International Journal of Pure and Applied Physics
ISSN 0973-1776 Volume 5, Number 1 (2009), pp. 1–14

http://www.ripublication.com/ijpap.htm

Artificial Neural Network Modeling In Heavy Ion Collisions

E. El-Dahshan*, A. Radi*, M. Y. El-Bakry, and M. El Mashad****

** Department of Physics, Faculty of Sciences,*
Ain Shams University, Abbassia, Cairo, Egypt
E-mail: e_eldahshan@yahoo.com {E. El Dahshan}
amr1165@Yahoo.com {A.Radi}
******* *Department of physics, Faculty of Education,*
Ain shams University, Egypt

Abstract

The neural network (NN) model and parton two fireball model (PTFM) have been used to study the pseudo-rapidity distribution of the shower particles for C^{12}, O^{16}, Si^{28} and S^{32} on nuclear emulsion. The trained NN shows a better fitting with experimental data than the PTFM calculations. The NN is then used to predict the distributions that are not present in the training set and matched them effectively. The NN simulation results prove a strong presence modeling in heavy ion collisions.

Introduction

Deep inelastic scattering of an electron on a proton is a very interesting process to probe with high precision the structure of proton. The parton model of Feynman [1, 2], in which the structure of a nucleon (or a hadron) is characterized by a set of parton distributions, was proposed originally in late 1960's to treat high energy deep inelastic scattering, and later many others high energy physics experiments involving hadrons.

Developments concerning the partonic structure of the proton are described [3-9]. Also described a recent parton distribution analysis which incorporates the new precision deep inelastic scattering and related data. Our later works have been used partonic structure for proton and pion as Feynman structure [10-13].

Low energy nuclear experiments showed that the volume of the nucleus is proportional to the number of constituents (nucleons A). Its shape is spherical, except for a few nuclei which are not spherical. We use parton two fireball model which assumed that the number of partons from the projectile hadron will interact with equal number of partons from one or more nucleon in the target nucleus [13]. Also, the

nucleon in the nucleus consists of point like particles called partons.

The quark and gluon distributions in nuclei [14, 15] are investigated by a parton model, where the common partons of several nucleons and the non – nucleonic components are considered.

Quantum Chromo Dynamic (QCD) is able to predict [16] very tiny features of multiplicity distributions at very high energies which demonstrate that the negative binomial distribution is inappropriate for precise description of experimental data. However, it is remarkable that perturbative QCD becomes a powerful tool in describing the soft processes when properly treated with higher order terms taken into account.

The most direct road of attack starts from QCD, now firmly established as a theory strong interactions. Unfortunately, since the problem of confinement is unsolved [16], QCD can only be used as a guideline to build phenomenological models for soft hadronic phenomena. While successful for positron-electron annihilation, such models remain at present unsatisfactory for most other processes, and in particular for hadron-hadron collisions. The models deficiencies are often invoked in support for claims of "new physics", but also this matter is far from being settled.

The present work tries to study of the pseudo rapidity distribution for nucleus-nucleus (A-A) collision in the framework of the parton two-fireball model depending on an impact parameter analysis. The basic assumption of this model [10-13] is that hadrons are composite objects which may be treated as loosely bound states of the spatially separated constituents (quarks) which in turn are composed of point like particles called partons. This nucleon structure have been used in different models [17-18] along with other assumptions to describe hadron–hadron, hadron-nucleus and nucleus-nucleus interactions.

On the other hand, we try to use the artificial neural network (ANN) [19-21] technique to model the A-A collision features at high energies and proposes to calculate the pseudo-rapidity distribution of pions by using the ANN model. The ANN model is chosen for his ability to perform complex functions in various fields of application including pattern recognition, modeling, identification, classification, speech, vision and control systems [22-26].

This paper is organized on three sections. Section two deals with the theoretical (PTFM) study of the pseudo rapidity distribution for nucleus-nucleus (A-A) collision. Section 3, introduce the development in the artificial neural network (ANN) and describe the manner of modeling and training the A-A collision. Following section provides the results and conclusion. Results of the shower particles pseudo-rapidity distribution for C^{12}, O^{16}, Si^{28} and S^{32} on nuclear emulsion has been calculated using PTFM and ANN and compared with the corresponding experimental data [27, 28].

Theoretical Modeling

A Parton of Two-Fireball Model (PTFM) [10-13]**.**

i. Our theoretical treatment will be based on the following basic assumptions:

ii. Hadrons are composed of point – like particles called partons. These partons behave as free-point like particles in a high energy collision. It is assumed to be homogeneously distributed in the hadron volume. In high energy, the nucleus can be considered also to be made up of a collection of partons. During the collision, the majority of parton's momentum is carried by the longitudinal component.

iii. In A-A collisions, only those partons in the overlapping volume of the two interacting nuclei have a probability to interact which are assumed to be stopped in their center of mass (CMS). Their CMS-kinetic energy will be consumed in the excitation of the produced two fireballs.

iv. Each fireball will decay into a number of new created particles (mainly pions) with an isotropic angular distribution in its own rest frame.

v. It is clear now that, at any impact parameter, the excitation energy and then the number of new created particles will be defined by the impact parameter and the corresponding overlapping volume. Then using the above assumptions we can investigate the rapidity distribution in A-A collision.

The Overlapping Function

Let us assume that the projectile and the target nuclei at rest are spheres of radii (R1, R2) respectively; then the statistical probability of impact parameter (b) within an interval (db) is given by:

$$p(b)db = \frac{2b}{(R_1 + R_2)^2}db$$

i.e.

$$p(b)db = \frac{2b}{r_0^{\,2}(A_1^{1/3} + A_2^{1/3})^2}db \qquad (1)$$

Where r_0 = (1.22-1.5) fm and A1 and A2 are the mass numbers of the projectile and target.

nuclei. In terms of a dimensionless impact parameter (x) defined as $x = \frac{b}{r_0}$ eq. (1) tends to:

$$p(x)dx = \frac{2x}{(A_1^{1/3} + A_2^{1/3})^2}dx \qquad (2)$$

If one assumes that the partons from the incident nucleus in the overlapping volume will interact with the nuclear matter of the target, then, we can calculate the overlapping volume V(x) in the incident nucleus rest frame. Then we can calculate the fraction of partons from the projectile nucleus participating in the interaction (Z) as a function of (x). As:

$$Z(x) = \frac{V(x)}{V_\circ} = (\frac{A_1}{2} + \frac{3}{4}A_1^{2/3}A_2^{1/3} - \frac{A_2}{4}) + (\frac{3}{4}A_2^{2/3} - \frac{3}{4}A_1^{2/3})x - \frac{3}{4}A_2^{1/3}x^2 + \frac{x^3}{4} \qquad (3)$$

Where, V^0 is the volume of nucleon.

From Eq. (2) and Eq. (3) we can get the Z-function distribution as:

$$p(z)dz = \frac{2x}{(A_1^{1/3} + A_2^{1/3})^2 \left\{ (\frac{3}{4} A_2^{2/3} - \frac{3}{4} A_1^{2/3}) - \frac{3}{2} A_2^{1/3} x + \frac{3}{4} x^2 \right\}} dz \tag{4}$$

^{12} ^{16} ^{28} ^{32}

we have calculated Eq. (4) for C , O , Si and S on nuclear emulsion components.

Taking A=70 for emulsion as in ref [29]. Clearly, eq. (4) shows that, p (z) is a complicated nonlinear. In the present work, approximate distributions have been employed[10-13] and are given by:

$$p(z)dz = \sum_{k=-1}^{3} C_k Z^k dz \tag{5}$$

Rapidity Distribution of Pions in the Main Center of Mass System (CMS):

In the CMS, the rapidity of fireball with a total energy E_o and momentum P_f (with a component $P_{\| f}$ along the incident direction) can be written as:

$$y_f^* = \frac{1}{2} \ln \left(\frac{E_o + P_{\| f}}{E_o - P_{\| f}} \right) = Sinh^{-1} \frac{P_{\| f}}{\mu_f} \tag{6}$$

Where μ_f is the fireball transverse mass, which is given by:

$$\mu_f = (M_f{}^2 + p_{tf}{}^2)^{\frac{1}{2}}$$

In high energy we can assume the produced fireballs move in the incident direction, then P_{tf} can be neglected and $P_f \cong P_{\| f}$. Therefore, the CMS rapidity of a fireball can be written in the form:

$$y_f^* = Sinh^{-1} \left(\frac{P_f}{\mu_f} \right) = Sinh^{-1} \left(\frac{(A-z)P_0}{Am + T_0 z} \right) \tag{7}$$

Where $P_0 = \frac{1}{2} \left(s - 4m^2 \right)^{\frac{1}{2}}$, $T_0 = \frac{\sqrt{s}}{2}$ are respectively the momentum and K.E of the incident hadron in the CMS of the colliding hadrons and $\sqrt{S}$ is the CMS energy.

A pion emitted in the fireball rest frame a has fixed momentum $\overline{p\pi}$ and total energy ε.

Hence,

$$p_{\|} = \mu \sinh y. \tag{8}$$

$$\varepsilon = \mu \cosh y. \tag{9}$$

Where, $\mu = (m_\pi{}^2 + P_T{}^2)^{\frac{1}{2}}$ is the pion rest mass. Hence,

$$Sinhy = \frac{\overline{p_\parallel}}{\mu} = \frac{\overline{p_\pi} Cos\theta}{\left(m_\pi^2 + \overline{p}_\pi^{\,2} \sin^2\theta\right)^{1/2}} \tag{10}$$

For an isotropic emission in the fireball rest frame, then the probability of emission within a solid angle $d\Omega$ is given by:

$$\frac{d\psi}{d\Omega} = \frac{1}{4\pi} \quad \text{and} \quad \frac{d\psi}{d\theta} = \frac{1}{2}\sin\theta$$

Where,

$$\int_{\theta=0}^{\pi} \frac{d\psi}{d\theta} d\theta = 1$$

Hence,

$$\frac{d\psi}{dy} = \frac{d\psi}{d\theta}\frac{d\theta}{dy} = \frac{1}{2} Sin\theta \frac{d\theta}{dy} \tag{11}$$

$$\text{Now, } \cosh y dy = \frac{1}{\mu}\left(\frac{dp_\parallel}{d\theta}\right)d\theta - \frac{\overline{p}_\parallel}{\mu^2}\left(\frac{d\mu}{d\theta}\right)d\theta$$

This gives,

$$\frac{d\psi}{dy} = \frac{-\varepsilon}{2\overline{p_\pi}\cosh^2 y} \tag{12}$$

The rapidity distribution at a given impact parameter is in the form:

$$\frac{dn}{dy} = n\left|\frac{d\psi}{dy} - \right| = \frac{n\varepsilon}{2\overline{p_\pi}\cosh^2 y} \tag{13}$$

$$\text{with, } -\sinh\frac{\overline{p_\pi}}{m_\pi} \le y \le \sinh^{-1}\frac{\overline{p_\pi}}{m_\pi}$$

To determine the rapidity distribution of the created secondaries in the main CMS, let us consider that the pion has a rapidity y in the fireball rest frame, then the pion rapidity in the main CMS(y*) will be given by

$$y^* = y + y_f^* \tag{14}$$

Where y_f^* is the fireball rapidity in CMS given by Eq. (6).

The rapidity distribution of the created pions (n(z)) from a fireball produced from the interaction at a particular impact – parameter in the main CMS corresponding to the overlapping function z is given by

$$\left(\frac{dn}{dy^*}\right)_z = \frac{n(z)\varepsilon}{4\overline{p_\pi}\cosh^2\left[y^* - y_f^*(z)\right]} \tag{15}$$

Where,

$$y_f^*(z) - b \le y^* \le y_f^*(z) + b \tag{16}$$

And $Sinh^{-1}\dfrac{\overline{p_\pi}}{m_\pi} = b$

Let us assume that the threshold value of the overlapping function for the multiparticle production is $z_{Min} = 2\,\varepsilon/Q$ where Q is the CMS free energy. Hence, the limits of y_f^* are given by.

$$y^*(z_{Min}) \ge y_f^*(z) \ge 0 \tag{17}$$

Corresponding to $z_{Min} \le z \le A$

Using Eq. (16) we can see that the pion rapidity in the CMS from one fireball (right moving) will fall into the region.

$$-b \le y^* \le y_f^*(z) + b \tag{18}$$

Therefore, the rapidity distribution of the created pions from right moving fireballs produced from the interactions at all the impact – parameters will be given by:

$$\frac{dn}{dy^*} = \frac{\sigma_t}{4\sigma_i}\int_{Z_L}^{Z_U} \frac{\varepsilon n(z)\,p(z)dz}{\overline{p_\pi}\cosh^2\left[y^* - y_f^*(z)\right]} \tag{19}$$

To determine the upper and lower limits z_U and z_L we notice that the absolute minimum for y^* is $y^*_{\min} = -b$ (i.e. z=A) and the absolute maximum is $y^*_{\max} = y_f^*(z_{\min}) + b$. Now, to define the fireballs which will give contribution at the rapidity y^* in Eq (16) it is clear that y_f^* will be $\ge y^* + b$ provided that it will not exceed the maximum value of the fireball rapidity i.e.

$$y^*_{f\max} \le y_f^*(z_{\min})$$

Therefore, for fixed pion rapidity y^*, the fireballs contributing to the integral must satisfy the conditions:

$$\max\left[\left(y^* - b\right), 0\right] \le y_f^*(z) \le \min\left[\left(y^* + b\right), y_f^*(z_{Min})\right]$$

or

$$\max\left[\sinh\left(y^* - b\right), 0\right] \le \sinh y_f^*(z) \le \min\left[Sinh\left(y^* + b\right), \sinh y_f^*(z_{\min})\right]$$

which can be written as,

$$\alpha \le \sinh y_f^*(z) \le \beta$$

Since, $y_f^*(z)$ and Sinh $y_f^*(z)$ both decreases with increasing (z) then the lower limit z_L corresponds to β while the upper limit z_U corresponds to α.

Accordingly, one can write,

$$\alpha = \text{Sinh}\ y_f^*(z_U) = \frac{(A - z_U)p_o}{m + T_o z_U}$$

and

$$\beta = \text{Sinh } y_f^*(z_L) = \frac{(A - z_L)p_o}{m + T_o z_L}$$

Solving for α and β we obtain

$$z_L = \frac{p_o - \beta m}{p_o + \beta T_o} \quad , \quad z_U = \frac{p_o - \alpha m}{p_o + \alpha m} \tag{20}$$

The angular distribution of the charged created secondaries can be studied also in terms of another variable called to pseudo–rapidity defined as:

$$\eta = -\ln\left(\tan\frac{\theta}{2}\right) \tag{21}$$

Where θ is the angle of emission in the CMS. The pseudo – rapidity is related to the momentum of the particles as follows:

$$\eta = -\frac{1}{2}\ln\frac{p - p_\parallel}{p + p_\parallel} \tag{22}$$

Hence, the equation for pseudo – rapidity (η) is equivalent to that of rapidity (y) with (p) replacing (E). This means that the two variables are equivalent at very high energies. The particle densities are related by:

$$\frac{dn}{d\eta} = \beta_\pi \frac{dn}{dy^*} \tag{23}$$

Where $\beta_\pi = \left(1 - \frac{m_\pi^2}{\mu^2 \cosh^2 y}\right)$

Referring to Eq (19), if we assume that ε varies with n as ε = an+b where a and b are 0.04 and 0.35 respectively and consequently $\overline{p_\pi} = \left(\varepsilon^2 - m_\pi^2\right)^{1/2}$. Substituting $n = \frac{ZQ}{2\varepsilon}$, then from Eq. (19) we can write :

$$\frac{dn}{d\eta} = \frac{Q\sigma t \beta_\pi}{8\sigma_i} \int_{z_l}^{z_u} \frac{zp(z)dz}{\overline{p_\pi}\cosh^2\left[y^* - y_f^*(z)\right]} \tag{24}$$

Artificial Neural Networks (ANNs)

Basic principle

An artificial neural network is made up of many simple and highly interconnected computational elements [19-21], which are called artificial neurons or nodes (Figure 1). These nodes are distributed on many different layers: one input layer, one or many hidden layers and one output layer. The nodes in adjacent layers are fully or partially interconnected with weighted links. The net input into the j^{th} layer node (in[j]) is equal to the sum of weighted outputs from the prior i^{th} layer (out[i]).

$$in[j] = \sum w_{ij} out[i] \tag{25}$$

where, w_{ij} is the weight factor.

The hidden neurons (neurons of the hidden layers) and the weight factors of the links between them play a critical role during the learning process. In the case of supervised training, the numerical values of the weight factors change according to the training data sets, in order to minimize the difference between the actual outputs and the target value. Thus, the relationship between causal factors and response is mapped during the learning process.

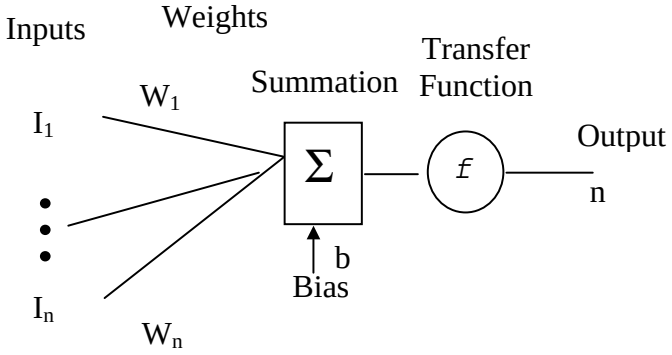


Figure 1: Neuron Model.

The transfer function of processing nodes is used to determine the output value of the node based on the total net input from nodes in prior layer. The most widely used transfer function is a sigmoid function, which is shown in the following equation:

$$out[j] = \frac{1}{1+e^{-in[j]}} \tag{26}$$

where in[j] is the sum of the net inputs from nodes of the prior layer and out[j] is the output from the j^{th} hidden layer node.

Inputs to neurons could be either from external stimuli or from the outputs of neurons from prior layers. Single outputs from neurons could be inputs to many other neurons in the network. When the inputs of the neuron exceeds a certain threshold, specified by a threshold (transfer) function with bias, the neuron is fired and an output signal is produced. When the weights are tuned via some kind of learning process, the network recognizes the input output relation (mapping function)

The number of nodes in the input layer is determined by the number of independent variables to be investigated. The number of nodes in the output layer is determined by the number of dependent variables. The number of hidden layers and the number of hidden nodes in each layer is strongly dependent on the complexity of the problem [30, 31]. The optimal number of hidden nodes depends on many factors such as the number of input and output nodes, the number of training data sets, the amount of noise in the targets and the complexity of the function or classification to be learned. There is no magic formula to use, in order to determine the number of hidden layers, and the number of nodes in each layer. Trail and error approaches remain the most commonly used methods; however, some rules of thumb and empirical formulas may be used as guidance.

Modeling the A-A Pseudo Rapidity Distributions Using ANN

The proposed ANN model of the pseudo-rapidity distribution have three inputs and one output. The inputs are: the lab momentum (P_L), the mass numbers of the projectile nuclei (A) and the pseudo rapidity (η).The output is the pseudo-rapidity (density) distribution ($\frac{1}{N}\frac{dN}{d\eta}$)

Using this input-output arrangement, different network configurations were tried to achieve good mean sum square (MSE) and good performance for the network. The configuration, shown in Figure 2, was chosen. It consists of an input layer, two hidden layers of 4 and 9 neurons, respectively, and an output layer. The transfer functions where chosen to be a log sigmoid function for the first and the second hidden layers and a linear sigmoid function for the output layer.

Training of the A-A-ANN.

Training an ANN consists of making a particular input leads to a specific target output. The weights are adjusted, based on a comparison of the output and the target, until the network output gets as close as possible the target value.

The proposed ANN model was trained using Levenberg-Marquardt optimization

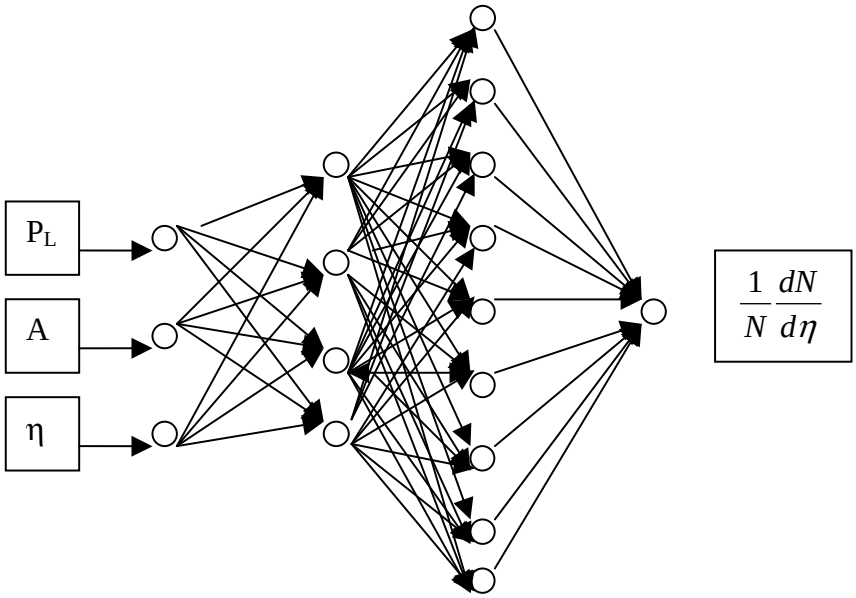


Figure 2: The A-A NN based modeling

Technique [32-34]. This optimization technique is more powerful than the conventional gradient descent techniques. The Levenberg-Marquardt updates the network weights using the following rule:

$$\Delta W = (\mathrm{J}^{\mathrm{T}}\ \mathrm{J} + \mu\ \mathrm{I})^{-1} J^{T} e \tag{27}$$

Where J is the Jacobian matrix of derivatives of each error with respect to each weight, μ is a scalar, changed adaptively by the algorithm and e is an error vector. The

initial training weight were also chosen using the Nguyen-Widrow random generator in order to speed up the training process.

Data collected from experiments are divided into two sets, namely, training set and validation set. The training set is used to train the ANN model by adjusting the link weights of network model, which should include the data covering the entire experimental space. This means that the training data set has to be fairly large to contain all the required information and must include a wide variety of data from different experimental conditions, including different formulation composition and process parameters.

Initially, the training error keeps dropping. If the error stops decreasing, or alternatively starts to rise, the ANN model starts to over-fit the data, and at this point, the training must be stopped. In case over-fitting or over-learning occurs during the training process, it is usually advisable to decrease the number of hidden units and/or hidden layers. In contrast, if the network is not sufficiently powerful to model the underlying function, over-learning is not likely to occur, and the training errors will drop to a satisfactory level. Therefore, the training data can be used to check the architecture and training progress of the ANN model.

The validation data set is used to confirm the accuracy of the ANN model. It ensures that the relationship between inputs and outputs, based on the training and test sets are real, and not artifacts of the training process. The validation data set should include data, which are not included in the training data set, but lie in the data boundaries of the training data set.

The cases for A = 16 and P_L = 60 AGeV/c were used as validation data sets. The ANN training convergence is adversely affected if the input levels are far apart from each other. Thus, the input data for P_L, A and η was scaled down by 10, 100 and 10 respectively in order to keep both ANN input levels close to each other. Training was terminated after average sum square error of 10^{-4} was reached. For work completeness, the function which describes the pseudo-rapidity distribution at lab momenta for different beams in heavy ion collisions is given by:

$$N(P_L, A, \eta) = \text{purline}[\tan\text{sigmoid}(\text{net.LW}\{4,3\}\log\text{sigmoid}(\text{net.LW}\{3,2\}$$
$$\tan\text{sigmoid})\text{net.LW}\{2,1\}(A + \text{net.b}\{1\}) + \text{net.b}\{2\}) + \text{net.b}\{3\}) + \textit{netb.}\{4\})]$$

Where:

A: is the input which consists of two inputs (wavelength, doped percentage D values)

Purline (x) = x and tan sigmoid(x) = 2/(1+exp(-2*x))-1

net.LW{2,1}: linked weights between the input layer and first hidden layer.

net.LW{3,2}: linked weights between 1st hidden layer and 2nd hidden layer.

net.LW{4,3}: linked weights between 2nd hidden layer and output layer.

net.b {1}: the biases for input layer.

net.b {2}: the biased for first hidden layer.

net.b {3}: the biases for second hidden layer.

net.b {4}: the biases for output layer.

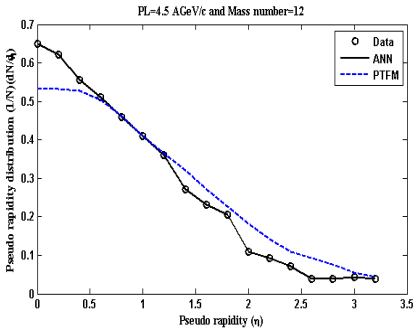
The weights and biases in the equation are collected in appendix A.

Results and Discussion

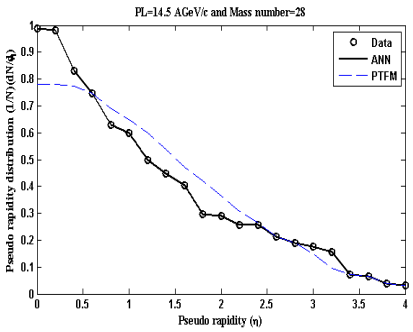
By integrating Eq. (24) numerically and summing the contributions of the target fireball and the projectile fireball, we can get the pseudo-rapidity distribution for each of the projectiles on the emulsion components separately. We can, also, combine these distributions to get the final shower particles pseudo-rapidity distributions for each projectile with the target emulsion. The resultant pseudo-rapidity distributions have been normalized to the charged pseudo-rapidity distributions.

Simulation results based on NN model and PTFM for modeling the pseudo-rapidity distribution of C^{12}-Em at 4.5 AGeV/c, Si^{28}-Em at 14.5 AGeV/c, O^{16}-Em at 200 AGeV/c and S^{32}-Em at 200 AGeV/c are given in Figure 3-a, b, c, and d respectively. We notice that the PTFM is in fair agreement with experimental data. The curves reproduced by the trained ANN model show an exact fitting to the experimental data in the four cases. In the case of O^{16}-Em at 60 AGeV/c described by figure 3-e, we notice an exact fitting between the predicted curves and the experimental data. Then, the ANN technique is able to exactly model for pseudo-rapidity distribution at lab momenta for different beams in heavy ion collisions.

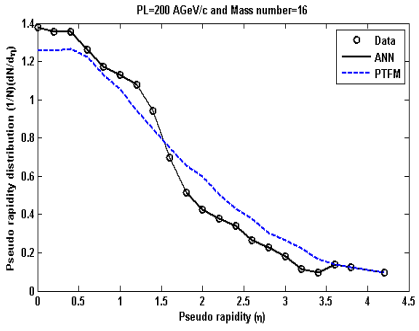
a



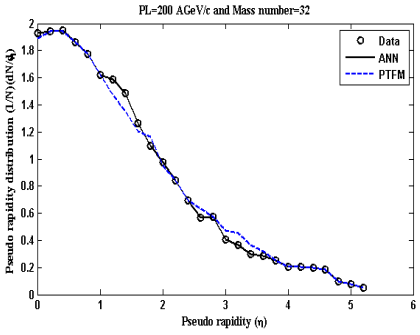
b



c



d



e

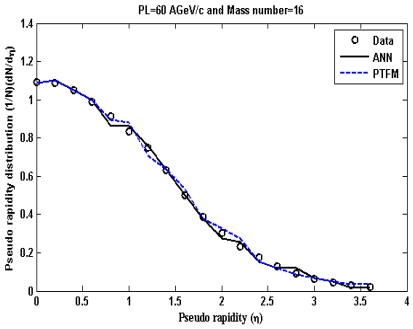


Figure 3: (a,b,c,d &e): The training and the predict result.

References

[1] R.P. Feynman, "Photon–Hadron Interactions", Reading, Massachusetts: Benjamin (1972).

[2] W. Y. Pauchy Hwang, Bulletin of American Society, Vol. 40 (2), pp. 948a (1995).

[3] A. Corsetti et al, Instituto Nazionale Difizica Nucleare, Frascati (Italy), Lab. Nozionale di Frascati (1996).

[4] J. S.Bell et al, World Scientific (Singapore), 950 P, 616–619 (1995).

[5] V. V. Lugovoj, Uzbekistan Fizika Zhurnali, V. 4, P. 10–16 (1996).

[6] Hu–Guoju, Journal of Qingdao University (China), Natural Sciences Edition, V. 10(2), P.55–59; Proceedings of Tenth National Conference of Nuclear Physics, Qingdao, August (1997).

[7] S. Kuhlmam, Nucl. Phys. B. Proc. Suppl. (Netherlands), Vol. 79, pp. 108 (Oct.1999).

[8] W. J. Stirling, Journal of Physics. G: Nucl. Part. Phys. 26, 471–480 (2000).

[9] A. D. Martin et al, J. Phys. G: Nucl. Part. Phys. 26, 663–665 (2000).

[10] K. A. El-Metwally, T. I. Haweel and M. Y. El-Bakry, "A universal neural network representation for Hadron-Hadron interactions at high energy", IJMPC International Journal of Modern Physics C, Vol. 11, No. 3, pp. 619–628 (2000).

[11] M. S. El-Nagdy, S. M. Abdelhalim, M. N. Yasin, E. I. Khalil, M. Y. El-Bakry, M. El-Mashad and M. Tantawy, Arab Journal of Nuclear Science and Applications, 33(2), pp.(115 – 126). (2000).

[12] T. I. Haweel and M. Y. El-Bakry, "Fourier analysis for proton-proton interaction at high energy", International Journal of Theoretical Physics, Vol. 38, No. 10, pp. 2605-2616 (1999).

[13] M. Tantawy, M. El-Mashad and M. Y. El-Bakry, "Multi-particle production process in high energy nucleus-nucleus collisions", Indian Journal of Physics, 72A, 110 (1998).

[14] Xue–Dali;Li–Guanglie, Chinese Physics Letters, Vol. 13(7), pp.508–511 (1996).

[15] S. Kumano, Australian Journal of Physics, Vol. 50(1), P. 45–51 (1997).
[16] E. A. De Wolf, I. M. Dremin and W. Kittel, Phys. Rep. 270, 1-141 (1996).
[17] E. Fermi, Prog. Theor. Phys. 5,570 (1950).
[18] J. Ranft, Phys. Lett. 31B,529 (1970).
[19] P. Bhat, et. al., "Using neural networks to identify jets in hadron-hadron collisions", Proceeding of The Summer Study on HEP, Research Directions-the Decade, Snowmass, Colorado, (1990).
[20] Y. Sun, Y. Pengand, A. Shukla, "Application of Artificial Neural Networks in the Design of Controlled Release Drug Delivery Systems," Advanced Drug Delivery Reviews 55 pp. 1201-1215 (2003).
[21] J. Bourquin, H. Schmidli, P. Van Hoogevest, and H. Leuenberger, "Application of Artificial Neural Networks (ANN) in the Development of Solid Dosage Forms," Pharm. Dev. Technol. 2 pp.111-121 (1997).
[22] R. P. Lippman, "An introduction to computing with neural nets", IEEE Acoustics, Speech, and Signal Processing Magazine, pp. 4-22, April (1987). Simon Haykin, "Neural Networks: A Comprehensive Foundation", IEEE Press (1994).
[23] I .A. Basheer, M. Hajmeer, "Artificial neural networks: fundamentals, computing, design, and application," Journal of Microbial Methods 43, pp. 3–31, (2000).
[24] J .C. Carpenter, M.E. Hoffman, "Understanding neural network approximations and polynomial approximations helps neural network performance," AI Expert 10, pp. 31–33. (1995).
[25] R. Hecht-Nielsen, "Kolmogorov's mapping neural network existence theorem," in the Proceeding of the First IEEE International Joint Conference on Neural Networks, San Diego, 1987, pp. 11–14.
[26] M. El-Nady et al., Int. J. of Mod. Phys. E. Vol. 2, No 2, page 381, (1993).
[27] P. L. O. Jain, T. K. Sngupta and G. Singh, Phys. Rev. C44, 2, 844 (1991) ; Phys. Rev. C43, 5, R2027 (1991).
[28] M. N. Yasin, IL Nuovo Cim. 108A8, 929 (1995).
[29] E. Baum, D. Haussler, "What size net gives valid generalization?," Neural Computation 1, pp. 151–160, (1989).
[30] C .G. Looney, "Advances in feed-forward neural networks: demystifying knowledge acquiring black boxes," IEEE Transaction on Knowledge Data Engineering 8, pp. 211– 226, (1996).
[31] M.T. Hagan and M.B. Menhaj, "Training feed-forward networks with the Marquardt algorithm," IEEE Transactions on Neural Networks, 6, pp. 861-867, (1994).
[32] M. Y. El-Bakry, K. A. El-Metwally, "Neural network model for proton-proton collision at high energy", Chaos, Solitons and Fractals 16, pp. 279-285, (2003).
[33] A. K. Hamid, "Scattering from spherical shell with a circular aperture using neural networks approach", Canadian Journal of Physics, Vol. 76, pp.63-67, (1998).

Appendix A.

The efficient ANN structure is given as follows: [3x4x9x1].
Weights coefficient after training are:

net.LW{2,1} = W_{43} = [-1.0002 1.9592 0.6655
1.1702 0.0764 -1.6571
0.7018 -0.0627 1.2969
2.1502 -2.2863 0.8206].

net.LW{3,2}= W^{94} = [-0.9057 -1.9083 -1.0978 0.1024
-0.4889 -1.5885 -0.6670 1.4674
3.0255 -0.3619 0.6745 0.4433
0.9203 2.1178 1.0522 1.0889
-0.7995 -1.1022 1.8372 -1.0877
0.5496 -0.4107 1.0406 2.3459
2.5448 1.1958 -1.2292 1.0330
1.0147 0.7808 -1.6959 0.1524
0.3763 0.7545 1.9527 1.1657].

net.LW{4,3}=W^{19} = [0.0202 -0.2694 0.0136 -0.0074 0.0465 0.0003 0.3887 0.1506 -0.0081].

net.b {1} = [-4.7175 -2.2157 3.6932].
net.b {2} = [0.9586 -0.9728 1.0535 2.7934].
net.b {3} = [2.1520 2.3558 -1.3002 -0.8495 0.3002 0.5594 1.8024 2.0971 2.1478].
net.b {4} = [-0.2071].