APPLICATION OF NEURAL NETWORK FOR THE ANALYSIS OF TWO-NEUTRON CORRELATIONS AT SMALL RELATIVE MOMENTA*

P. Kindziuk, J. Pluta, T. Pawlak, M. Przewłocki and K. Wosińska

Faculty of Physics, Warsaw University of Technology
Koszykowa 75, 00-662 Warsaw, Poland

(Received December 28, 2001)

Many of observed phenomena associated to physics experiments do not have a model that allow a good description. If some effects depend on known parameters in the way which cannot be well described, neural network can be a useful tool to solve occurring problems. In this paper neural network is applied to eliminate the false coincidences (cross-talk) in two-neutron correlation function analysis.

PACS numbers: 25.75.Gz, 87.80.Xa

1. Introduction

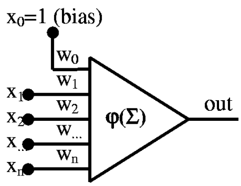
Study of two-neutron correlation function at small relative momenta can provide information about mechanism of heavy ion interactions. A difficult methodical problem in this analysis is a cross-talk effect. It occurs if the same neutron is registered in two or more detectors and leads to deformation of the correlation function. Known methods of the cross-talk elimination [1,2] using the kinematical relations of neutron pair cannot eliminate the multiple cross-talks and reject a lot of real coincidences. In this paper a new method of cross-talk elimination with neural network is presented. It is applied to the data from the E286 experiment performed in GANIL in which Ar–Ni collisions at 77 GeV/n were investigated.

* Presented at the XXVII Mazurian Lakes School of Physics, Krzyże, Poland, September 2–9, 2001.

2. Feed-forward neural network

2.1. Scheme of a single neuron

Fig. 1 presents a scheme of a single neuron. A neuron has an arbitrary number of inputs and a single output. Through every input the neuron receives signals (input vector), which are multiplied by appropriate values (weights). The sum of these values is transformed by a special function called activation function (φ) which gives the output of the neuron. The x_0 signal (polarization) is always 1. Such neural network architecture is commonly used [3].



$\vec{\mathbf{X}} = [x_0, \ldots, x_n]$ input vector

$\vec{\mathbf{W}} = [w_0, \ldots, w_n]$ vector of weights

$$\varphi(x) = \frac{1}{1+e^{-\beta x}}$$

$out = \varphi(\vec{\mathbf{X}} \cdot \vec{\mathbf{W}}^{\mathrm{T}}) = \varphi(\sum_{i=0}^{n} x_i w_i)$ output value

Fig. 1. Scheme of a single neuron.

2.2. Architecture of the network

The multilayer neural network contains a number of neurons forming layers. In the feed-forward neural network there are every-to-every connections between neurons from neighboring layers and there is no information flow between cells from the same layer. The architecture used here contains three inputs, two hidden layers and an output layer (Fig. 2).

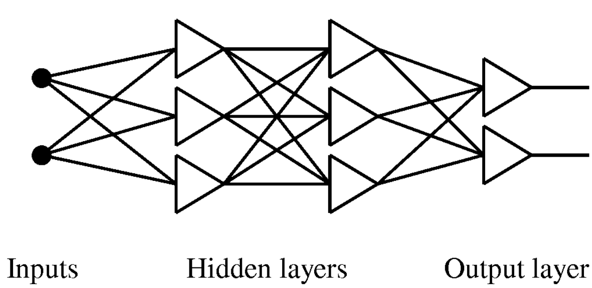


Fig. 2. Scheme of a three-layer feed-forward neural network.

2.3. Learning

The weights of connections between neurons contain information about phenomena being analyzed. Learning is a process of changing weights in order to minimize error of the network. To solve problem indicated in this paper the *back-propagation* method was used. In learning step j the output value of neuron m is

$$y_m{}^{(j)} = \varphi\left(\sum_{i\in \boldsymbol{M}_i} w_i{}^{(m)(j)} y_i{}^{(j)}\right), \tag{1}$$

where M_i is a group of neurons that gives the input signal to neuron m. The weights of the network change according to the formula are

$$\Delta w_i{}^{(m)(j)} = \eta \delta_m{}^{(j)} \frac{d\varphi(x)}{dx_m{}^{(j)}} y_i^{(j)}, \tag{2}$$

where η is the learning rate parameter, Δw change of weight, y neuron's output value, m indexes neurons which receive signal, i is the index of a neuron that signal is coming from and j is the learning step. The difference between expected value z and the output of the network y is

$$\delta_m{}^{(j)} = z_m{}^{(j)} - y_m{}^{(j)}. \tag{3}$$

The error can by calculated using this formula only for the output layer. For hidden layers error is propagated backward according to

$$\delta_m{}^{(j)} = \sum_{k\in M_0} w_m{}^{(k)(j)} \delta_k{}^{(j)}, \tag{4}$$

where M_0 is a group of neurons which receives the signal from neuron m. Changes of values are then described by the equation

$$\Delta w_i^{(m)(j)} = \eta \delta_m^{(j)} \left(1 - y_m^{(j)}\right) y_i{}^{(j)} y_m{}^{(j)}. \tag{5}$$

3. Cross-talk elimination

3.1. The cross-talk problem

The interferometry type analysis of nucleus–nucleus collisions requires two-particle information from the detector. The question is whether the detector response connects the two particles originating from interaction point or it is a false response of one particle going through two or more different detector modules. That can be known for sure only in simulated data. In case of neutron detector the parasite effect of cross-talk is really significant and can considerably distort the results. The usual way of cross-talk elimination takes into account the relations between times of flight of neutrons and the detector geometry. The measured energy deposition can also be used in criterion definition. The E286 experiment has a detector geometry specially designed to allow the cross-talk elimination [4]. But the effective cross-talk elimination procedure is very expensive as far as the statistics is concerned. So the cross-talk elimination is still an open question, which may be solved by using neural networks.

3.2. Description of the method

The network has 3 inputs, 10 neurons in two hidden layers and a single output. It was taught using data simulated according to three moving source model [5].

In order to simulate the detectors answer the MENATE code [6] was applied. The learning set had 1476556 events. Each of the events contained three values for inputs of the network:

1. angular distance between detectors that detected neutrons;
2. difference of linear distances between these detectors and emitting source;
3. time of flight difference between neutrons and corresponding desired value of the output neutron, which was 0.9 if neutron pair was a cross-talk and 0.1 in case of true coincidence. These values are determined by the shape of the activation function which gives value from open range (0,1).

To construct net and perform training the special program was written by the authors.

The training of the net was performed iteratively. In each of 100 iterations whole set of data was proceeded. In the progress of iterations the learning rate parameter η was systematically varied to span the range 0.2–0.002.

To check the generalization ability of the already trained net it was tested using another set of data which contained 1528328 events simulated by the same event generator. The next step was testing learnt net using simulated data.

3.3. Results

During the presentation of events to the already trained net, the output neuron accepts the values from the range 0.1–0.9. It was checked that this choice does not influence the efficiency of the algorithm. In the ideal case this quantity would be exactly 0.1 or 0.9 for the case of true coincidences or cross-talks, respectively. In the real case the corresponding spectrum of outputs is continuous and varies during the learning procedure so, that true coincidences occupy the region of low output values and cross-talks the higher ones. In the real application, the net response to the particular event is classified as acceptable (*i.e.* as true coincidence), if the output value is lower, or as rejected from the statistics (*i.e.* as cross-talk), if it is higher than some preselected threshold value. Thus, one can obtain relative enrichment of the accepted data sample into the true coincidence events.

Two parameters can be defined: the "efficiency" *i.e.* the ratio of the identified false coincidences to the total number of false coincidences existing in the data sample and "eliminated statistics" *i.e.* the relative number of coincidences which had the neural signal higher than the threshold. Both quantities are functions of the threshold value and they are plotted in Fig. 3. The optimum setting of the threshold should be chosen so, that the efficiency is possibly high and, simultaneously, the eliminated statistics is low.

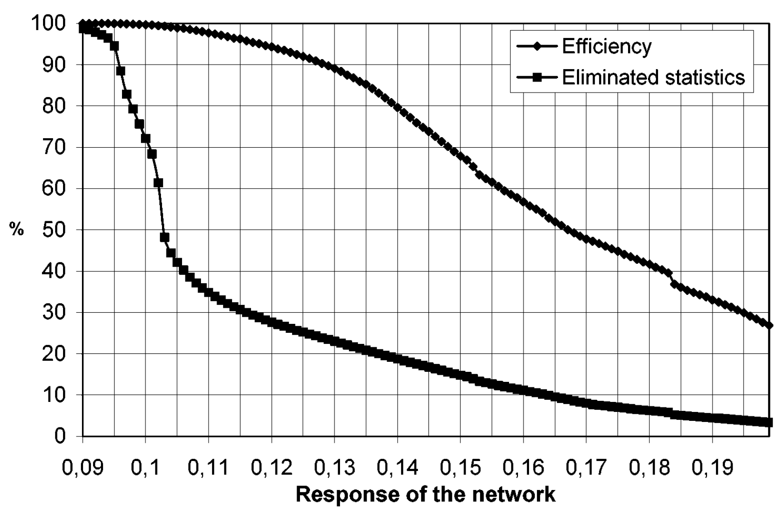


Fig. 3. Efficiency of cross-talk elimination.

3.4. Multiple cross-talks

The neutron can be scattered in two detectors and give three signals (probability of 4 or more is negligible). The efficiency of elimination for one and two false detections per event is shown in Fig. 4. The multiple cross-talks are eliminated with higher efficiency than single cross-talks. Such result is impossible in case of the kinematical methods of cross-talk elimination.

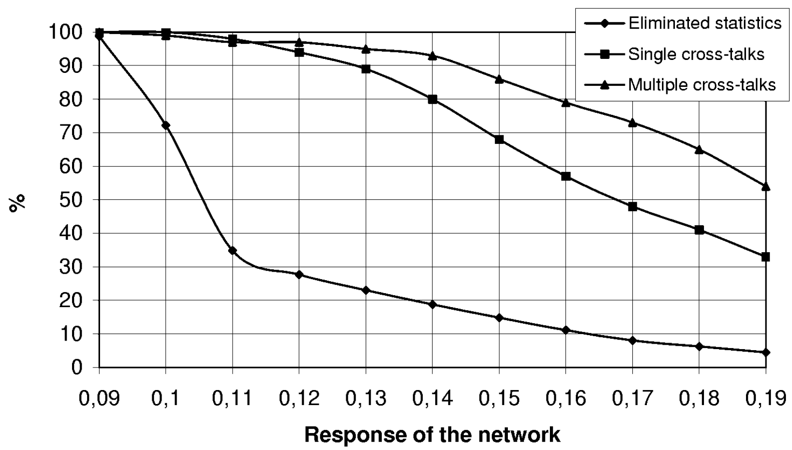


Fig. 4. Efficiency of elimination for single and multiple cross-talks.

3.5. Correlation function

Fig. 5 presents comparison of eliminated statistics for simulated and experimental data. Difference between these values is probably caused by the fact that simulation program does not take into account all aspects of

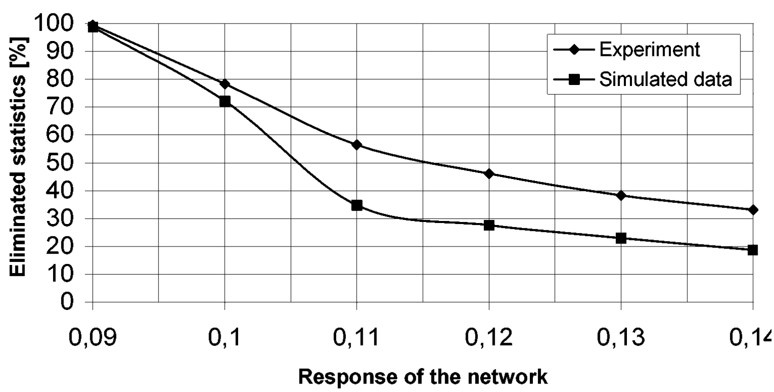


Fig. 5. Eliminated statistics for simulated and experimental data.

the experiment. The optimum threshold was chosen by observing the experimental correlation function for different thresholds. Analysis started at value 0.09 where practically all cross-talks were rejected. The price of this efficiency is an elimination of about 98% real coincidences. Increasing acceptable output value (with step of 0.005) does not change correlation function up to 0.110. In this way the optimum answer of network was determined as 0.105 what corresponds to 99% cross-talk rejection (according to simulated data) and 67% eliminated statistics. Fig. 6 presents the experimental correlation functions before and after cross-talk elimination for three different thresholds and the method based on kinematical dependencies [1].

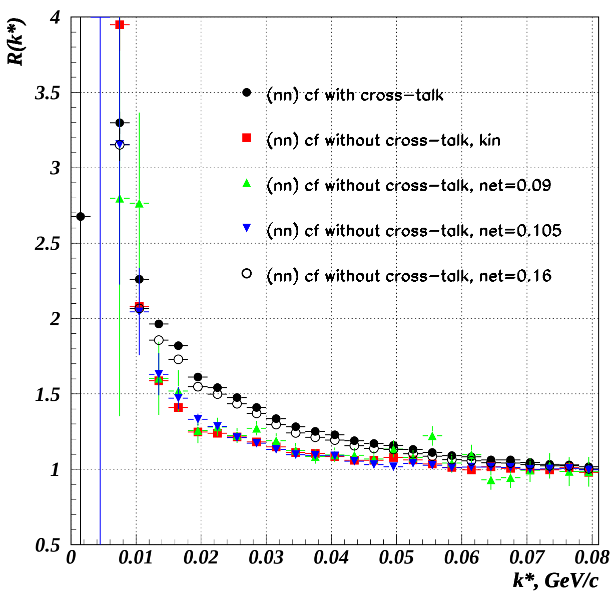


Fig. 6. Experimental correlation functions before and after cross-talk elimination with the kinematical method [17] and the network method using different threshold values.

4. Conclusions

The neural network method can be a useful tool for eliminating parasite effects in correlation analysis of nucleus–nucleus collisions. In particular the method is effective eliminating cross-talks in the study of two-neutron small angle correlations. Contrary to other approaches it takes into account multiple cross-talks too. In an easy way the neural network can incorporate other input parameters like energy loss information.

The authors are grateful to Prof. Robert Kosiński for fruitful discussion.

REFERENCES

[1] J. Pluta *et al.*, *Nucl. Instrum. Methods Phys. Res.* **A411**, 417 (1998).
[2] R. Ghetti *et al.*, *Nucl. Instrum. Methods Phys. Res.* **A421**, 542 (1999).
[3] P. De Wilde, *Neural Network Models* (second edition), London 1997.
[4] M. Przewłocki *et al.*, *Acta Phys. Pol.* **B411**, 379 (2000).
[5] J. Pluta *et al.*, *Eur. Phys. J.* **A9**, 63 (2000).
[6] P. Désesquelles *et al.*, *Nucl. Instrum. Methods Phys. Res.* **A307**, 366 (1991).