ATLAS Jet Trigger Update for the LHC Run II

Ademar T. Delgado
On behalf of the ATLAS Collaboration

Abstract—The CERN Large Hadron Collider is the biggest and most powerful particle collider ever built. It accelerate bunches of particles that cross up to 40 million times per second, in each crossing dozens of proton-proton collisions are generated at unprecedented energies, to explore the fundamental laws and properties of nature. The ATLAS experiment is one of the detectors that analyses and records these collisions. It generates dozens of GB/s of data that has to be reduced before it can be permanently stored. The event selection is made by the ATLAS trigger system, which reduces the data volume by a factor of $10^5$. The trigger system has to be highly configurable in order to adapt to changing running conditions and maximize the physics output whilst keeping the output rate under control. A particularly interesting pattern generated during collisions consists of a collimated spray of particles, known as a hadronic jet. To retain the interesting jets and efficiently reject the overwhelming background, optimal jet energy resolution is needed. Therefore the jet trigger software requires CPU-intensive reconstruction algorithms. In order to reduce the resources needed for the reconstruction step, a partial detector readout scheme was developed, which effectively suppresses the low activity regions of the calorimeter. In this paper we describe the overall ATLAS trigger software, and the jet trigger in particular, along with the improvements made on the system. We then focus on detailed studies of the algorithm timing and the performance impact of the full and partial calorimeter readout schemes. We conclude with an outlook of the jet trigger plans for the next LHC data-taking period.

Contents

I Introduction 1

II The ATLAS trigger system 2
   II-A The HLT software. 2
   II-B The High-Level Jet Trigger. 3
   II-C The Partial Calorimeter Scan 3

III Jet Trigger physics performance during Run 1. 4

IV The HLT calorimeter software improvement for Run 2. 4
   IV-A Calorimeter algorithms speed-up 4
   IV-B Partial Scan readout scheme 5

V Conclusions. 5

References 6

I. INTRODUCTION

The CERN Large Hadron Collider (LHC) [1] was build to explore the fundamental constituents of nature and the forces between them, at unprecedented energies. It is a circular accelerator with a perimeter of 27 km, where two proton beams cross 20 or 40 million times per second producing 20 to 40 collisions on average in each intersection. The LHC started colliding protons in 2010 at an energy of 7 TeV. In 2011 the energy was increased to 8 TeV and operated in this regime until the beginning of 2013. After these three years of successful data taking, known as Run 1, the LHC was shut-down for two years of maintenance and improvement. In 2015 the LHC will resume operation, for a Run 2, with an energy of 13 TeV, increased luminosity of $2 \times 10^{34} \text{cm}^{-2}\text{s}^{-1}$ and a proton bunch separation of 25 ns, instead of the Run 1’s 50 ns. This will increase the average number of particles crossing the detectors along with the average particle energy, tending to increase the reconstruction load and making more difficult the separation of patterns from different particles that end superimposed in the same detector region. These effects are known as pile-up.

The ATLAS experiment [2], which analyses and records these collisions, is one of the two LHC multi-purpose particle detectors. It has 46 m length and 25 m diameter with a cylindrical shape, and is made of subcomponents with different purposes: an inner tracking detector surrounded by a thin solenoid magnet, electromagnetic and hadronic calorimeters, and a muon spectrometer. The $10^8$ electronic channels of the detector are read out every 25 ns (synchronously with the LHC clock), resulting in events with an average size of 1.6 MB. The total data volume of $\sim 64$ TB/s then becomes unfeasibly large to be recorded or processed with precision on the fly. However only a small fraction of these events have a significant physics interest. The selection of which events should be kept for later analysis is made by the ATLAS trigger system[3], which probes the event data against a menu of desirable physical characteristics, typically the presence of a certain physical object (e.g. a highly energetic electron), that leaves a distinctive pattern in the detector. This menu contains a few thousands of different possibilities. The event selection is hampered by background patterns which mimic the desired objects. Each event has to be processed in real-time and the data volume reduced by a factor of $10^5$ to $\sim 600$ MB/s (400 Hz) for offline storage, in Run 1.

A particularly interesting pattern, known as hadronic jet, is initiated by an energetic quark or gluon and consists of a collimated spray of particles leaving energy depositions in both calorimeters. The energy deposits are joined up by the reconstruction algorithms and calibrated to provide the jet momentum measurement. The component perpendicular to the

\[ \ln \tan(\theta/2). \]
z-axis, known as transverse momentum ($p_T$), is then calculated. Jets are the most common high $p_T$ objects produced at the LHC and are important for a wide range of physics analyses. The jet trigger software selects events containing high $p_T$ hadronic jets. The event reconstruction is uniquely challenging at a hadron collider where a large fraction of events have significant jet activity. The higher Run 2 energy, luminosity and pileup will increase the jet event rate at least by a factor of five. Therefore action is needed in order to keep the trigger thresholds low and acceptable trigger rates.

In this paper we describe the overall ATLAS trigger, the limitations identified during the first run of the LHC and the improvements made in the system for the next data taking period, with an emphasis on the speed-up of the calorimeter software. We then focus on the particular case of the jet trigger, presenting some physics performance results from Run 1, an alternative data reduction algorithm developed for Run 2, detailed timing studies and the performance impact of two different detector readout schemes. We conclude with an outlook of the jet trigger plans for the next LHC data-taking period.

II. THE ATLAS TRIGGER SYSTEM

![Figure 1: Simplified description of the ATLAS Trigger and DAQ system architecture. In the left side the three level architecture from Run 1: The hardware based Level-1 and the two software based Level-2 (L2) and Event Filter (EF). In the right side, the new Trigger architecture is shown, with only two levels: The hardware based Level-1 and the High-Level Trigger, resulting from merging of the L2 and EF. The black slim arrows represent the events selected, the blue wide arrows represent the data transfer. Each level reduces the data rate passed to the next level.](image)

The ATLAS trigger system was originally designed with three sequential processing levels, as shown in figure 1. In each level an event selection is performed, reducing the event rate and, consequently, the data volume sent to the following one, that in its turn, applies a tighter selection using more time consuming algorithms. The trigger system has also to be highly configurable to adapt to the changing running conditions, maximizing the physics acceptance whilst keeping the output rate under control.

The first trigger level, known as Level-1, is hardware based, and uses calorimeter and muon system information to take a decision within a latency of 2.5 μs. It uses a simple and fast reconstruction, over a coarse granularity readout of the calorimeter and muon spectrometer, to find Regions of Interest (RoI), where high transverse energy ($E_T$) objects like electrons ($e$), photons ($\gamma$), muons ($\mu$) or jets are found. The rate of Level-1 accepted events reached 70 kHz (100 GB/s) during Run 1. For Run 2 the system has been improved to handle a Level-1 output rate of 100 kHz (160 GB/s) [4], nevertheless the event rate will increase by a higher factor so improved and tighter selections remain required at Level-1 in order to limit the accepted rate. To improve the selection, new calorimeter hardware and topological trigger modules were included. The former will allow event-by-event pileup subtraction, crucial to keep the thresholds low, whilst the latter will allow combination of information from different RoIs, allowing for angular and mass selections for jet triggers.

In the original design, deployed during Run 1, the second and the third trigger levels were software based, ran on different processing farms and formed the High-Level Trigger (HLT). The second level, known as Level-2, had access to the full detector granularity. It requested fragments of data for the detector regions around the objects found by Level-1, which corresponded to an average load of 10% of the complete detector data for each event. The Level-2 nodes delivered a decision within a few tens of milliseconds, leading to an output rate of 6.5 kHz (10 GB/s) [4]. They were typically configured for a rejection factor of ten. The data of the events accepted by Level-2 is then requested by the Event Builder, that assembles all the detector data fragments and only at this time does the full event become available, as shown by the blue arrows in the left side of figure 1. The third trigger level, known as Event Filter (EF), could thus perform the complete reconstruction of the event or run in Rol-based mode. The EF ran advanced algorithms similar to the ones used in the offline reprocessing of the acquired events, in order to make a final decision within few seconds and applying another rejection factor around ten, leading to a final output rate of 400 Hz to 600 Hz (640 MB/s to 960 MB/s) for offline storage.

The HLT processing levels executed chains of reconstruction feature extraction algorithms followed by signature hypothesis algorithms. The signature algorithm’s job is to compare the features produced against some configured hypothesis and accept or reject the events. When a signature is validated the event is accepted. The system was designed for early event rejection. One of the limitations of the system, identified during the Run 1, was the need to pack, transfer and unpack the Level-2 result, the repeated access to the same detector data and the repeated execution of common reconstruction steps. To avoid this for Run 2, the two physically distinct Level-2 and EF, along with the Event Builder, were merged into a single level, simply called the HLT. The HLT algorithms run henceforth in the same farm computer nodes, sharing the same resources and reducing the data transfer and processing...
time, as shown in the right side of figure 1. In the new scheme the Event Builder is executed on demand. The new architecture also provides increased flexibility, facilitating the algorithm organization and configuration, so to better adjust to the farm work load. The new HLT nodes have a few hundred milliseconds to deliver a decision, selecting about one in a hundred events for storage and further analysis. In Run 2 the recording event rate will be \( \sim 1 \text{ kHz} \), (few GB/s).

A. The HLT software.

The HLT uses C++ algorithms that usually have a set of run time configurable parameters, defined through associated python modules. In order to identify specific physical objects the algorithms are organized in sequences, known as chains, specified in the trigger menu.

During Run 1 the HLT used two different data loading schemes: the complete calorimeter data load, known as Full Scan (FS), and an RoI-based scheme, which retrieves only the calorimeter cells along the RoI path. The FS allows precise reconstructions and recovers Level-1 inefficiencies in multi-jet events but requires more computation, making it more expensive and harder to integrate in the Run 2 HLT time constraint, at least at the Level-1 output rate. Therefore, during the shut-down an effort was put on the speed-up of the reconstruction algorithms and the FS data preparation, details can be found in section IV-A. In the RoI approach the trigger processes each RoI separately, one at a time, reducing the volume of data transferred by about a factor of ten when compared to the FS, therefore significantly reducing the processing time. However the RoI-based approach had two important downsides: On one hand, as it processed only the regions around the Level-1 jets, it relies on the Level-1 to find objects, but the sliding window algorithm used has lower efficiency to identify close-by jets. On the other hand as each RoI is processed independently, any overlap between regions will be processed multiple times, which is complex to account for and correct. This paper presents an intermediate solution, the partial calorimeter readout scheme, known as Partial Scan (PS). The details will be explained in section II-C.

B. The High-Level Jet Trigger.

The jet trigger software comprises the trigger algorithms and configurations specific to the reconstruction and selection of physical jet objects. It was designed to keep an approximately constant jet event rate output in various \( p_T \) intervals.

An HLT generic jet algorithm chain is shown in detail in figure 2. The typical jet feature extraction steps are represented by the rectangular red boxes and the hypothesis algorithms by the blue ellipse. The grey arrows represent the Level-1 region of interest objects that seed and trigger the HLT chains. The FS and PS chains start with a seeding preparation step, that ensures only one FS execution per event or gathers all the Level-1 regions into one object.

The region(s) to load is(are) passed to a cell container maker algorithm, which requests the respective data from the detector buffers, in byte-stream format, and converts it into suitable C++ cell objects. This step is known as data preparation.

The cell container is then passed to the clustering algorithm, responsible to aggregate the cells into three-dimensional clusters, corresponding to calorimeter energy depositions. Presently, the ATLAS Topological Clustering [5] algorithm is used by default, due to its noise suppressing properties. This is the most time consuming task of the chain in the FS case.

The last feature extraction algorithm in this chain is the jet maker. In this step the clusters known as “jet constituents”, present in the cluster container, are grouped to form jets. ATLAS uses the \( \text{Anti-}k_T \) algorithm [6], [7] for that job. In this work the \( \text{Anti-}k_T \) algorithm was configured to reconstruct jets with a radius of 0.4 in the \( \eta, \phi \) coordinate system.

The chains end with one or more hypothesis algorithms to decide if the event is to be kept for offline storage and further processing. The decision is taken comparing the features produced, for example the number of jets above a certain \( E_T \), against configured hypotheses.

C. The Partial Calorimeter Scan

The partial calorimeter scan was first proposed to make the Level-2 as close to a full scan as possible, thus improving the Level-2 reconstruction, accounting for region overlaps in Level-1 and recovering part of the Level-1 inefficiency for identifying close-by jets. As so, it was designed to perform a faster single pass scan over the full detector, but with low activity detector regions suppressed (zero-suppressed). The partial scan is particularly interesting for the improved HLT due to its lighter event processing, that reduces the overall computational power required by the system, as can be seen in section IV-B2, thus increasing the system flexibility. To do that, the PS processes all the Level-1 RoI’s at the same time. It then requests all the detector fragments around the Level-1 jet positions, resolving any overlap that may exist between different regions.

The size of the calorimeter region (window) needed for the PS algorithm is a crucial parameter to be tuned. If the PS window size is too small, part of the jet may lie outside of the window, resulting in lower reconstructed jet energy which may reduce the trigger efficiency. On the other hand, too large a region doesn’t improve the selection but penalizes the processing time severely. In a worst case scenario, where the regions are too big or there are many RoIs, the complete calorimeter may be requested. The Level-1 threshold and efficiency also affect the PS performance. However, with a sufficiently large window, PS is expected to recover part of the close-by jet inefficiency of Level-1.

III. JET TRIGGER PHYSICS PERFORMANCE DURING RUN 1.

Jet signatures are important for top quark and QCD measurements and searches for new particles decaying into jets, see references [8], [9], [10], [11] as examples. A usual trigger performance metric is its efficiency with respect to events taken by an independent trigger 100% efficient in the relevant region. The efficiency reflects the capability of the trigger to efficiently selects events at a given \( E_T \) threshold.

During 2011 Run 1 data taking the Level-1 was fully efficient for jets with transverse energy above 50 GeV. As
Level-1 uses simpler reconstruction algorithms, over a coarse granularity, its turn-on curves are not as steep as those from the EF. This is shown in the efficiency turn-on curves of figures 3a and 3b. At the EF, despite the use of algorithms similar to the ones used offline, the jet energy scale calibration is not the same. As a result, there is still some slope in the turn-on region, though this is much steeper than at Level-1. As an example, for an EF threshold of 30 GeV the full efficiency was reached for jets with transverse energy above 55 GeV. Due to better energy resolution for high momentum jets, allied with the smaller jet footprint in this regime, the Level-1 efficiency curves becomes sharper at higher energies as shown in figure 3a. In these plots the comparison between real data and two different Monte Carlo (MC) predictions, PYTHIA [12], [13] and HERWIG [14], is shown. Both MC simulations can accurately replicate the real data efficiency curves, within a few percent.

IV. THE HLT CALORIMETER SOFTWARE IMPROVEMENT FOR RUN 2.

Algorithm execution time optimization is very important in a very time restrictive environment such as the ATLAS trigger system. Besides the possible cost reduction due to the reduction of the computation power required by the system, a reduction of the processing time means larger available time for more complicated and efficient analysis. The next section describes the results obtained from two different solutions adopted: the algorithm optimization and the data reduction from PS.

A. Calorimeter algorithms speed-up

The algorithm speed-up effort main goal was to allow FS analysis, at the HLT, for all events for signatures involving jets.

Two algorithms received special attention, the cell unpacking and the ATLAS Topological Clustering. The speed-up achieved is shown in figure 4, as a function of time in the form of software release, since the end of Run 1. The cell unpacking was optimized by adding a specialized tool for the full-scan mode. This work was done right after the end of Run 1 and the algorithm execution time was reduced by a
factor of around seven, as shown by the blue dotted line in figure 4. For the ATLAS Topological Clustering, the measured times include the steps and corrections typically used by the jet trigger chains. The trigger was configured to select events containing at least one jet with $E_T$ greater than 50 GeV at Level-1. The overall improvement was a factor above two relative to Run 1. Among the optimizations done were the data pre-fetching, to better exploit the memory access, the code inlining to reduce some function call overhead, and the introduction of more efficient data structures. During the shutdown a new event data model was introduced, that temporarily affected the processing time. These results were obtained using real data from 2012 proton-proton collisions. The data was recorded in a high luminosity run with an average of 34 interactions per bunch crossing, at a centre of mass energy of 8 TeV and a bunch spacing of 50 ns. The tests were done in an Intel® Xeon® CPU E5645 at 2.40 GHz.

Figure 4: Calorimeter Software processing performance optimization over time, since the end of Run 1. The blue dotted line represents the time taken to retrieve the calorimeter cells in the Full Scan. The red line represents the processing time for cluster reconstruction using the ATLAS Topological Clustering algorithm, from calorimeter cells readout in a Full Scan. The light blue and yellow bands represent the standard deviation.

1) Data reduction: The number of calorimeter cells readout is shown in figure 5a for the Full Scan, (black line), and the two configurations of the Partial Scan, PS$_1$ (in red) and PS$_{1.5}$ (in blue). This result shows a significant data reduction in the PS with the PS$_1$ reading out $\sim 3.5\%$ of the detector cells and the PS$_{1.5}$ $\sim 7\%$.

2) Time performance: The time necessary to format the byte-stream data read out from the calorimeter into proper C++ objects is shown in figure 5b. Formatting the whole calorimeter (FS) takes around 10 ms while PS$_1$ takes $\sim 3$ ms (30% of FS) and the PS$_{1.5}$ takes $\sim 5$ ms (50% of FS). As expected the time required to read the cells is reduced by PS but less than the reduction observed in the number of cells loaded, as consequence of the time reduction obtained for the FS cell unpacking due to the specialization in LS1.

The processing time required to cluster cells into three-dimensional objects, using the ATLAS Topological Cluster algorithm, is shown in figure 5c, in this plot the cluster calibration step is not included. The clustering over the full calorimeter (FS) takes 98 ms, while the PS$_1$ takes $\sim 6.3$ ms (6% of FS) and the PS$_{1.5}$ takes $\sim 9.7$ ms (10% of FS). Using PS the clustering processing time can be reduced almost 80 ms per event, depending on the PS window size.

3) Physics performance: Since the PS processes the regions around the Level-1 RoIs positions, only the most central jets, in each region, are expected to be well reconstructed. To avoid confusion with jets in the central region of the detector, the closest jet to each of the Level-1 jets is called a focal jet. The definition of a focal jet is important to allow a fair comparison between PS and FS.

The $E_T$ difference between the corresponding PS and FS focal jets is plotted in figure 6. The $y$-axis shows the $E_T$ difference between the PS and the FS focal jet divided by the FS focal jet $E_T$ and the $x$-axis shows the FS focal jet $E_T$ in GeV. For the FS jet above 110 GeV the $E_T$ variation is well below 0.5%, for both PS readout configurations.

### V. CONCLUSIONS.

Jets, collimated sprays of particles resulting from the hadronization of quarks and gluons, are important for a variety of physical signatures. The ATLAS jet trigger was designed to efficiently select high transverse momentum jets whilst rejecting the low $p_T$ ones, in order to keep the rate under control. At same time, it should be flexible to adapt to different requirements and data taking conditions. An efficient event selection needs optimal jet energy resolution, which requires CPU-intensive reconstruction algorithms. During the first data taking period the jet trigger demonstrated an overall excellent performance. Following a long shut-down, the second LHC data acquisition period, Run 2, is about to start at CERN with a centre of mass energy of 13 TeV, 40% higher than in the first run. The ATLAS trigger system introduced important improvements to better exploit the Run 2 potential.

Concerning the jet selection at trigger level, offline-like algorithms and a complete calorimeter full scan analysis, at the Level-1 output rate, improve the high level trigger (HLT) efficiency, particularly in multi-jet events with close-by jets.
Figure 6: Relative transverse energy difference between jets reconstructed using the Partial Scan (PS) and Full Scan (FS) calorimeter readout schemes, shown as a function of the Full Scan jet transverse energy, $E_T^{FS}$, for High Level Trigger jets matching jets identified at Level-1. Partial Scan jets are displayed for two alternative settings: the red line corresponds to reading out regions of $\eta \times \phi = 1 \times 1$ around the Level-1 jet positions and the blue line corresponds to reading out geometrical regions of $\eta \times \phi = 1.5 \times 1.5$ around the Level-1 jets. The error bars represent the standard error on the mean.

The speed up of this algorithm got special importance and significant performance improvements were achieved during the shut-down, the cell unpacking becoming seven times faster and the cell cluster processing time being reduced by a factor greater than two.

To make the system more flexible and robust to face the challenging Run 2 environment, an alternative partial calorimeter scan algorithm (PS) was developed and tested. The PS collects all the Level-1 identified regions and loads at once the detector data around these regions, removing possible overlaps, and reducing the data to process to 3.5% to 7% of the whole calorimeter. This feature makes PS a zero suppressing algorithm, filtering low activity regions and removing them from subsequent processing. With the introduction of the PS significant time improvements were observed with respect to FS, up to 10 times faster. In addition, the comparison of the jets reconstructed with PS and FS demonstrates that PS obtains very close results to FS for single jets, with agreement in the jet transverse energy within 0.5%, for the most central jet of each region.

ACKNOWLEDGEMENTS

A. T. Delgado acknowledges the support of the IDPASC doctorate network and Fundação para a Ciência e Tecnologia, Portugal, through the grant SFRH/BD/51792/2011 and the FEDER/COMPETE-QREN.

REFERENCES


