlab, The Math Works, Inc., <http://www.mathworks.com>
28. C.B. Pushpalatha, Multivariate analysis methods in particle physics. *Ann. Rev. Nucl. Part. Sci.* **61**, 281–309 (2011). <https://doi.org/10.1146/annurev.nucl.012809.104427>
29. Y. Zhang, M.J. Er, R. Zhao et al., Multiview convolutional neural networks for multi document extractive summarization. *IEEE Trans. Cybern.* **47**(10), 3230–3242 (2017). <https://doi.org/10.1109/TCYB.2016.2628402>
30. F.G. Zhao, J. Chen, L. Gou et al., Neuro-fuzzy based condition prediction of bearing health. *J. Vib. Control* **15**(7), 1079–1091 (2009). <https://doi.org/10.1177/1077546309102665>
31. Y. Wu, B.B. Zhang, J.B. Lu et al., Fuzzy logic and neuro-fuzzy systems: a systematic introduction. *Int. J. Artif. Intell. Expert Syst.* **2**(2), 47–80 (2011)