Fast Tracker: a hardware real time track finder for the ATLAS trigger system

N. Kimura

Waseda University,
3-4-1 Okubo, Shinjuku, Tokyo

E-mail: naoki.kimura@cern.ch

ABSTRACT: The Fast Tracker (FTK) is an integral part of the trigger upgrade program of the ATLAS detector at the Large Hadron Collider (LHC). As the LHC luminosity approaches its design level of $10^{34} \text{cm}^{-2}\text{s}^{-1}$, the combinatorial problem posed by charged particle tracking becomes increasingly difficult due to the swelling of multiple interactions per bunch crossing. The FTK is a highly-parallel hardware system intended to provide high-quality tracks with transverse momentum above 1 GeV in real time for online trigger system. The FTK system’s design, based on a mixture of advanced technologies, and the expected physics performance will be presented.

KEYWORDS: Trigger concepts and systems (hardware and software); Trigger algorithms

*On behalf of the ATLAS Collaboration.
1 Introduction

Selecting interesting events online in the Large Hadron Collider (LHC) experiments is a very challenging task. By the end of 2014, the LHC will run with an energy of 13 or 14 TeV and with instantaneous luminosities which could exceed $10^{34}$ cm$^{-2}$s$^{-1}$. The triggering in the ATLAS detector [1] is realized using a three-level trigger approach, in which the first value is hardware-based and the second and third stage are realized using large computing farms. It is a crucial and non-trivial task for triggering to maintain a high efficiency for events of interest while suppressing effectively the very high rates of inclusive QCD processes, which constitute the main background. At the same time the trigger system has to be robust and provide sufficient operational margins to adapt to changes in the running environment. In the current design track reconstruction can be done only at second trigger stage in limited regions of interest, with CPU requirements that could further limit the performance of this task at the highest instantaneous luminosities. Providing high quality track reconstruction over the entire detector volume for the second stage trigger decision would allow gains in efficiency and background rejection for trigger selections on tau leptons and b-hadrons. This would also help to improve the performance in the isolation of electrons and muons which is reduced due to the increase in luminosity. The Fast Track Trigger (FTK) [3] is an ongoing upgrade project aimed at providing complete tracking at the begin of HLT, using the silicon microstrip and pixel detectors. Pattern recognition and track fitting are executed in a hardware system utilizing massive parallel processing and achieve a tracking performance close to that of the global track reconstruction. Its speed and physics performance have been estimated at simulated luminosities up to $3 \times 10^{34}$ cm$^{-2}$s$^{-1}$, corresponding to 69 interactions per bunch crossing.
2 The ATLAS detector and trigger

ATLAS is one of the general-purpose detectors at the LHC. It is designed to detect the products of proton-proton, proton-Ion and Ion-Ion collisions at a center of mass energy of 14 TeV. Particles are detected by the inner tracking detector, the electromagnetic and hadronic calorimeters and the muon system. The inner tracker is inside a 2 T solenoidal magnetic field. The FTK will use information from the pixel and strip inner silicon tracker systems shown in figure 1. To select interesting physics events created by multiple p-p collisions at the 40 MHz design frequency, a three-level trigger system is used [4]. The Level-1 trigger is hardware-based and has a maximum output rate of 100 kHz (70 kHz in 2012). It provides regions of interest to the next two levels. Level-2 and the Event Filter are implemented in software and are collectively known as the High Level Trigger (HLT). The HLT runs on a farm of several thousand CPUs. Level-2 operates within regions of interest with an average output rate of about 5 kHz in 2012, while the Event Filter has access to all information in the event, with an output rate of 350 Hz to 1 kHz of events to be promptly reconstructed. Additional events are sent to permanent storage, both for calibration purposes and for delayed reconstruction.

3 FTK system

FTK is an hardware system that rapidly finds and reconstructs tracks in the inner-detector pixel and semiconductor Tracker (SCT) layers for every event that passes the Level-1 trigger. To deal with the large input data rate as well as the large number of track candidates due to the hit combinatorics at high luminosity, the FTK is highly parallel with the system segmented into 64 $\eta - \phi$ towers, each with its own pattern recognition hardware and track fitters. FTK is going to use 12 logical layers (4 pixel including insertable B-layer + 8 axial and stereo channels on 4 physical SCT layers). FTK operates in two stages. In the first stage, 8 of the 12 silicon layers are used to perform pattern recognition and do the initial track fitting. Pattern recognition is carried out by a dedicated device called the Associative Memory (AM) [5] which helps finding track candidates in coarse resolution roads. For each road containing silicon hits on all layers or all except one, the track parameters are calculated using the full resolution hits. Hit as used here refers to a particle’s energy deposition cluster centroid. Each combination of one hit per logical layer within the road is fit to determine a goodness of fit.
The tracking on the matched road is further simplified by transforming the helix parameters ($p_T$, $\eta$, $\phi$, $d_0$, $z_0$) and the quality estimator calculations into a set of scalar products of the form

$$p_i = \sum_j C_{i,j}x_j + q_i$$

where $x_j$ are the hit coordinates in each detector layer, $C_{i,j}$ and $q_i$ are precalculated terms, and $p_i$ is either a helix parameter or a term of the $\chi^2$ quality estimator. This approximation can be implemented in modern FPGAs, exploiting their parallelism, to achieve about 1 fit/ns. The obtained helix parameter resolutions are nearly as good as those obtained with an iterative fit if the approximation above is used small detector region. In FTK that region is called a sector and consists of a single silicon module in each layer, typically a few centimeters wide.

Track passing a relatively loose $\chi^2$/ndof < 6 cut are kept, where ndof is the number of degrees of freedom of the fit. If there are hits in all layers and the $\chi^2$ test have fewer than 6 non-matching hit coordinates, they are considered to be duplicate tracks and the best track is retained based on the $\chi^2$ and the numbers of pixel and SCT layers that have a hit. Tracks from the first stage pass to the second stage where they are extrapolated into the 4 logical silicon layers not used in stage 1. Nearby hits in those layers are found and the tracks are refitted using the hits in all 12 layers. Track candidates are allowed to have at most 2 missing hits, but not a missing hit in both a pixel layer and an SCT layer. After a tighter $\chi^2$/ndof < 4 cut is applied, duplicated tracks are removed using the same non-shared hit requirement as in the first stage. In this stage, tracks available for duplicate track removal are not restricted to a common road.

The AM rapidly carries out what is usually the most CPU intensive aspect of tracking by employing massive parallelism - processing hundreds of millions of roads nearly simultaneously as the silicon data pass through FTK. The road width must be optimized. If it is too narrow, the needed size of the AM and hence the cost of the system is too large. If roads are too wide, the load on the track fitters can become excessive due to the large number of fake roads and the extremely high number of hit combinations in the road. This increases both the number of roads the track fitter must process and the number of fits within the road due to the hit combinatorics. To make maximal use of the available pattern space, FTK employs a “variable resolution” feature. In each pattern, the width can be varied independently layer by layer.

Figure 2 shows a functional sketch of FTK, which is FPGA based with the exception of one specially designed custom chip for the associative memory.

The large amount of processing required to do global tracking at a 100 kHz Level-1 trigger rate at luminosities up to $3 \times 10^{34}$ cm$^{-2}$s$^{-1}$ necessitates organizing FTK as a set of independent engines, each working on a different region of the silicon tracker. There are 64 $\eta - \phi$ towers, 16 in $\phi$ by 4 in $\eta$. This segmentation can generate some inefficiency at tower boundaries that is removed by allowing an overlap in these regions. The overlap in $\phi$ covers the range of track curvature and multiple scattering, while that in $\eta$ also takes into account the size of the beam’s luminous region along the $z$ axis.
Figure 2. Functional sketch of FTK. AM is the Associative Memory, DO is the Data Organizer, FLIC is the FTK-to-Level-2 Interface Crate, HW is the Hit Warrior, ROB is the ATLAS Read Out input Buffer, ROD is a silicon detector Read Out Driver, and TF is the Track Fitter. Second Stage Fit is referred to as the Second Stage Board elsewhere in the document.

The pixel and strip data are transmitted from the RODs on fibers and received by the Data Formatters (DF). DF mezzanine cards, the FTK IM, perform cluster finding, two-dimensional for the pixel layers. Clusters consist of pixels connected side-by-side or diagonally. The clustering algorithm uses a moving window of programmable size [6]. Clusters extending beyond the window boundary are truncated.

The DFs reorganize the data into the FTK $\eta - \phi$ towers and transmit the cluster centroids to the core crates containing the pattern recognition and track fitting hardware. The hits in each logical layer are received in a core crate and sent to the Data Organizers (DO). The hits are also converted to a coarser resolution (Super-Strips or SS) that is appropriate for pattern recognition, with the SS sent to both the AM and the DO.

The Data Organizers are smart databases where full resolution hits are stored in a format that allows rapid access based on the pattern recognition road ID and then retrieved when the AM finds roads with the requisite number of hits. Dual-output HOLAs, which replace the existing HOLA output mezzanine cards in the pixel and SCT Read Out Drivers (ROD), provide FTK with an identical copy of the silicon data being sent to the DAQ Read Out System (ROS) input buffers. FTK receives the hits at full rate as they are sent from the RODs following a level-1 trigger. After processing, FTK fills ROSs with the helix parameters and hits for all tracks with $p_T$ above 1 GeV.

The AM boards contain a very large number of preloaded patterns corresponding to the possible combinations of a Super-Strip in each of 8 silicon layers for real tracks. The patterns are determined in advance from a full ATLAS simulation of single tracks. The AM is a massively parallel system in that all patterns see each silicon hit nearly simultaneously. As a result, pattern recognition in FTK is completed shortly after the last hit has been transmitted from the silicon RODs. When a pattern has 7 or 8 layers hit (such patterns that contain track candidates are referred to as roads), the AM sends the road ID number back to the Data Organizer which fetches the associated full resolution hits and sends them and the road number to the Track Fitter (TF). The TF has access to a set of constants for each detector sector. Because a sector is quite narrow, the TF can provide high resolution helix parameters using a simple calculation that is linear in the position of the hit in each layer. Fitting a track is thus extremely fast since it consists of a series of integer
multiply-and-accumulate steps. In a modern FPGA, approximately $10^9$ track candidates can be fit per second. Following fitting, duplicate track removal (the Hit Warrior or HW function) is carried out among those 8-layer tracks in a road that pass the $\chi^2$ cut.

The Second Stage Boards (SSB) receive from the Data Formatters the cluster centroids in the 4 layers not used in stage 1. When a track passes the stage-1 criteria, the road number and hits are sent to the SSB. The track is extrapolated into the 4 additional layers, nearby hits are found, and a full 12-layer fit is carried out. A minimum of 3 hits in the 4 layers is required. Duplicate track removal is again applied to the tracks that pass the $\chi^2$ test, but now tracks in all roads are used in the comparison. SSB output tracks consisting of the hits on the track, the $\chi^2$ the helix parameters, and a track quality word that includes the layers with a hit are sent to the FTK-to-Level2 Interface Crate (FLIC). The FLIC organizes the tracks and sends them to the High Level Trigger ROSs using the standard ATLAS protocols, and carries out monitoring functions. The FLIC crate is organized so that in the future global event functions can be carried out in other cards in the crate.

4 FTK expected performance

This section describes the utility of the FTK as a toolbox for the HLT by detailing the performance on both single particles and multi-track objects using MC simulation without insertable B-layer. Each of the FTK components has been simulated in software to tune the system and evaluate its expected performance. FTK tracks are compared with tracks reconstructed using the ATLAS offline tracking algorithm.

4.1 Single particle performance

Single muon and pion MC samples were simulated with a flat distribution in the particle helix parameters: $|\eta| < 2.5$, $0 < \phi < 2\pi$, $|d_0| < 2.2$ mm, $|z_0| < 120$ mm and curvature, $1/(2p_T) < 1/(2 \times 800)$ MeV$^{-1}$. Although pile-up is absent, offline tracks are selected with the standard “high-pile-up” cuts:

- Number of pixel plus SCT hits on the track plus the number of inactive modules crossed by the track must be at least nine.
- There must be no pixel holes on the track.

In addition to the hit requirements above, the following phase-space cuts are applied to the muons to ensure that the tracks are within the FTK training phase-space. These cuts are also applied to the offline tracks when taking the efficiency of the FTK tracks with respect to the offline tracks:

- The impact parameter, $|d_0|$, must be less than 2 mm.
- $|z_0|$ must be less than 110 mm.

Truth particles are matched to FTK tracks based on the fraction of hits on the track which were produced by the particle. Tracks with at least 50% of the hits coming from a single truth particle are considered matched. Figure 3 shows the track-finding efficiency for FTK tracks with respect to tracks found by the offline tracking algorithm. The offline tracks are required to satisfy the cuts
listed. Here the relative inefficiency between muons and pions is reduced to less than 1% because the nuclear interaction probability affects offline and FTK tracking similarly. The inefficiency with respect to offline which remains is due to the geometric coverage of the patterns.

Resolutions on the track parameters were also studied as a function of the curvature of the muon; results are shown in figure 4, compared with those obtained using offline tracks. The resolution is defined as the width of a gaussian fit in a range of 3 times the RMS of the distribution. The same definition is used for the offline tracks to facilitate an accurate comparison. FTK and offline have similar resolutions. At small values of curvature the FTK hit resolution and fit precision contribute to a decreased relative resolution. The longitudinal quantities such as $\eta$ and $z_0$ show a worse resolution with respect to offline even at large curvature. The FTK $z_0$ resolution is three times that of the offline tracks at small curvature.

### 4.2 b-jet tagging

New physics that couples to heavy fermions may be rich in final-state b quarks, but not necessarily other easily identifiable objects which can be used in the trigger. Given the large QCD production of light quarks and gluons, it is important that the ATLAS trigger efficiently selects jets from b-quarks while providing a large rejection factor against other jets. This is a challenging task for the current Level-2 trigger because purely jet-based triggers either have very high thresholds for single jets, or have many jets in multi-jet triggers. The single jet thresholds are typically too high for b-tagging to be useful, and the multi-jet triggers produce too many ROIs (Region Of Interest) for the time and
CPU constraints at Level-2. The availability of FTK tracks following a Level-1 trigger removes the constraints on the number ROIs to be considered for b-tagging and allows the use of sophisticated algorithms to select b-quark jets at Level-2. The transverse impact parameter distribution is the basic input to b-tagging. This distribution is sensitive to the long lifetime of particles within the b-jet and provides a means of separating b-jet from light-flavor jets. The transverse impact parameter of tracks associated to light-flavor and heavy-flavor jets is shown in figure 5(a) for the both the FTK and the offline tracks in the $\mu = 69$ samples. These are pile-up samples intended to reproduce different instantaneous luminosity conditions. The impact parameter distribution is signed such that track displacements in the direction of the jet are given positive values, while tracks with displacements opposite of the jet direction are given negative values. A longer positive tail is present for the tracks associated to heavy-flavor jets. Figure 5(b) shows the FTK efficiency relative to offline vs the offline IP2D efficiency for the three FTK working points corresponding to different light-jet rejection in the barrel region ($|\eta| < 1.1$). IP2D [2] is a likelihood-based tagger utilizing the transverse impact parameter significance and is robust to pile-up effects. As can be seen in the figures, with offline operating points slightly below the online FTK operating point, good online rejection can be achieved with little loss in performance.

4.3 Primary vertex finding

Primary vertex finding is an important part of the physics analyses. There are many pile-up dependent corrections that make use of the number of primary vertices in offline analyses, and an accurate
d0 [mm]
-2 -1.5 -1 -0.5 0 0.5 1 1.5 2
Normalized Entries
-5 10
-4 10
-3 10
-2 10
-1 10
1
ATLAS Simulation, no IBL | < 1.1)
η Barrel (| η > = 69 µ <
Offline Light-Flavor
Offline b-Jet
FTK Light-Flavor
FTK b-Jet

(a)

(b)

Figure 5. (a) Transverse impact parameter of tracks associated to light-flavor (black) and heavy-flavor (red) jets. The solid lines shows the distribution for the offline tracks, whereas the points shows the FTK tracks. (b) Efficiency of the FTK b-tagging algorithms as a function of the IP2D offline selection efficiency in the barrel region for 3 different FTK working points [3].

Figure 6. (a) Number of primary vertices reconstructed using FTK tracks vs the number of in-time pile-up interactions (Red) and the number of vertices reconstructed by offline tracking (Blue). (b) Uncertainty in z-direction as reported by the vertex fit of the hard scatter FTK primary vertex in < µ > = 46 (Red) and < µ > = 69 (Black) [3].

determination of the hard scatter interaction position allows for track selection for a variety of applications. Because primary vertex finding necessitates full detector tracking and is therefore very CPU intensive, it is not possible to determine the primary vertex in standard trigger processing. Since FTK performs full scan tracking in tens of µs, it can provide the primary vertex information. The algorithm used in this study is the ATLAS HLT beam spot finding and primary vertex reconstruction algorithm [7] adapted for FTK tracks. Figure 6(a) shows the number of primary vertices found by both FTK and offline as a function of the number of interactions in the event. The performance shows a linear correlation with the number of interactions for both FTK and offline. The hard scatter vertex is well reconstructed with a σz as reported by the vertex fit of 56 µm with and uncertainty of 19 µm in the transverse position. The σz distribution is shown in figure 6(b).

4.4 Tau tagging

The τ is the heaviest known lepton and plays a special role in Standard Model processes such as Higgs boson decays. Hadronically decaying τs typically decay in to one or three charged hadrons,
which are call 1-prong and 3-prong decays, respectively. Therefore the number of track information of $\tau$ candidate is good value to select $\tau$ from huge QCD jet. The current Level-2 selection first imposes calorimetric cuts in order to reduce the ROI rate to a level acceptable for the tracking algorithms, then tracking is performed on the remaining ROIs. Figure 7 shows the distribution of the number of the tracks of $\tau$ candidate of offline, current Level-2 tracking in limited ROI (TauB) and FTK. Two clear peaks are visible for the 1-prong and 3 prong decays.

5 Conclusion

The FTK systems design was described and its expected performance was presented. The design of individual boards is in progress. In particular some of the dual output HOLA cards are already installed. Other boards are already at prototype level, and under the test. The FTK is expected to play a significant role in the ATLAS trigger system. Simulations of the system shows that it performs well with up to $\langle \mu \rangle = 69$ pile-up events.

References

[1] ATLAS collaboration, G. Aad et al., The ATLAS Experiment at the CERN Large Hadron Collider, 2008 JINST 3 S08003.
[2] ATLAS collaboration, b-Tagging Algorithms and their Performance at ATLAS, arXiv:0809.4896.
[3] ATLAS collaboration, Fast TracKer (FTK) Technical Design Report, CERN-LHCC-2013-007 ATLAS-TDR-021.
[4] ATLAS collaboration, Expected Performance of the ATLAS Experiment - Detector, Trigger and Physics, arXiv:0901.0512.
[5] F. Morsani, S.R. Amendolia, S. Galeotti, D. Passuello, L. Ristori et al., The AMchip: A VLSI associative memory for track finding, Nucl. Instrum. Meth. A 315 (1992) 446.
[6] ATLAS collaboration, A Multi-Core FPGA-based Clustering Algorithm for Real-Time Image Processing, ATL-DAQ-PROC-2013-037 (2013).
[7] ATLAS collaboration, Characterization of Interaction-Point Beam Parameters Using the pp Event-Vertex Distribution Reconstructed in the ATLAS Detector at the LHC, ATLAS-CONF-2010-027 (2010).