Artificial Intelligence on FPGAs ATLAS

Georges Aad

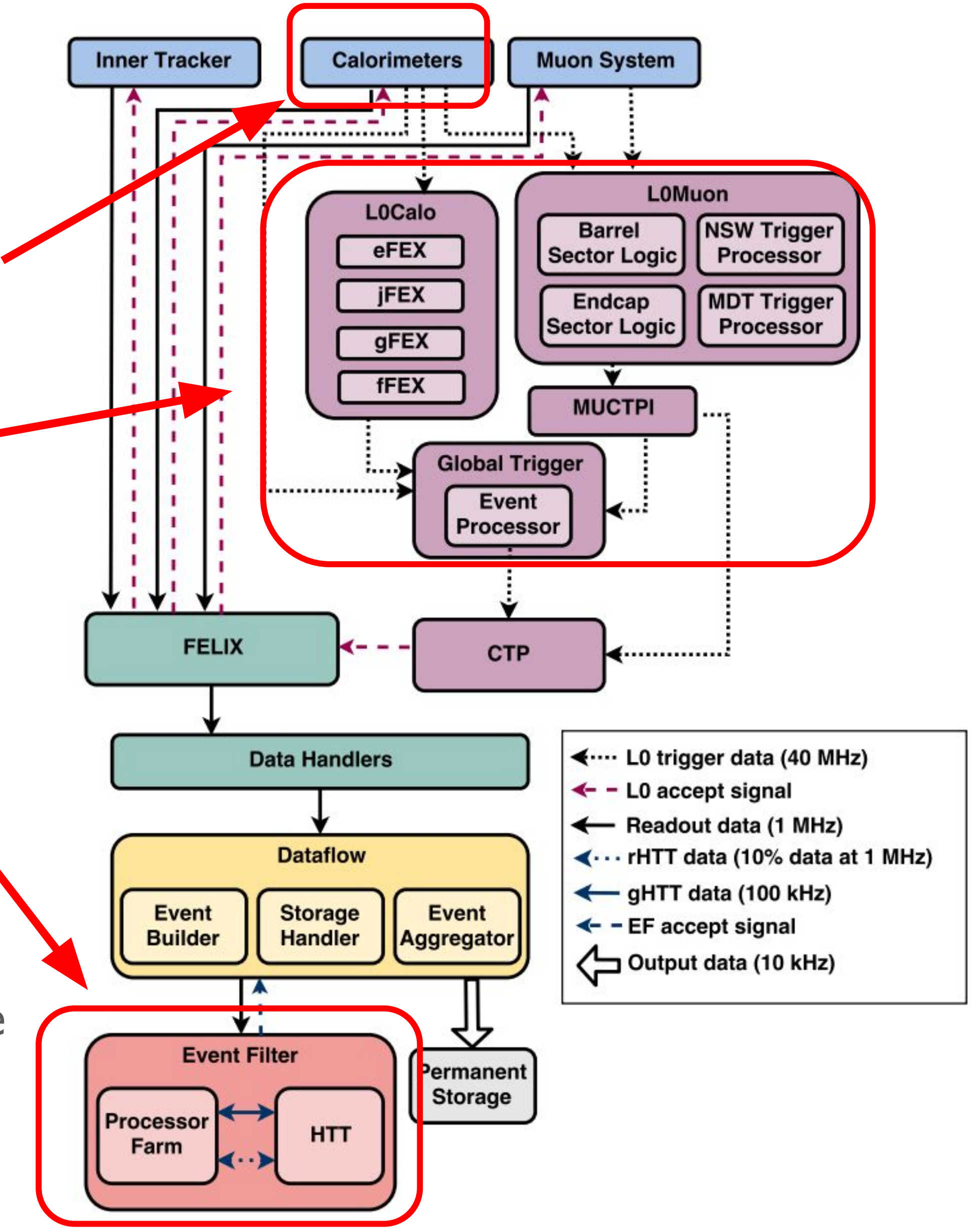
THINK kickoff meeting - 03/03/2020

Disclaimer:
this talk is by no means a comprehensive view of what is done in ATLAS
I present a selection of ideas with preliminary results

ATLAS TDAQ Architecture

- 3 Main stages where Artificial intelligence can be used to improve trigger performance

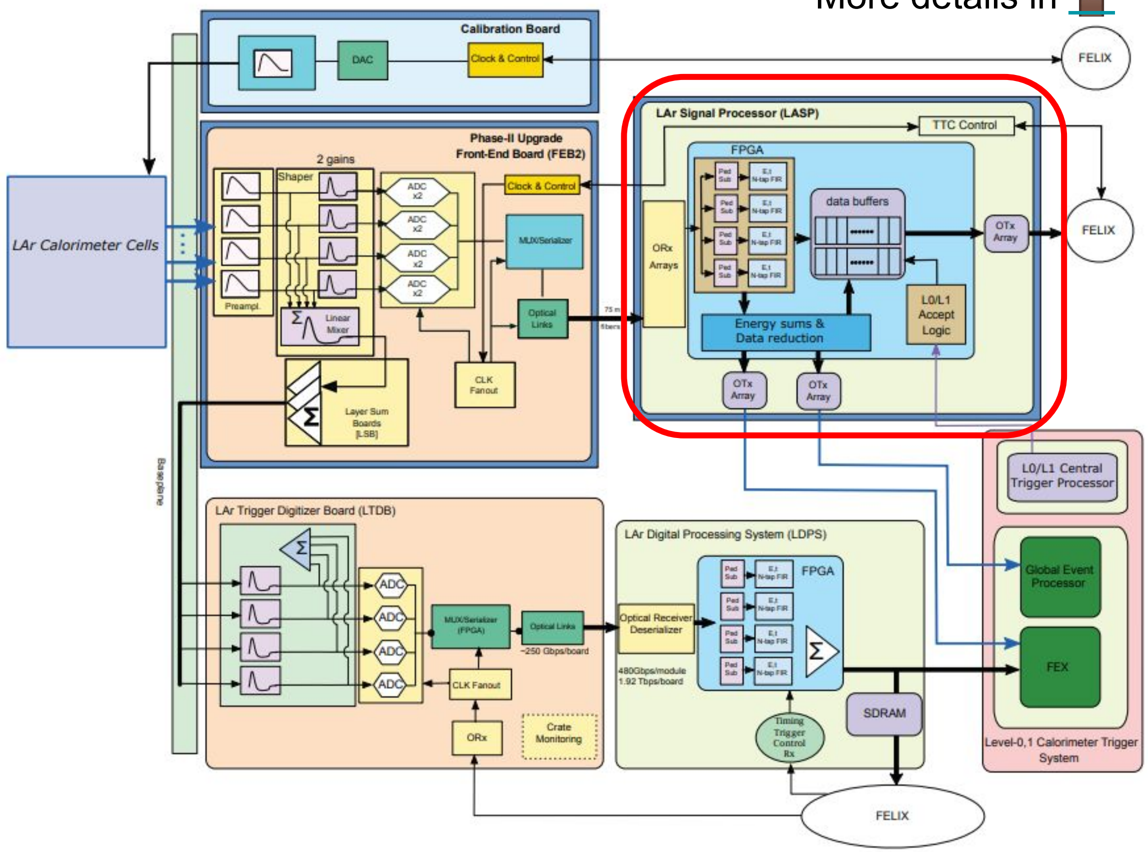
1. Preprocessing of raw detector output
   a. E.g. Computation of energy deposits in the calorimeters
2. Identification of the presence interesting events/objects at L1 (hardware) trigger
   a. E.g. Identifying the presence of muons above a certain pT threshold
3. Reconstruction and identification of objects at the High-Level-Trigger (software)
   a. E.g. Fast electromagnetic shower pre-selection to improve CPU time
   b. Hardware acceleration can be used

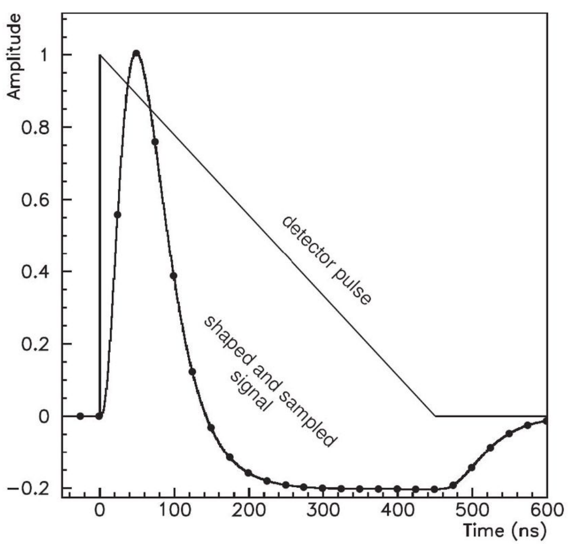


LAr Calorimeter Data Processing using RNNs

- New backend board to compute energy deposited in the calorimeter
  - Based on high-end FPGAs
  - Stratix 10 or Agilex
- Identify BCID (time) of collision and compute the deposited energy
- Total throughput: ~300 Tb/s

More details in



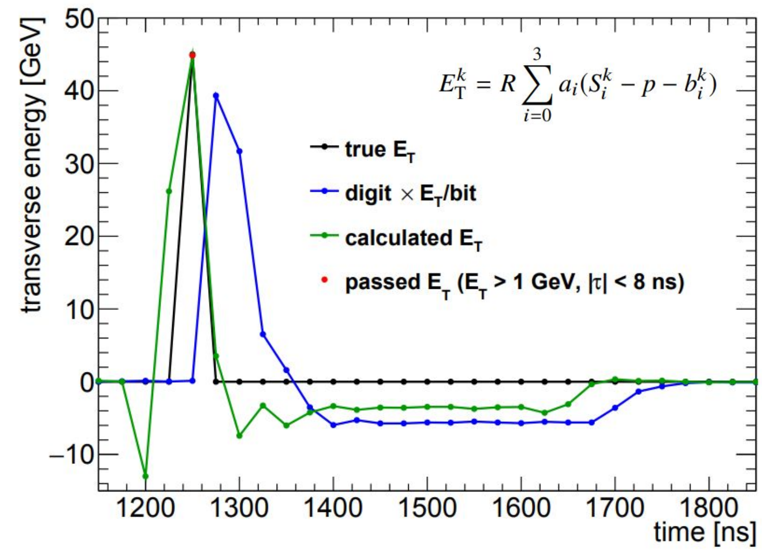


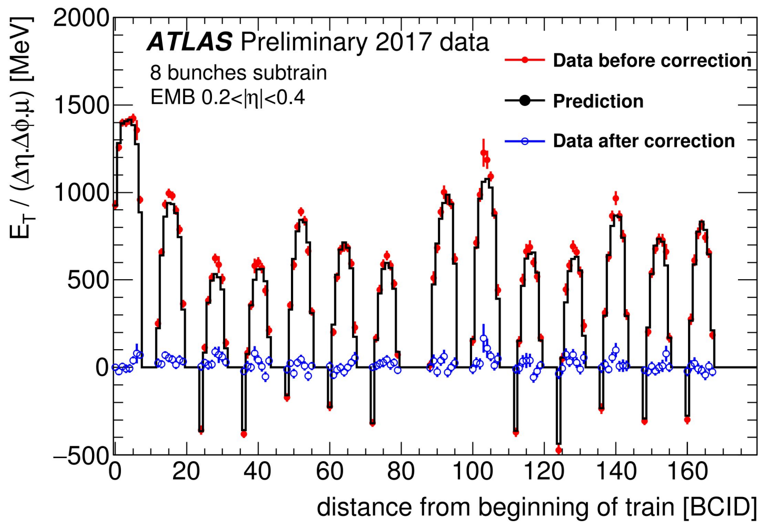
- Electronic signal shaped (bi-polar shape) and digitized at 40 MHz
- Samples (ADCs) around the peak used to compute the deposited energy and detect the deposited time

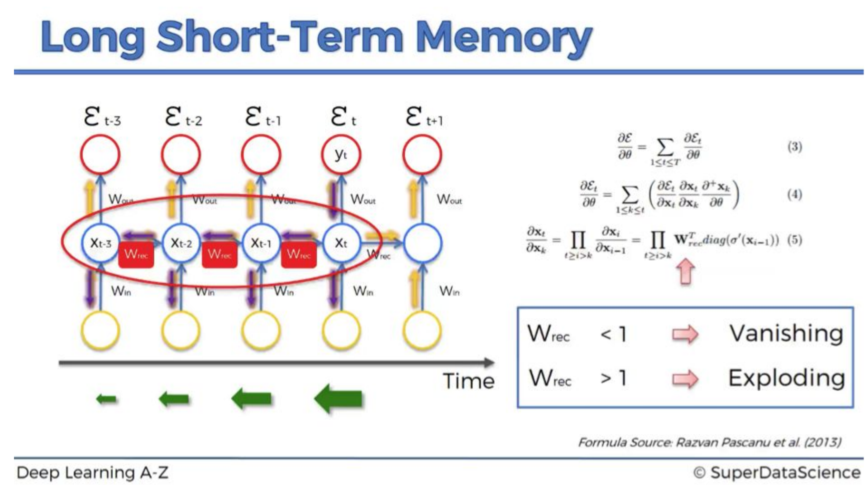
Special Run 2 Optimization studies

More details in

- Current algorithms using optimal filtering technique (assuming perfect pulse shape)
  - Breaks at High pileup
- Bipolar pulse shape designed to cancel out-of-time pileup
  - Breaks with LHC buch train structure
- Use RNNs (LSTM) to compute energy deposit at each bunch crossing
  - Learn energy from shape around the peak
  - Learn pileup contribution from ”history”





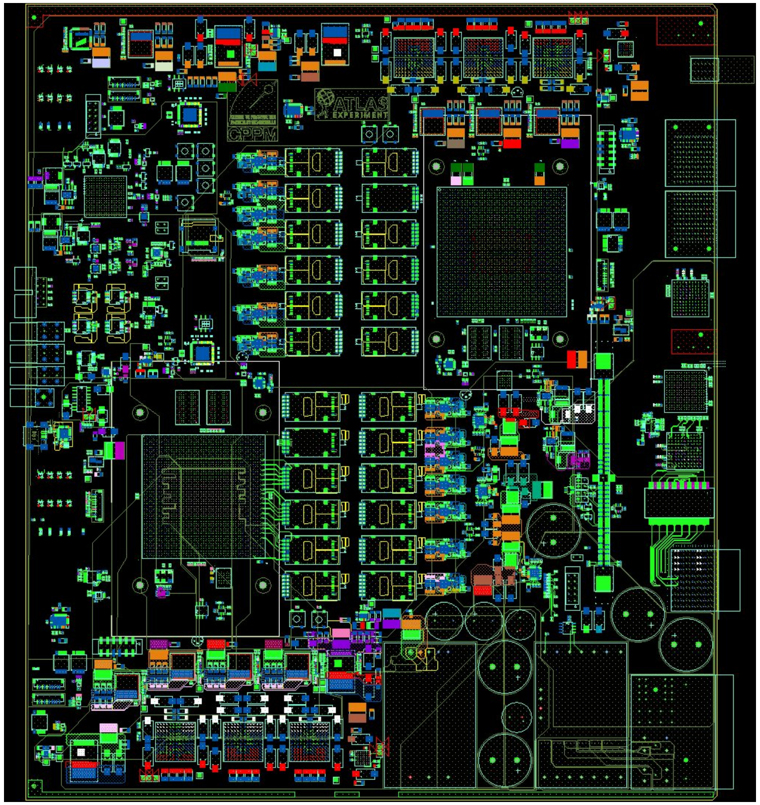


Special Run 2 Optimization studies

	PRODUCT LINE	AGF 004	AGF 006	AGF 008	AGF 012	AGF 014	AGF 022	AGF 027
Resources	Logic elements (LEs)	392,000	573,480	764,640	1,200,000	1,437,240	2,200,000	2,692,760
	Adaptive logic modules (ALMs)	132,881	194,400	259,200	406,780	487,200	745,763	912,800
	ALM registers	531,525	777,600	1,036,800	1,627,119	1,948,800	2,983,051	3,651,200
	eSRAM memory blocks	0	0	0	2	2	0	0
	eSRAM memory size (Mb)	0	0	0	36	36	0	0
	M20K memory blocks	1,900	2,844	3,792	5,568	7,110	11,616	13,272
	M20K memory size (Mb)	38	56	74	110	139	210	259
	MLAB memory count	6644	9720	12960	20,338	24,360	32,788	45,640
	MLAB memory size (Mb)	4.3	6.2	8.3	13	15.6	21	29.2
	Variable-precision digital signal processing (DSP) blocks	1,640	1,640	2,296	4,000	4,510	6,250	8,736
	18 x 19 multipliers	2,300	3,280	4,592	8,000	9,020	12,500	17,056
	Single-precison or half-precision tera floating point operations per second (TFLOPS)	1.7 / 3.4	2.5 / 5.0	3.5 / 6.9	6.0 / 12.0	7.0 / 13.9	9.4 / 18.8	11.8 / 23.6

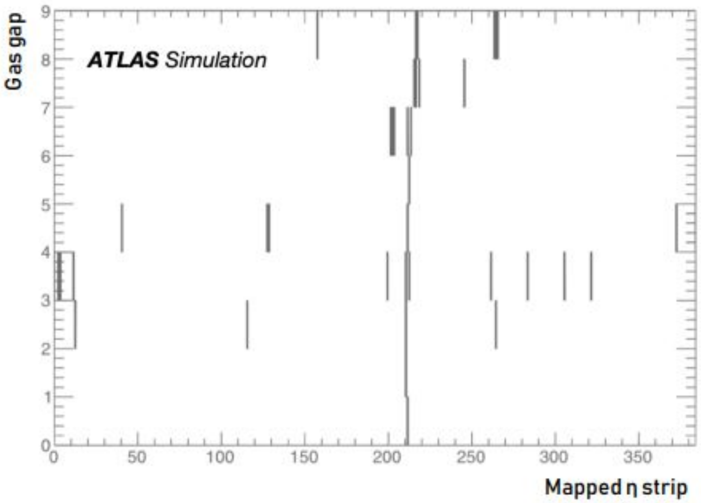
- Order of ~100 or ~50 input fibers per FPGA with 12 channels each
  - Same order for the number of NNs should be implemented in the FPGA
  - Depends on serialisation capacity
    - Larger (x[2-3]) latency with respect to phase 1 (order of 300 ns can be available for this processing)
- Need FPGA with maximum logic and DSPs
  - x[5-7] with respect to phase 1 available
- Very important to reduce (prune) the NN and to share logic between channels

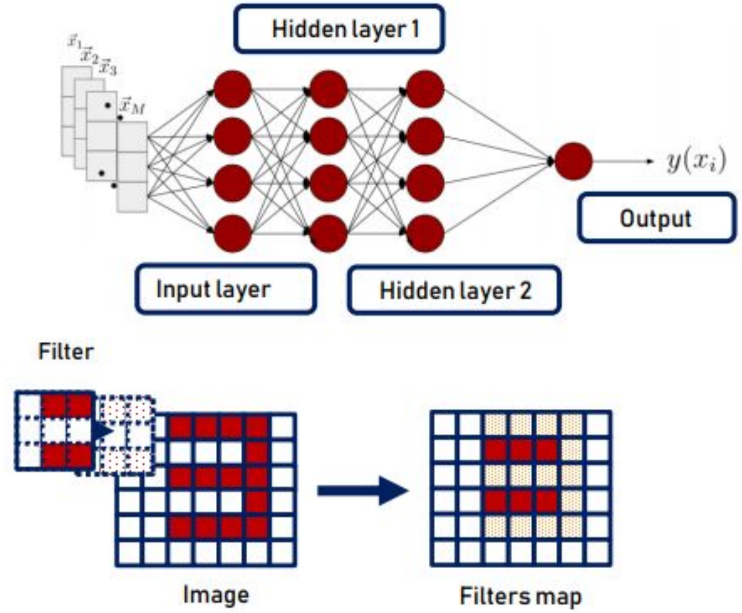
Prototype in development at CPPM



Muon Identification at L1 Using CNN

More details in

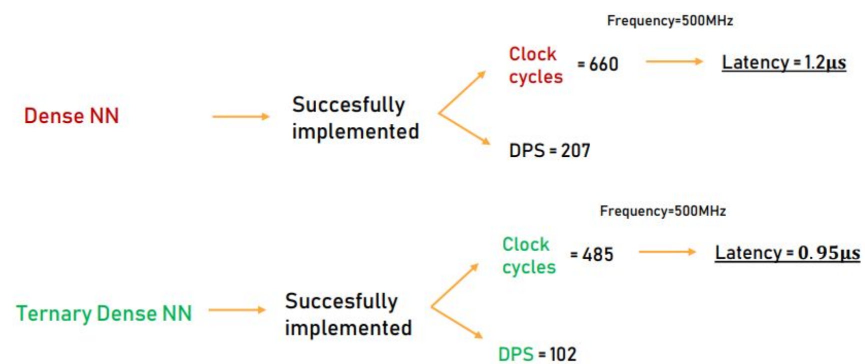




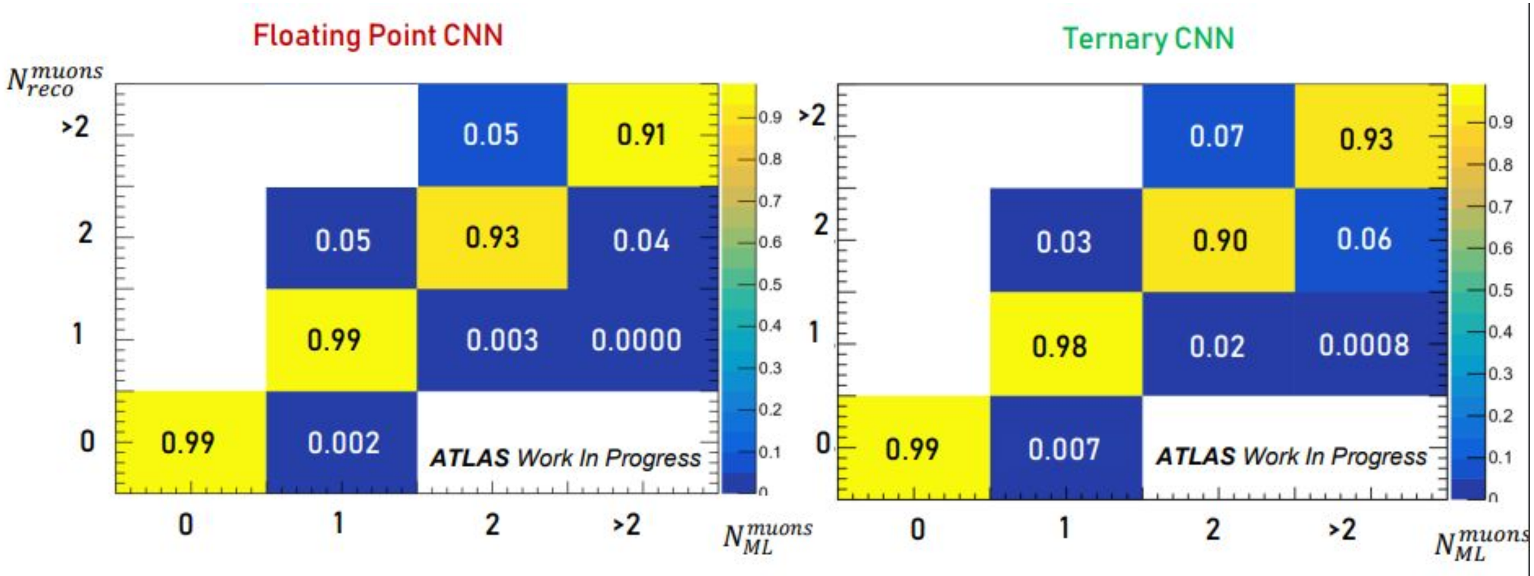
- Transform muon hits into a 2D picture
  - CNN used to detect muon patterns in the picture
  - 5D output: pT and eta of the 2 leading muons and Number of muons
- NN with 500K parameters
  - Tested with 32 bit floating point precision and ternary NN (2 bits, up to a factor 16 reduction)

Implementation ongoing on Virtex7 FPGA using

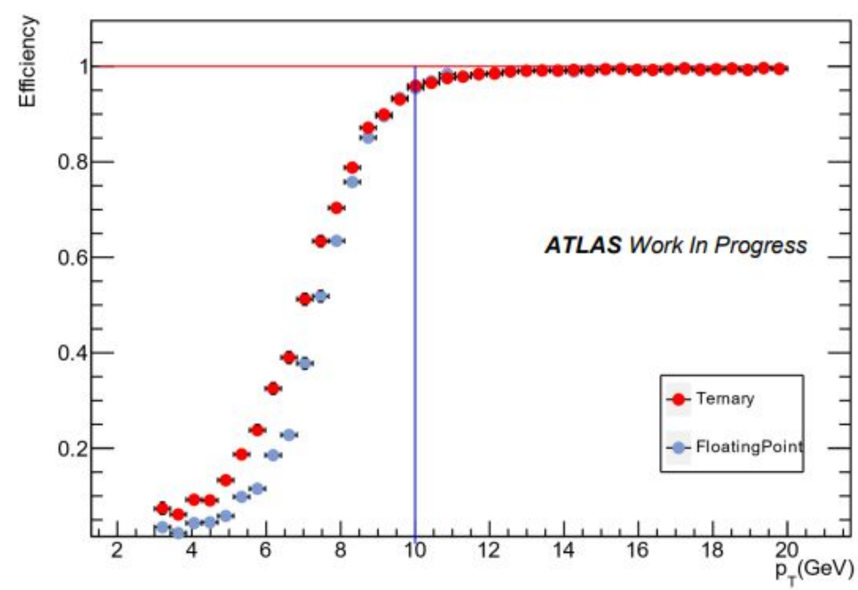
hls4ml



Muon Identification at L1 Using CNN

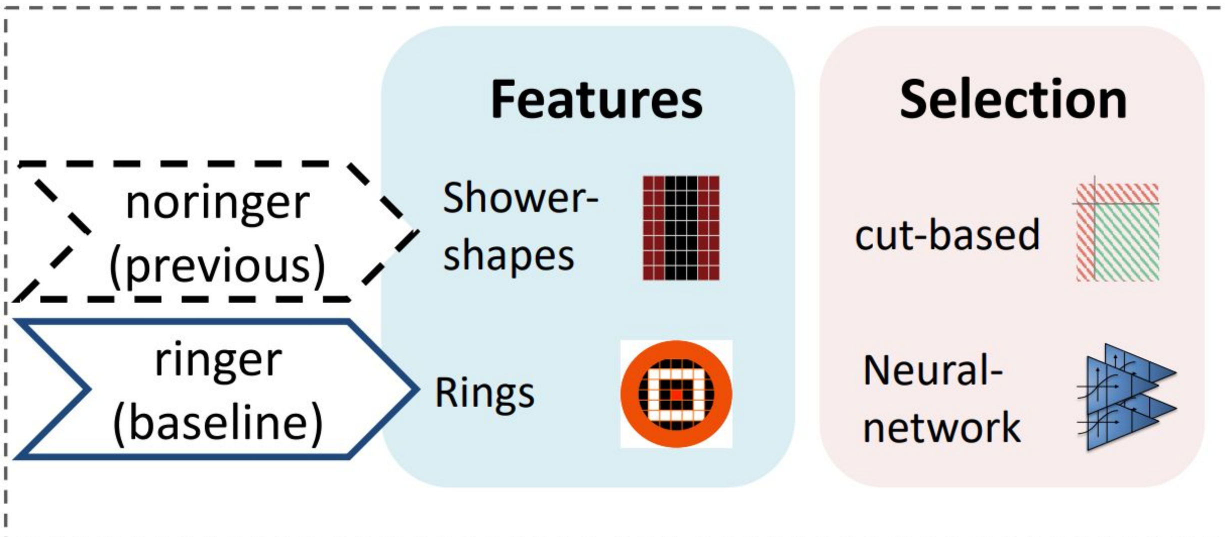


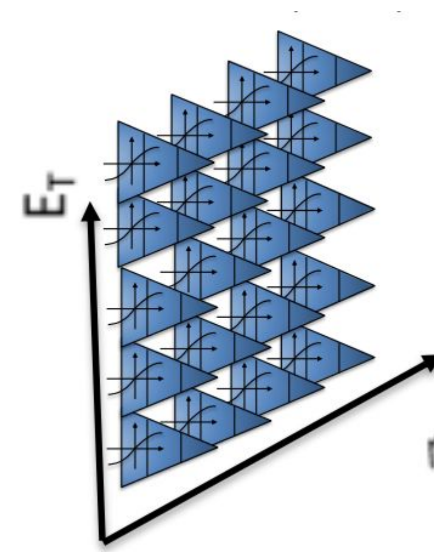
- More information with respect to standard algorithms
  - Provides properties and number of muons
- High efficiency down to low pT muons
  - Need to check trigger rates
- Usage to ternary NNs does not have a significant impact on the performance
- Plans to port this development to Ultrascale+ FPGAs



Electromagnetic Shower Selection Using MLP

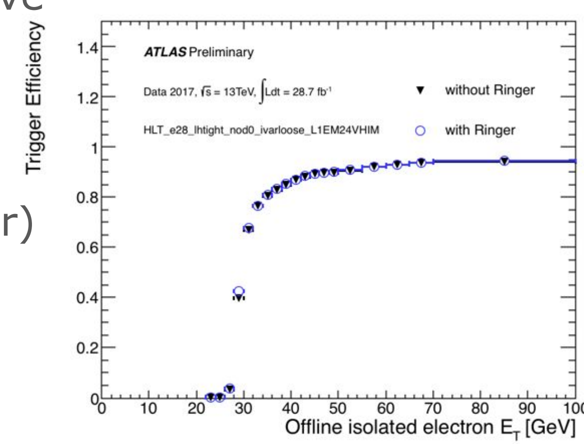
More details in





Single hidden layer fully connected (dense) MLP

- Goal: Improve electron/photon pre-selection at the HLT level to reduce CPU consumption
  - Idea can be ported to L1 (hardware) trigger to improve L1 trigger rates and efficiency
- Idea: Energy sum in rings as input to a NN
  - Assume perfect Cone for the shower shape
  - Takes into account longitudinal shape (sums per layer)
- No trigger efficiency loss by applying a selection
- Should parametrise as function of ET and eta
  - Non regularities in the detector
- Many small NN (MLP) for different topologies
  - Implementation in FPGAs (at L1) to be investigated



Conclusion

- Using Artificial intelligence at early processing stages (mainly trigger) of data processing in ATLAS is in its infancy period
    - Great opportunity for R&D and new ideas
- Few projects ongoing with preliminary ideas and results
    - We should have clearer results in the next months/years
- Preparation for phase II ongoing now
    - Hardware mostly fixed but still some time for few changes
    - Need to have a clearer view on the needed for possible NN based algorithms as soon as possible
- In parallel industrial development are in fast expansion in this domain
    - Should follow this expansion and maybe contribute to it
- Plan to test on LASP boards soon
    - RNN and LSTM for LAr signal processing
    - But can test also other architectures like CNNs and simple MLP.