Intelligent preprocessing for neural networks in the H1 experiment

JC.Prévotet[1], B.Denby[1], P.Garda[1], B.Granado[1], W.Fröchtenicht[2], G.Grindhammer[2], L.Janauschek[2], C.Kiesling[2], T.Kobler[2], B.Koblitz[2], S.Schmidt[2], B.Tzamariudaki[2], S.Udluft[2]

[1]Laboratoire des Instruments et Systèmes, Université Pierre et Marie Curie, Paris VI, France
[2]Max Planck Institut für Physik München, Germany

Abstract. After the upgrade of the HERA machine at DESY in 2001, an increase in the luminosity of a factor 5 is expected. Since the data output rate of the L2 trigger should be kept at the pre-upgrade level, a smarter way of preprocessing data has been developed, extracting the most physically relevant information in order to optimize the neural networks. We describe here the new neural preprocessor DDB2 (Data Distributed Board) and focus especially on the algorithmic principles. A general overview of the hardware implementations of such algorithms is then discussed, notably the use of the current, fast FPGA technology to combine parallelism and speed.

INTRODUCTION

The L2 neural network trigger in the H1 experiment has shown its effectiveness since 1995, producing interesting physics results. Some of them are described elsewhere (1). The next technical improvements of the Hera collider will lead to better performances forcing the trigger selectivity to improve. For this purpose, a more intelligent pre-processing has been envisaged to keep the data flow workable but reduce the dimensionality of the problem by exploiting more correlations between data. The information is built on properties of particles emitted during a collision (momentum, energy, type of particle) collected from all sub-detectors in the H1 experiment. Using such “intelligent” input variables, the efficiency of the neural networks is consequently improved, resulting in a better discrimination between interesting events and background. The pre-processing is performed in four steps. The first is the clustering in which all signals within one sub-detector, localized in the same area are associated in order to constitute one entity. Clusters found in all subdetectors are then combined into objects according to their angular coordinates before being sorted according to particuliar criteria. The final step is called post-processing and is used to execute straightforward calculations and generate variables to make them directly usable for the neural networks. All these algorithms have to be implemented in hardware since the process must be executed in real time (8μs). We summarize here the hardware specifications and describe the DDB2 architecture.

ALGORITHMIC PRINCIPLES

Clustering

The basic idea of the clustering is to arrange signals within a specific subdetector from their position within a θ,ϕ plane. Due to the nature of the subdetectors, data may represent energies coded in 8 bits or have single bit fields, leading to two different kinds of processing.

Energy Clustering

LAR clustering: Clustering in the liquid argon (LAR) calorimeter is performed separately in the electromagnetic (EM) and the hadronic (HA) parts of the calorimeter. Data are coded in 16 bits (8 address bits + 8 data bits) representing respectively their position in the (θ,ϕ) plane and their energy value (Fig.1). The incoming data are picked out by energy and the cluster having the largest one is taken as a center. The positions of its neighbors are then determined via LUTs (Look Up Tables) and their energies are summed. Up to 16 neighbors are tolerated although 9 or 10 are sufficient in principle. As output from this algorithm, parameters such as the energy of the cluster's center (E_{cent}), the summed energies of the adjacent neighbors (E_{ring}), the total energy (E_{tot}), the number of non-zero cells in the neighborhood (N_{hit}) and the angular coordinates of the cluster's center (θ,ϕ) are obtained.

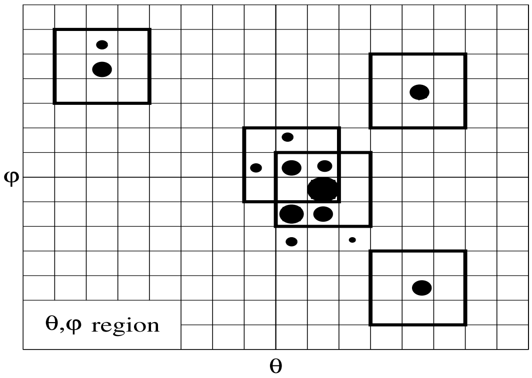

FIGURE 1. Clustering algorithm for the calorimeter information

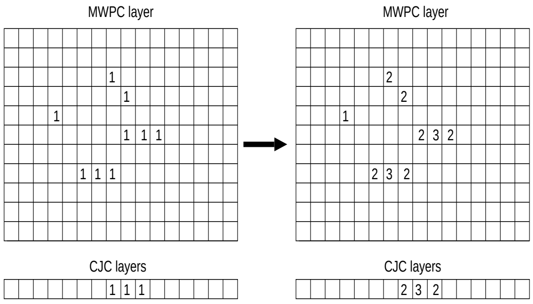

FIGURE 2. Preclustering algorithm for bit fields: MWPC above, and Drift chambers below.This step is necessary before applying the clustering algorithm.

Bit field clustering

MWPC clustering: Data coming from the multiwire proportional chambers (MWPC) are bit fields distributed on a 16x16 grid. Because of the binary nature of data, a preclustering step is necessary before applying the clustering algorithm itself. All non zero bits in the map are summed with their corresponding neighbors and the map is updated. See Fig.2.

Similar clustering principles as in the LAR are then performed starting with the largest cluster. Parameters such as P_{tot}, P_{cent}, P_{ring}, N_{hit}, θ, ϕ are also calculated here.

CJC clustering: Data coming from the central jet chamber (CJC) provide only the ϕ information, delivering a one dimensional array of 45 bits. Four arrays are provided, for high momentum negative tracks, high momentum positive tracks, low momentum positive tracks, low momentum negative tracks. The clustering procedure is simpler than for the other sub-detectors, due to the single dimensionality. Drift chambers parameters such as D_{tot}, D_{cent}, D_{ring}, n_{hits}, θ, ϕ are provided as outputs.

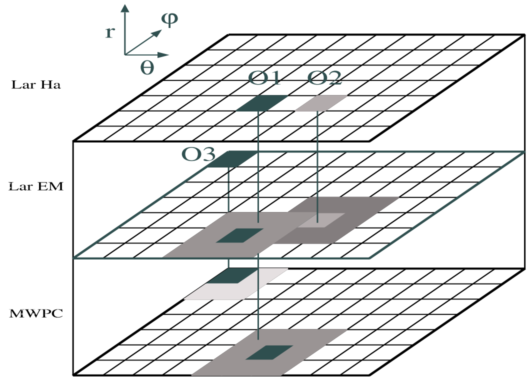

FIGURE 3. Example of the matching algorithm in the first 3 layers. Here are 3 objects found.

The matching algorithms

Two matching algorithms work on data coming from subdetectors belonging to two different angular ranges. The first one matches the angular coordinates of clusters coming from the HA and EM part of the calorimeter, MWPC, CJC, muon, FTT *(fast track trigger)*. The second processes the data coming from another angular region of the detector : The SpaCal trigger chambers, the muon chamber and the backward silicon tracker (BST). These algorithms consist in detecting a cluster with the maximum energy in a seed layer and compare its coordinates (θ, ϕ) with the coordinates of all available clusters in the other sub-detectors. See Fig.3. A match is defined by $(|\Delta\theta|,|\Delta\phi| <$n where n can be set to a value in the range 0-16). The number of clusters within a layer is configurable but limited to 16 and each layer is transformed according to a universal (θ, ϕ) grid in order to have the same basis for coordinate comparisons. Clusters matched are removed to avoid double-counting, except those belonging to the CJC layers which can be used several times since they only contain phi information. The outputs of these algorithms consist of maximum 16 objects, constituted by the properties of the clusters that form it.

Sorting

All objects are sorted according to selectable properties. Three sorting procedures are executed in parallel, providing three different lists of sorted objects. If the sorting key is not unique, a second criterion is taken. Quantities like energies and angular coordinates can be sorted.

The second role of this algorithm is to modify the object's coordinates, redetermining the angular information relative to the biggest object in the lists.

Post-processing

This algorithm constitutes the last step of the preprocessing chain in which many types of calculations can be made in order to optimize the data input for the neural networks. At present, calculation of straightforward quantities such as cluster counting, angular differences between leading clusters are performed. The addition of new features is still under investigation.

HARDWARE CONSIDERATIONS

The different preprocessing algorithms are destined to be implemented on programmable boards with an execution time constraint of 8μs.

L2 Trigger Hardware organization

The same configuration has been kept as in the previous preprocessing. More details are given in (2). Data coming from the sub-detectors are collected in receiver cards and sent via a 8 by 16 bits L2 bus to the DDB boards in which the preprocessing is performed. DDB boards reside in a VME crate and are connected to their corresponding CNAPS board which implement the neural networks. All networks are 3 layer feed forward MLPs and are trained individually for specific physics processes. The CNAPS boards run at 20.8 MHz and deliver a result in 8μs. Further technical details about the L2 neural network trigger are given elsewhere (3). Five new DDB2 boards and their associated CNAPS boards are foreseen to be concurrently added to the previous existing ones, offering two kinds of preprocessing.

The DDB2 (Data Distribution Board)

An overview of the board is shown in Fig.4. Data coming from all sub-detectors arrive in parallel on 8 sub-buses constituting the L2 bus and are directly provided to the corresponding clustering modules in which they are sorted. A preclustering step is previously executed in the case of bit field data in order to find a maximum before actual clustering. The addresses are then sent to their associated matching modules to determine the vicinity within all layers, while the parameters related to a specific cluster are stored in memories.

After the matching step, addresses of matched clusters constituting an object are provided and used as pointers to the RAMs in which the parameters have been previously stored.

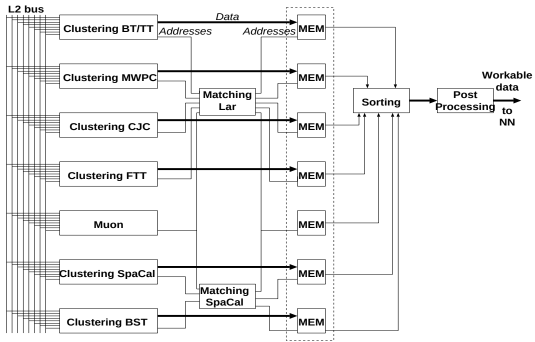

FIGURE 4. Description of the DDB2 board

Parameters are then accessible and are sent to the sorting machine for list generation before being processed in the last post-processing unit and sent to the neural network.

All modules have made intensive use of pipeline structures and parallel computation techniques in order to meet the timing constraint requirement. Moreover, the need of a large amount of small memories used as LUTs has opened the way to use specific FPGA circuits. The Xilinx Virtex family was chosen to implement such designs providing more flexibility and lower development costs than standard ASICs.

CONCLUSION

The DDB2 preprocessor represents an important innovation in the preprocessing step of the L2 neural network trigger because it takes advantage of more physically relevant information. The design have already been simulated and significant improvements have been noticed. More details about the DDB2 can be obtained in (4). Concerning the hardware, FPGA circuits present an effective solution in terms of speed and flexibility and seem appropriate to process all algorithms in 8μs.

REFERENCES

1. Janauschek, L. et al., in *New Computing Techniques in Physics Research VI,Greece* .
2. Kiesling, C. et al., in *these proceedings* .
3. Köhne, J., *Nuclear Instruments and Methods in Physics research* .
4. Schmidt, S., *Untersuchungen zur Verbesserung des neuronalen Triggers beim H1 experiment*, Master's thesis, Fakultät für Physik der Technischen Universität München, unpublished, 2000.