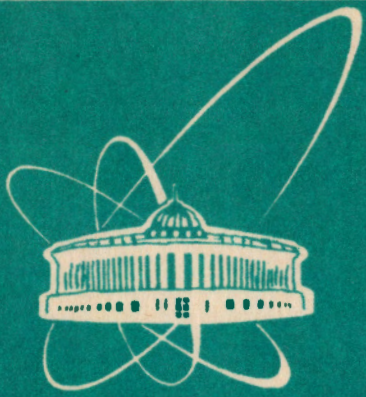

сообщения
объединенного
института
ядерных
исследований
дубна

E10-93-86

S.Baginyan, I.Kisel, E.Konotopskaya, G.Ososkov

TRACK FILTERING BY ROBUST NEURAL NETWORK

1993

1 Introduction

The Artificial Neural Networks (ANN) approach was sufficiently used last years for solving many difficult optimization and recognition problems [1, 2]. The track finding in high energy physics is one of them [3, 4]. It can be considered as the research on the global minimum of some ANN energy function. The computer time of this research depends on, at least, two main factors:

- capability of ANN algorithm to avoid sticking into one of local minima;
- number of ANN degrees of freedom, which for ANN with N signals is equal to N^2 for a conventional ANN algorithm.

Between the methods overcoming the first factor the most known is the simulated annealing schedule [5], that is rather time consuming. Therefore we concentrate ourself on the different approach with looking for such an initial ANN configuration that is situated in a vicinity of the energy function global minimum. A specially constructed cellular automaton was used with this purpose in [6]. We also use so-called rotor model of ANN, since it allows, in particular, to decrease the number of the ANN-degrees of freedom. In this rotor model each neuron n_i is associated with a rotor, i. e. an unit vector $\vec{s}_i$. On the energy function such constraints are imposed that the ANN-evolution leads to orienting of every rotor along its track.

So in our previous work [7] we applied the modify rotor model for the track reconstruction in multiwire proportional chambers (MWPC) of the ARES spectrometer (JINR, Dubna). In our modification we used specific features of MWPC to set up an initial rotor configuration, where each rotor direction and position determine, the tangent and coordinates of a corresponding particle track. We used rotors as non-unite vectors, so their length was considered as an activation level characterizing their belonging to track. Such an initial configuration brings the ANN to a neighborhood of the global minimum of the energy function.

Besides we suggested a quite natural view of the connection strength (weight) w_{ij} between n_i and n_j which depends from the geometrical features of a particle trajectory, but does not depend on the distance between n_i and n_j. It allows to fulfil some fitting procedure obtaining track parameters and, from the another hand, to simplify the expression for energy function, which does not include any special constraint terms. Calculations were speeded up by rejecting the non-significant neurons.

This modification had demonstrated its satisfactory behavior on the statistics of 10^4 real three-track events.

Nevertheless, we realized quite well that our model is too tightly oriented on MWPC applications. It undoubtedly needs for further working out to the region of TeV-energy events with their higher multiplicity and crossings of close going tracks. Besides there are some indications of noise sensitivity of such ANN-models [8].

Therefore, in the present paper we study the following problems of track information extraction by our ANN model:

- providing initial ANN configuration by an algorithm general enough to be applicable for any discrete detector in- or out of a magnetic field;
- robustness to heavy contaminated raw data (up to 100% signal-to-noise ratio);
- stability to the growing event multiplicity.

One could find similar consideration [9], where the authors propose a quite reasonable track finding approach, combining the local Hough transform with the deformable template method.

However, starting from our ideas mentioned above, we introduce here considerable innovations in almost every stage of ANN algorithm:

1. On the stage of initial determination of rotors (i. e. their directions and lengths) we decline the Hough transform due to its computational requirements and substitute it by a special one-dimensional histogramming (see next section). In principle, it was a good possibility to create templates (track candidates) and specify a number of them, but we did not go this way, leaving for the network itself to develop this initial information into the most significant neuron connections.
2. A repulsive force allowing to neglect noise and other non-significant neurons, we multiply the weights by a specially designed robust multiplier [11].
3. We replace the simulated annealing schedule by the ANN dynamics with an optimally fixed temperature.

Our approach is valid for both circular and straight (non-magnetic) tracks and tested on $2D$ simulated data, although there is no problem to generalize it to $3D$ straight or helix tracks. Therefore, after inferring below the general mathematical formalism, we use further data simulated for non-magnetic tracks emanating from different vertices inside target. That gives tracks crossing each other under very small angles. A similar model is also used in [9].

Track data are contaminated by 100% noise points distributed uniformly. To be closer to some reality in our simulation, we keep parameters of the cylindrical spectrometer ARES [10] (see an example on fig. 5).

Although this paper is bearing a preliminary character and the study is supposed to be continued, the results of simulations presented in the last section look very promising.

2 Initialization

We consider ANN rotor model with N $2D$ signals (measured points (x_i, y_i) $i = \overline{1, N}$ of tracks and noise signals) assuming tracks to be whether circles or straight lines (which is a particular case of circle with $R = \infty$). Every rotor $\vec{s}_i$ starts from (x_i, y_i) in direction determined by its angle φ_i with the axis OX and has the length $|\vec{s}_i| = s_i$.

The best initiation of an ANN would be to attach every rotor $\vec{s}_i$ to its signal (x_i, y_i) varying φ_i, s_i in such a way that the ANN energy function

$$E = -\frac{1}{2}\sum_{ij} \vec{s}_i w_{ij} \vec{s}_j, \tag{1}$$

would appear as close as possible to its global minimum.

Here weights w_{ij} are determined following [7] as functions of the angle $\varphi_{L_{ij},\widehat{OX}}$ between OX-axes and segment L_{ij} connecting neurons n_i and n_j.

$$w_{ij} = \begin{pmatrix} \cos(2\varphi_{L_{ij},\widehat{OX}}) & \sin(2\varphi_{L_{ij},\widehat{OX}}) \\ \sin(2\varphi_{L_{ij},\widehat{OX}}) & -\cos(2\varphi_{L_{ij},\widehat{OX}}) \end{pmatrix}. \tag{2}$$

If i-th and other points lie on the same circle, the transform (2) turns tangents of all other points on this circle to the same direction as the tangent to i-th point. It looks especially easy for a straight track, when all angles $\varphi_{L_{ij},\widehat{OX}}$ for points belonged to this track are equal and differ for all other points.

Thus, for the non-magnetic case that gives us the following procedure of a rotor $\vec{s}_i$ initializing. For given point i we histogram all angles $\varphi_{L_{ij},\widehat{OX}}$. Then we look for three neighboring bins with the maximum number of points in them. The initial rotor angle φ_i is determined as the center of gravity of these three major bins. The starting activation level s_i is determined by their mean value ($1/k$ of the total number of points in them, where k is the number of used chambers).

Since for any experimentally measured track the results of this procedure depend on the bin width, it must be chosen accurately. For our ARES-like simulated model a step between wires in MWPC of the setup is about 2 mm, therefore experimental points (hit wires) are not lying exactly on the line. In

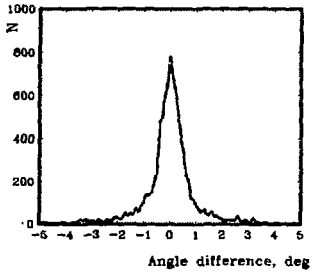

Figure 1: The distribution of the angle between a simulated straight track and lines, connecting all pairs of hit wires.

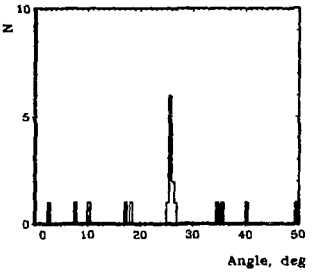

Figure 2: An example of histogramming method application.

fig. 1 the distribution over 100 tracks of the angle between a simulated straight track and segments connecting all pairs of hit wires is shown. Since the square above the interval $(-0.75^o,\ 0.75^o)$ covers over 80% of this distribution, the step of histogramming $\Delta\varphi = 0.5^o$ is enough to divide a peak into three parts. An example of the histogramming method application is presented in fig. 2.

The above initialization procedure could become a main time-waster of any program, if one would vary j over all possible values. To decrease the number of sorting trials, geometrical and physical considerations must be taken in to account. Since in our model each track starts from its own origin in the target area and crosses every chamber only one time, we ignore neurons lying on the same chamber and test others only from an admissible range lying between two straight lines crossing in the given point and embracing the target area.

This trick diminishes the number of each neuron degree of freedom from N^2 to $(p-1)N\psi$, where p is the maximal track point number (number of

used chambers) and ψ is a small coefficient. In our model $\psi \approx r/2\pi R$, where R and r are, correspondingly. radii of the spectrometer and target. $N\psi \sim 1$ for very small r.

An analogous histogramming procedure can be fulfilled in the case of circular tracks, though it is more complicate, since one needs at least three points to draw a circle, so the number of trials is larger. The region of admissible circles is determined similarly by two circles of the physically smallest radius. Besides there can be used some obvious physical restrictions relating to the specific setup of a given experiment.

3 Robustness and temperature considerations

Starting from this initial rotor configuration, we have to develop an algorithm, intended to change rotors iteratively approaching on the every iteration to the global minimum of the energy function (1).

A common simulated annealing procedure [5] allows to fulfil that avoiding local minima by thermalizing our set of neurons, i. e. assuming them randomly disturbed due to placing in a thermostat with a temperature T. According to the mean field theory (MFT), we substitute each rotor by its thermal average $\vec{v}_i = \langle \vec{s}_i \rangle_T$. In fact, this averaging over all other neurons is carried out by the updating equation [3]:

$$\vec{v}_i^{(m+1)} = \frac{\vec{H}_i^{(m)}}{|\vec{H}_i^{(m)}|} \frac{I_1(|\vec{H}_i^{(m)}|/T)}{I_0(|\vec{H}_i^{(m)}|/T)}, \qquad (3)$$

where m — is the iteration number, the local field $\vec{H}_i^{(m)} = \sum_j w_{ij} \vec{v}_j^{(m)}$ is obtained by vector summing up over all significant neurons (significant means with non-zero weights, $w_{ij} \neq 0$). The temperature T determines the slope of the function (3) at the origin, when $T \to 0$ it converges to a step-function.

This updating rule is asynchronous: the iteration is completed, when all neurons get new values one by one.

The simplified view of the energy function (1) without any constraint terms like in [2, 3, 4] could lead, in principle, to a track number increase due to appear fictive tracks.

To prevent this, we use the modified robust approach. Instead of using an implicit robustness of the Potts factor (see, for example [9]) we amplify the repulsive force explicitly by multiplying w_{ij} by a special robust multiplier responsible for the "soft" rejecting of all outlying neurons.

As such a multiplier we use the optimal weight function

$$w_{opt}(t) = \frac{1}{1 + c\exp(t/2)}, \tag{4}$$

which was derived for the robust M-estimate of the average [12]. Here the only parameter c is inversely proportional to the signal-to-noise ratio within the narrow corridor around a track. The argument in (4) is equal to

$$t = \varphi_{\widehat{\vec{v}_i \vec{v}_j}} \tag{5}$$

with $\vec{v}_j' = w_{ij}\vec{v}_j$.

Due to the closeness of $w_{opt}(t)$ to the famous Tukey's bi-weights [11]

$$w_{opt}(t) = \begin{cases} \left(1 - \left(\frac{t}{c_T}\right)^2\right)^2 & ,|t| \le c_T; \\ 0 & ,|t| > c_T \end{cases} \tag{6}$$

and for the sake of computing time economy in our calculations we use (6) instead (4) with $c_T = 2^o$.

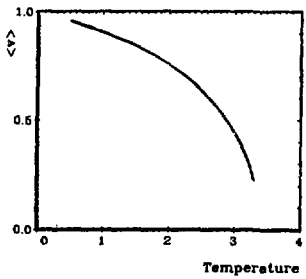

Figure 3: Final average activation level as a function of the temperature for 10 points per track.

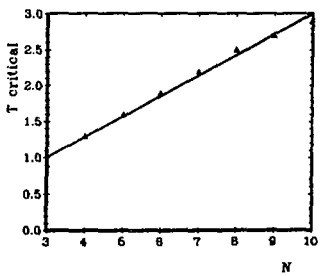

Figure 4: The dependence of cut-off temperature on a number of points per track.

In the simulated annealing schedule the ANN dynamics begins from some starting temperature and lowering with every iteration passes the phase transition point T_{ptp} [5, 9]. As it was mentioned above, our initial rotor values are designed to set ANN in a vicinity of the global minimum, i. e. *below* T_{ptp}.

Therefore, it is reasonable to substitute the simulated annealing recommendation to use a decreasing temperature sequence by applying the only temperature T_{opt} optimally chosen for all iterations. As a criterion of the optimum we use the average value of the rotor activation levels

$$B = \frac{1}{p}\sum_{i=1}^{p}|\vec{v}_i|. \qquad (7)$$

In our calculations we suppose the following criteria of ANN stable state and neuron non-significance:

1. The global minimum is reached, if

$$\max_{i=1}^{N}|\vec{v}_i^{(m+1)} - \vec{v}_i^{(m)}| < 0.05. \qquad (8)$$

2. Neuron is ignored, if

$$|\vec{v}_i| < 0.5. \qquad (9)$$

Thus, from (7) and conditions (9) a track with $B < 0.5$ can be considered as a random cluster of neurons. The dependence of B upon temperature after attaining the stable state is shown in fig. 3. One can see, that holding (9) the temperature must be less than a critical value 2.9 to keep 10-points track. Therefore, we can use this temperature as a cut-off parameter. Fig. 4 shows the dependence of this cut-off temperature on a number of points per track. In our case $T = 1.5$ is suitable to keep tracks having more than 5 experimental points (neurons).

4 Simulation results

This robust rotor model of neural network was tested on simulated events obtained on the basis of parameters of the ARES spectrometer (i. e. 10 cylindrical chambers with a distinctive set up). As was mentioned above, all the events contain only straight tracks and have 100% of noise counts. An example of 30-track event is presented in fig. 5. In this event there are about 300 points corresponding to tracks and 300 noise points (10 noise points per track). The result of the application of the robust rotor model one can see in fig. 6. Here rejected points are denoted by dots, but active neurons are shown as vectors with lengths proportional to their activation levels. Outer lines indicate simulated straight tracks. In this case 251 noise points are rejected by our network and the rest 49 are attached by tracks (on the average 1.6 per track).

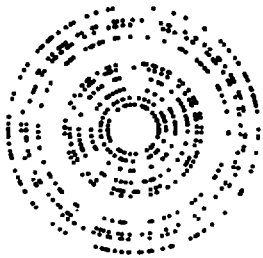

Figure 5: An example of 30-track event with 100% noise.

Figure 6: The result of the application of the robust neural network to data presented on the previous figure.

The results of simulation of 1000 events are presented in fig. 7–9. As it is seen in fig. 7, the average number of iterations grows in dependence of the multiplicity of events. In the same time the dispersion of the iteration number is rather big. For example, its range for events with 25 tracks is vary from 4 to 22. More detailed analysis shows the iteration number growth depends mainly on the random track condensations, than on multiplicity. The latter influences implicitly due to the higher probability of cases track intersecting under the small angles or going close, almost parallel.

A comparison of simulated tracks and track-candidates found by network demonstrates that suggested criteria (8, 9) allow to include to a set of track-candidates all the simulated tracks. However, in this set there are some fictive track-candidates formatted both by noise points and by "spoiled" tracks obtained whether by parts of close going ones or combined from pieces of tracks with addition of noise points.

The corresponding information is presented on fig. 8 given the dependence of the point number of track-candidates upon event multiplicity. The average of this point number decreases from 10 (number of chambers) to 7 for 50 tracks per event.

The next quality characteristic of our algorithm is the rate of the noise

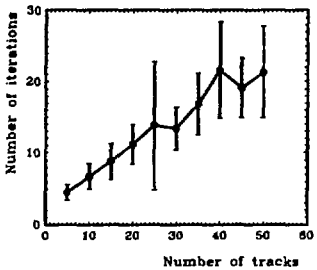

Figure 7: Average number of iterations as a function of event multiplicity.

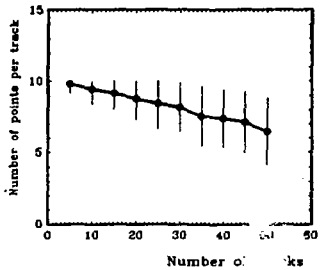

Figure 8: Average point number of track-candidate as a function of event multiplicity.

reduction. It is shown in fig. 9, where the total number of points rejected as noise is given as a function of event multiplicity. It also decreases almost linearly, since in our simulations the number of real track points is doubled by noise points. Latters are getting closer and closet to simulated tracks with growing multiplicity, so the probability increases, inevitably, to include noise points among points of some real track.

5 Conclusion

Concluding we can state the results of testing the suggested innovations to our previous algorithm [7]:

1. Algorithm liberates from its orientation on a concrete detector, becomes more general and with comparable simple changes can be adapted to magnetic field case and/or $3D$ search.

2. Quality of algorithm improved considerably:

   - it becomes less sensible to multiplicity. The dependencies shown in fig. 8 and 9 are to be much weaker, when algorithm will be adapted to $3D$ search.

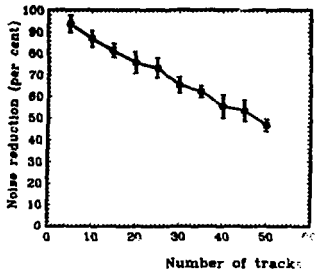

Figure 9: Total point number rejected as noise points.

- it is quite robust in M-estimate sense.

3. Due to both an essential decrease of the number of degree of freedom and the satisfactory initial ANN configuration, our algorithm is fairly fast and suitable for parallel processing.

In our outlook we should stress again that in this paper we restrict ourself on the stage of track filtering leaving for the future problems of track fitting for removing from track-candidates fictive tracks and purifying accepted tracks from noise signals. We do not touch also a problem how would change the algorithm quality due to transition from the asynchronous ANN mode to synchronous one used in parallel ANN implementation.

References

[1] J.J. Hopfield and D.W. Tank, Biological Cybernetics, 52 (1985) 141.

[2] R. Durbin and G. Willshaw, Nature, 326 (1987) 689.

[3] L. Gislén, C. Peterson and B. Söderberg, Neural Computation, 4 (1992) 727.

[4] G. Stimphl-Abele and L. Garrido, Comp. Phys. Commun., 64 (1991) 46.

[5] S. Kirkpatrick, C.D. Gellat and M.P. Vecchi, Science, 220 (1983) 671.

[6] A. Glazov et al., JINR Commun. E10-91-507, Dubna, 1991.

[7] A. Glazov et al., JINR Commun. E10-92-352, Dubna, 1992.

[8] M. Gyulassy and M. Harlander, Comp. Phys. Commun., 66 (1991) 31.

[9] M. Ohlsson, C. Peterson and A.L. Yuille, Comp. Phys. Commun., 71 (1992) 77.

[10] V. Baranov et al., Nucl. Instr. and Methods, B17 (1986) 438.

[11] G. Ososkov, Proc. 2d Intern. Tampere Conf. in Statistics., Tampere Univ. Press (1987), 615.

[12] F. Mosteller and J.W. Tukey, Data Analysis and Regression, Addison-Wesley Publishing Company, 1977.

Received by Publishing Department
on March 19, 1993.