A FUZZY SYSTEM TO DETECT AND COUNT PARALLEL NOISED TRACKS

Ignazio D'Antone INFN (*) Bologna
Enzo Gandolfi Istituto di Fisica Università di Bologna
Massimo Masetti Istituto di Fisica Università di Bologna
Caterina Vitullo INFN (*) Bologna
(*) INFN National Institute for Nuclear Physics

KEYWORDS: Fuzzy, Trigger, VLSI

ABSTRACT

This paper explores the possibility of using a dedicated Fuzzy processor to solve a tipical problem in the trigger system for HEP (High Energy Physics) experiments, that is to count parallel noised tracks in a very short time: few µS. To get this result we decided to have two processors working in parallel and this solution is illustrated in this paper. We are also designing, using VLSI technology, the dedicated Fuzzy processor (here not described). The track counting can accept an error of a few percent. A comparison with traditional solutions is described.

INTRODUCTION

A trigger in HEP experiment is a device which decides whether an event is "interesting" and then must be stored or not (background) according to the configuration of the various pieces of data which make up the event. The event selection is based upon few different conditions: one of these is to identify and count nuclear particle noised tracks which are parallel. This paper explores the possibility of using a hardware dedicated fuzzy processor designed to solve this pattern recognition problem. In most cases the event selection process requires a very short time: that is few microseconds, as in our case.

The Fuzzy system [1], we have developped, is much more powerful to detect and count parallel tracks in presence of positive or negative noise in comparison with a traditional solution where a threshold system is used. To solve the problem we have simulated two Fuzzy systems:

- one with a 6 input processor;
- one with two processors: the first has 4 inputs, the second 3 inputs and they work in parallel.

The second solution is much more attractive both for the point of view of the necessary rules to be implemented and of the processing speed, because, having two processors working in parallel, the necessary time to process one event is much smaller. The number of possibile rules for the two processors also is much smaller than that one for the single processor.

We hope that the hardware processor, here not illustrated, which we are designing at present with VLSI techniques, will work at 50 MIPS (Mega Inferênces Per Second).

The trigger problem, in HEP, has been solved in the past year with dedicated HW, but to have a flexible solution DSP has been added to the system. Recently HW neural networks [2], [3], [4] have also been used. The problem of track finding, from a general point of view, has been faced with neural network [5], while, as far as we know, the problem reported in the title has non yet been faced.

THE INPUT DATA

The data to be processed come from a layer detector and can assume either high or low logic level. This detector is made of some planes (ten in this case), each of them is composed of a pixel linear array (in this case each array is made of 120 elements or pixels). When a particle crosses a pixel, it lights up. With x we indicate the position of the lit pixel along the plane while y indicates the plane itself.

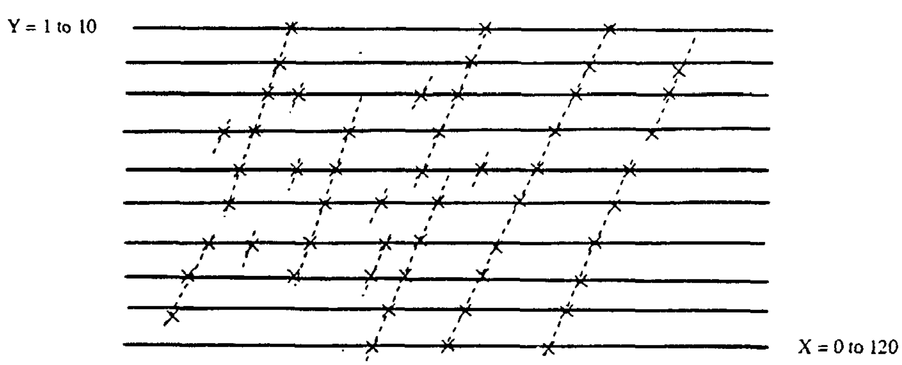

Figure 1. The input data from the layer detector

These data are preprocessed by a dedicated HW both according to the solution with a HW threshold logic and the present solution with a Fuzzy processor as explained in the next paragraphs.

THE TRADITIONAL SOLUTION

In this solution the data coming from the layer detector are organized in the preprocessor in such a way that each x coordinate is followed by the number n of the lit pixels in the different planes. The x data range from 0 to 120 and n ranges from 0 to 10. X and n are integer data. A tipical input data is shown in fig. 2 together with the threshold circuit which drives a counter.

If there is a track we have a peak for the n variable, this into account. An improved solution could be obtained adding a DSP to process the data, but the related program requires a processing time which may be too long to take the decision, as required by the trigger process. The solution with a fuzzy processor has the advantage of taking into account the above ambiguity problems and also of not being time consuming as a dedicated DSP processor.

THE HW PROPOSED SOLUTION

The fuzzy system proposed for the tracks counting problem is made of a wired hardware preprocessor, two fuzzy processors and a counter. The optimal solution, we found, from the speed point of view, is illustrated in fig. 3. The preprocessor gets data from three adjacent x

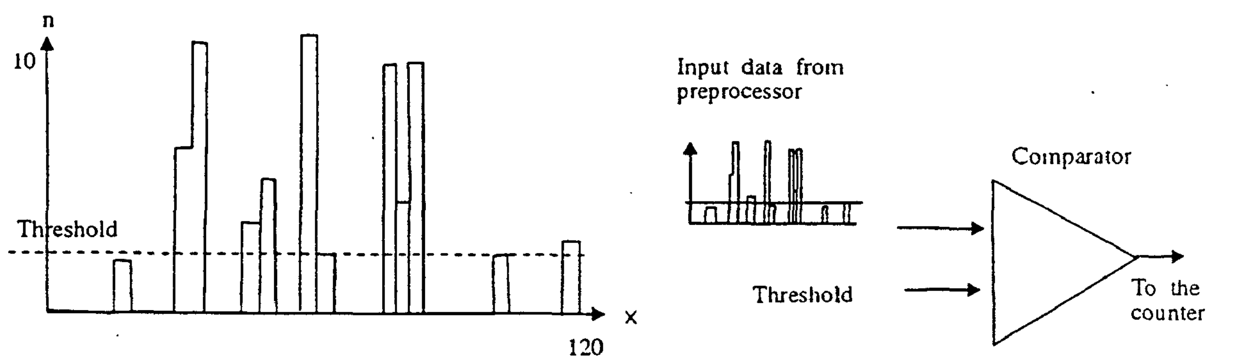

Figure 2. The traditional solution

therefore with a threshold circuit we can count the track number. But because of the noise a single track can occupy two following x positions one after the other, therefore it looks like two tracks. A threshold cannot take positions which we call adjacent columns. It selects 6 variables, 2 for each of the three columns, data are sent to the fuzzy processors then it moves from one column to the next one and so on.

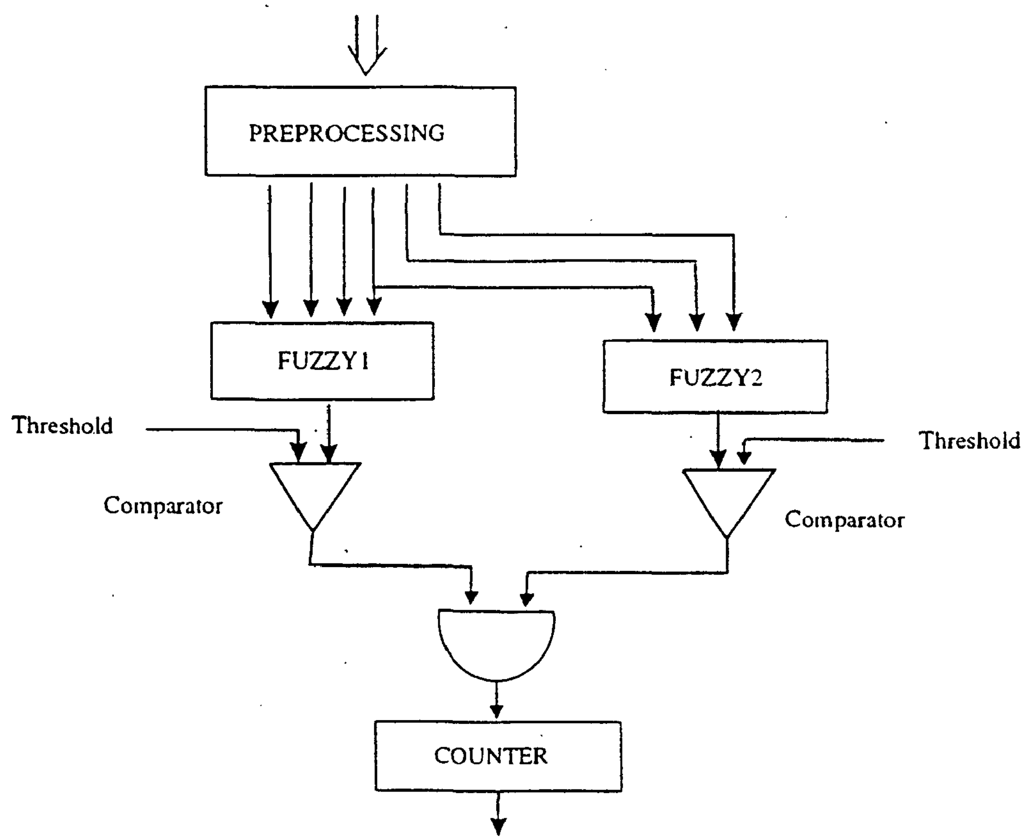

Figure 3. the proposed solution with Fuzzy processors

The variables for each column are:

1) the number of the lit pixels,
2) the barycentre of the lit pixels.

The two fuzzy processors, working in parallel, have one input in common. The inputs and the outputs are analog. The two outputs are compared with a reference signal, threshold, in two comparators whose outputs are the inputs of an AND gate. The threshold level has been determined by the output of the Fuzzy processor whose input are real data. A counter computes the total number of detected tracks.

THE FUZZY LOGIC SYSTEM DEVELOPPED

The fuzzy logic system consists of two parts: optimization of membership functions and implementation of fuzzy rules.

Optimization of membership functions

A Fuzzy neural network model [6] is used to find the optimal I/O parameters of membership functions. In our application the membership functions are Gaussian functions and the center and the width are computed by self-organized learning techniques analogous to statistical clustering. This allows the domains of membership functions to cover only the regions of the I/O space where data are present. Kohonen algorithm [7] is adapted to find the center m_j of Gaussian function. Given the training input data X_i and the total number of membership functions k for each linguistic variable, the algorithm consists of an adaptive formulation which runs for each I/O variable:

$$\|X(t) - m_{closest}(t)\| = \min (\| X(t) - m_j(t) \|)$$
$$\text{where} \quad 1 \le j \le k$$
$$m_{closest}(t+1) = m_{closest}(t) + \alpha(t).(X(t) - m_{closest}(t))$$
$$m_j(t+1) = m_j(t) \qquad \text{for } m_j \neq m_{closest}$$

where $\alpha(t)$ is a monotonically decreasing scalar learning rate.
Once all the centers are found, the second parameter σ_j (width) is determined by the heuristic first-nearest-neighbor as:

$$\sigma_j = \frac{|m_j - m_{closest}|}{r}$$

where r is an overlap parameter.

Implementation of Fuzzy rules

The three barycentres and the counting of the present column are the four inputs of the first Fuzzy processor, while the three countings are the inputs of the other one. The knowlodge can be expressed by an expert, writing the necessary rules that describe the process, among all the possibile rules. The inference method implemented is the MAX-MIN composition and the defuzzification process is the centroid scheme [8], [9]. The Fuzzy system description of the first processor, whose inputs are the three barycentres and the counting of the present column, is shown in the following table:

VARIABLES	DOMAINS
X1	L, M, H
X2	L, M, H
X3	L, M, H
X4	VL,L,M,H
Y	YES, NO

where:

X1 = barycentre of the lit pixels in the present column,
X2 e X3 = barycentres of the lit pixels in the two previous columns,
X4 = counting of the lit pixels in the present column,
Y = output.

VL = Very Low, L = Low, M = Medium, H = High,
YES = YES track, NO = NO track.

In the first Fuzzy processor, the rules have been written taking into account that a track can be selected if there is a Medium - High number of counting or if there is a Low number of counting and the barycentres of three adjacent columns are lined. In this way we take into account the possibility that a particle may cross the planes not exactly at 90°.

We report here, for example, two rules:

- If X1=H and X2=M and X3=L and X4 L => YES
- If X4=VL => NO

The Fuzzy system description of the second processor, whose inputs are the counting numbers, is shown in the following table:

VARIABLES	DOMAINS
X4	VL,L,M,H
X5	VL,L,M,H
X6	VL,L,M,H
Y	NEW, OLD

where:

X4 = counting of the lit pixels in the present column,
X5 e X6 = counting of the lit pixels in the two previous columns,
Y = output.
VL = Very Low, L=Low, M= Medium, H = High;
NEW = NEW track, OLD = OLD track.

In the second Fuzzy processor, the rules have been written taking into account that a new track can be selected if the counting in the present column increases.
We report, for examples, two rules:

If X4=H and X5=M and X6=L => NEW
If X4=L and X5=H and X6=H => OLD

In fig.4 a schematic picture of the input data, and the related detected tracks is shown. Only a part of the ten planes are displayed, and 1 means a lit pixel by the track or the noise. The arrows on the top of the picture illustrate the selected tracks.

RESULTS

The described system has been able to detect and count parallel tracks with a very good accuracy. If we refer to figure 4, where only part of the data are shown, only three tracks are present and they have been correctly counted even if the noise was very high. In figure 4 the tracks are vertical while in fig. 1 they have a slope: the necessary rotation, to process vertical lines, is performed by the hardwired preprocessor. The first two tracks, to be detected, starting from the left, are disturbed by the random noise, while the third one has a noised track at its right side which is not counted. In the figure the represented tracks are strictly vertical, but they may also have a small slope and the developped rules take it into account successfully.

```
          ↓     ↓           ↓
0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 0
0 0 0 0 0 1 0 0 1 0 0 0 0 0 1 1 0
0 0 0 0 0 1 0 0 1 0 0 0 0 0 1 1 1
0 0 0 0 0 1 0 0 1 0 0 0 0 1 1 1 0
0 0 0 1 1 1 1 1 0 1 0 1 0 1 1 1 0
0 0 1 1 1 1 1 1 0 1 1 0 1 1 1 0 0
0 0 1 1 1 1 1 1 1 1 1 0 1 0 1 1 1
0 0 0 0 0 0 0 0 1 1 0 0 1 1 1 1 0
0 0 0 0 0 1 0 0 0 0 0 0 0 1 1 1 0
0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
```

Figure 4. An example of part of the data to be processed

CONCLUSIONS

In this paper a Fuzzy processor has been used to detect noised parallel tracks as required in a tipycal event selection system for HEP. The same goal could have been reached by using:

1) a hardwired logic that can reach the desired speed but it in not flexible,
2) a DSP where we get the required flexibility but it is not possible to reach the desired speed,
3) a hardware neural network which is flexible and speedy but it does not allow to optimize "on-line" its performances.

The Fuzzy system presented matches all the necessary requirements: speed, flexibility and on line optimization.

Address:
Istituto di Fisica
Via Irnerio 46 40126 Bologna Italy
FAX +39 51 247244
Email: MASETTI@VAXBO.BO.INFN.IT

References

[1] L. A. Zadeh
Fuzzy Sets
Inform. Control, vol 8, pp. 338-353, 1965

[2] B. Demy, M. Campbell, F. Bedeschi, N. Chriss and F. Nesti
Neural Network for Triggering
IEEE Transaction on Nuclear Science Vol. 37, N. 2; April 1990

[3] P. Ribarics
A Second Level Neural Network Trigger in the H1 Experiment at HERA
1992 IEEE Nuclear Science Symposium and Medical Imaging Conference

[4] C. S. Lindsey, B. Demby, H. Haggerty
Drift Chamber Tracking with Neural Network
1992 IEEE Nuclear Science Symposium and Medical Imaging Conference

[5] G. Barbagli, G. Castellini, G. Landi, S. Vettori
A study of track finding with a neural Network Algorithm
International Journal of Neural Systems
Vol 3 (supp. 1992) 303-308

[6] C.T. Lin, C.S.G. Lee
Neural-Network-Based Fuzzy Logic Control and Decision System
IEEE Transaction on Computers
vol. 40, no. 12 - dicembre 1991

[7] T. Kohonen
Self-Organisation and Associative Memory
Berlin, Germany: Springer-Verlag 1988 p. 132

[8] B. Kosko
Neural Networks and Fuzzy Systems
Prentice Hall International Editions 1992

[9] Gianguido Rizzotto
Strutture dedicate in logica Fuzzy
Elettronica oggi - N 150 - 15 Novembre 1992