PROCEEDINGS OF SCIENCE

Vertex Reconstruction and Tracking in the Trigger Algorithm for Atlas

Schiavi Carlo[*]
Università degli Studi di Genova and INFN Genova
E-mail: Carlo.Schiavi@ge.infn.it

on behalf of the ATLAS Inner Detector Trigger Group

Online track and vertex reconstruction are fundamental ingredients for event selection at Large Hadron Collider (LHC) experiments. In the ATLAS experiment the first reconstruction stage where this goal will be achievable is the software-based Second Level Trigger (LVL2).

In this contribution we present algorithms for fast reconstruction of charged tracks and of secondary vertexes in the framework of the ATLAS High Level Trigger (HLT). Trigger tools for track fitting and extrapolation are also described.

Examples of performance evaluation methods adopted to characterise the presented algorithms are given, along with results obtained on simulated and real data. Efficient and robust tracking capabilities are demonstrated to be achievable with average execution times in agreement with the trigger requirements.

The 16th International Workshop on Vertex detectors
September 23-28 2007
Lake Placid, NY, USA

[*] Speaker.

PoS(Vertex 2007)031

1. Introduction

The ATLAS (A Toroidal LHC ApparatuS) experiment [1] will start taking data during summer 2008 at the Large Hadron Collider (LHC), a pp collider currently being prepared for operation at the European Organisation for Nuclear Research (CERN).

At the LHC, protons will collide at a centre of mass energy of 14 TeV, with a design luminosity of $10^{34}\,\mathrm{cm}^{-2}\,\mathrm{s}^{-1}$. At this luminosity, $\sim 25\,pp$ interactions will occur at every bunch crossing, in addition to any interesting physics events. Together with the bunch spacing of 25 ns, this poses stringent design demands on both the detectors and the Trigger and Data Acquisition (TDAQ) system [2].

As the region around the pp interaction point will be characterised by a very high density of charged tracks, high granularity detectors granting low occupancy are required. The ATLAS tracking system, the so called Inner Detector (ID) [3], comprises the Pixel detector, the SemiConductor Tracker (SCT) silicon strips and the Transition Radiation Tracker (TRT) straw tubes.

The large number of detector channels, most of which are due to the tracking detectors, leads to an average event size of $\sim$ 1.5 MB. This means that the TDAQ has to face a very challenging networking task, while at the same time it has to limit the final event storage rate to a maximum value of $\sim$ 200 Hz. This last requirement must be fulfilled through the event rejection performed by the trigger system.

1.1 The ATLAS trigger system

The required reduction of the 40 MHz interaction rate down to the 200 Hz maximum event data storage rate will be provided by three subsequent trigger selection stages, each refining the event processing and selection performed by the previous ones.

The hardware-based First Level trigger (LVL1) performs a first rejection step using only reduced granularity data coming from the calorimeters and the muon detectors; it operates within a 2 μs latency, producing an average output rate of 75 kHz, limited to a maximum of 100 kHz.

Further event selection is then performed by software tools running on dedicated commercial processor farms and is divided in two layers, the Second Level trigger (LVL2) and the Event Filter (EF), collectively referenced as HLT.

Reconstruction at LVL2, seeded by information collected at LVL1, can exploit full granularity information from all ATLAS sub detectors; it is performed processing data contained inside one or more geometrical regions identified at LVL1, the so called Regions of Interest (RoI), each corresponding on average to 2% of the total event.

The LVL2 decision must be taken, for each event, with an average processing time of 40 ms on a 2 GHz quad core machine, greatly constraining the LVL2 reconstruction algorithms.

The EF has much looser time constraints, of the order of 2 s, so more complex reconstruction algorithms and strategies can be adopted at this stage.

1.2 Tracking and vertexing in the trigger system

The earliest stage where tracking information can be used in the trigger selections is the LVL2, the first software selection layer.

The tight constraints on the execution times force algorithm development to a very delicate balance between time consumption and the performance in terms of efficiency, track parameter resolution

and fraction of fake track candidates.
On its turn, the vertexing capability at LVL2 is mainly constrained by the quality of the reconstructed tracks. Furthermore, the amount of iterative procedures adopted inside the vertexing algorithms must be kept to a minimum, to optimise the execution time.
Given the more relaxed timing requirements, the approach adopted for the EF tracking is based on the idea of reusing as much as possible the code developed for off-line reconstruction. To cope with the limited time budget available, some adaptations are mandatory. First of all, off-line code is executed within wrapper algorithms, which can access only data contained in the RoIs passing LVL2. Some additional tuning of algorithm parameters can also be performed, as an example reducing the number of iterations to save computing power or skipping part of the reconstruction process, exploiting the high modularity of the off-line tracking code.

2. Tracking and vertexing algorithms in the LVL2 trigger

In this section an overview of the tracking and vertexing algorithms currently available for execution in the LVL2 trigger is given. An exhaustive description of the technical details of the off-line algorithms adapted for EF operation can be instead found in [4].

2.1 The SiTrack silicon tracking algorithm

The SiTrack algorithm adopts a combinatorial pattern recognition approach to reconstruct tracks starting from the space points (SP) formed in the ID silicon detectors.
In order to combine them, the SP are first of all grouped into sets from which the entries of each combination will be extracted; the grouping is implemented using the idea of “logical layers”. Each of these corresponds to a list of physical detector layers and are labelled with increasing numbers moving away from the beam line. The same physical layer can be included in more logical layers, to increase the robustness of the track finding process. As an example, the default first logical layer includes the innermost two pixel layers.
The track reconstruction algorithm proceeds through the following five steps: track seeds formation; optional primary vertex reconstruction; track seeds extension; extended seeds merging; clone removal.
The formation of track seeds corresponds to a combinatorial pairing of SP coming from the innermost two logical layers. For each seed, the extrapolation to the beam line is evaluated, using a straight line approximation; this process is depicted in Fig. 1. A cut on the extrapolated minimum distance from the beam line is applied, reducing the number of seeds to be further processed. This selection fixes the values of the minimum momentum and of the maximum tranverse impact parameter which can be reconstructed with SiTrack; so, this cut is choosen large enough to pose no biases on the physics process to be selected and on secondary vertex reconstruction.
The subsequent step is the reconstruction of the position of the primary interaction vertex along the beam line, used to reject tracks not coming from the primary interaction. The vertex reconstruction is performed filling an histogram with the longitudinal impact parameter of the seeds and searching for histogram maxima; more than one vertex candidate can be retained and seeds not pointing to any of the reconstructed vertexes are discarded. This step is optional and, as an example, is used for jet reconstruction, while it is skipped in case of low multiplicity event topologies, e.g. for the

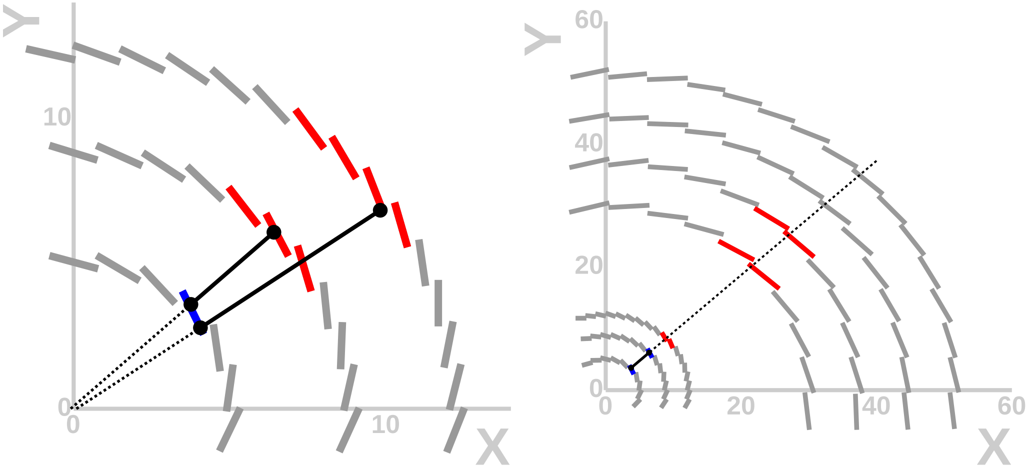

Figure 1: Pictorial scheme of the SiTrack combinatorial strategy for track seeds formation (left) and track seeds extension (right).

reconstruction of single isolated leptons.

Each retained seed is extended, as depicted in Fig. 1, extrapolating it to the outer logical layers and forming one or more SP triplets for each seed; extensions are selected applying a cut on the distance between the outer SP and the extrapolated seed. Each extended seed is then fit with a straight line in the longitudinal plane and parametrised as a circle in the transverse plane.

At this point, all the extensions found for each seed must be merged into a single full track, grouping the triplets having similar track parameters after the fit. The full track is thus defined as formed by the union of the SP from all the merged extensions. All the triplets not involved in the merging process are discarded, while track parameters are crudely estimated for the full track.

Two full tracks obtained from different track seeds may still share most of their SP; these tracks are defined clones. To eliminate these ambiguous cases, only the clone track containing the largest number of SP or having the lowest χ^2 value is retained. The full tracks are finally refit using one of the available common fit tools described in the following.

2.2 The IDScan silicon tracking algorithm

The IDScan algorithm follows an alternative pattern recognition strategy, based on space point histogramming. Reconstruction of the position of the primary vertex along the beam line is performed similarly to what is done in the SiTrack algorithm; this is a mandatory step for IDScan, since it provides a reference point for subsequent histogramming.

All the space points contained in the processed RoI are then arranged into a two-dimensional histogram in (η,ϕ). The content of bins with hits in at least four different detector layers is accepted, while all the other space points are rejected. A preliminary estimate of p_T and ϕ_0 is provided for all the possible SP triplets in each retained bin.

The surviving SP are put in another two-dimensional histogram in $(1/p_T,\phi_0)$ and bins containing hits in at least four different detector layers are taken as candidate tracks.

After this stage, clone tracks removal is performed and the candidate tracks undergo a full fit.

2.3 The common track fit tools

A suite of fit tools is available to evaluate the parameters of the reconstructed track candidates. All the tools share a common interface, so each track reconstruction algorithm can switch between them via configuration.
Currently, three different methods have been implemented: a distributed Kalman filter performing a full track fit, a distributed Kalman filter performing a perigee parameter estimation and a fit based on the Karimaki circle fit method.
The first uses the non-uniform ATLAS magnetic field map and performs track extrapolation between a set of surfaces, using a parabolic approximation. The other two methods use instead simple trajectory parametrisations, respectively approximating the tracks with a three-dimensional helix or decomposing their projections in terms of a straight line and a circle.
The first fit obviously gives the better track parameter estimation, comparable with the one obtained by off-line tools. At the same time, the full fit consumes a larger amount of processing time with respect to the other methods: on a 2.4 GHz Xeon CPU it takes $\sim$0.35 ms to perform a full fit of a single track in the silicon detectors, while each of the two other methods are executed in less than 0.1 ms.

2.4 The TRT extrapolation tool

Tracks identified in the Silicon detectors can be propagated into the TRT, to improve track parameter resolution and for particle identification purposes. A track following TRT extension tool, based on the Probabilistic Data Association Filter [5] technique is available; the choice of this method was driven by the requirement of having a fast and robust algorithm, capable of coping with the high occupancy of the detector (up to 50%). The tool is implemented as a single-pass recursive algorithm, computationally similar to the Kalman filter, with almost linear execution time. A single track is followed, associating to it at most one additional hit per layer, while all the other hits are rejected as background.
From the timing point of view, the extrapolation of a single track into the TRT requires $\sim$3 ms on a 2.4 GHz Xeon CPU.

2.5 The vertexing algorithm

Secondary vertex reconstruction at LVL2 is performed using a fast iterative algorithm based on a geometrical vertex fit.
This method doesn't use mass constraints and is equivalent to Billoir's full vertex fit [6]. It uses a linear transformation of track parameters which reduces the covariance matrix to a block-diagonal form, thus enabling to implement a Kalman filter with reduced-size measurement model, without need for time-consuming matrix inversions.
The algorithm provides the estimated three-dimensional vertex position, the track parameters at vertex and the full covariance matrix for the associated tracks; this enables to evaluate the invariant mass variance using the full covariance matrix.
This tool is as an example adopted for the online selection of $D_s \rightarrow \phi(KK)\pi$ decays, where, on simulated samples corresponding to a luminosity of $10^{33} cm^{-2} s^{-1}$, it reduces the background rate from 5 kHz (after LVL2 confirmation of the LVL1 triggering muon) to 200 Hz.

3. Applications and performance

In this section the role of track and vertex reconstruction in the trigger selections developed for the ATLAS experiment is described.
An overview of the performance of the above presented algorithms is then given on simulated and real data and in terms of their execution time.

3.1 Applications in trigger selections

The tracking information produced by the trigger tracking algorithms is used in the definition of the most crucial trigger items for the ATLAS experiment:

- selection of high-momentum isolated electrons and muons: tracks reconstructed in the ID are used to match information from the calorimeters and the muon detector; this is important for the rejection of fake candidates, to improve the momentum measurement and for particle identification purposes;
- reconstruction of tracks from tau decays: tracks are first of all used to match information from outer detectors; cuts on track multiplicity and track-based isolation criteria are also applied;
- tagging of b-jets: the impact parameters of the reconstructed tracks are used to evaluate the discriminant variables adopted for flavour tagging; development plans include the introduction of additional discriminant variables based on secondary vertex reconstruction, as the corresponding invariant mass;
- reconstruction of exclusive decays relevant for B physics: secondary vertex reconstruction and the consequent invariant mass calculations based on associated tracks are a fundamental ingredient for these selections.

3.2 Performance on simulated data

The performance of tracking and vertexing algorithms is constantly checked and monitored on simulated samples. In this context, efficiency and fraction of fake tracks can be easily obtained comparing the reconstructed tracks with the Monte Carlo information. In particular, reconstructed tracks are associated to simulated tracks exploiting the MC information on the detector hits by which they are built. As an example of a tracking performance study, here are summarised the results obtained from the analysis of $H \rightarrow b\bar{b}$ events, which are the standard signal sample adopted for single-jet b-tagging performance studies. This sample, containing no superimposed minimum bias pp interactions, have been reconstructed using the SiTrack and IDScan algorithms; the reported efficiency is normalised to the one obtained after LVL1.
In Fig. 2 the reconstruction efficiency as a function of the transverse momentum and the pseudorapidity η of the tracks is shown for the two algorithms. It can be noticed that the efficiency curves for the two algorithms are almost compatible over the acceptance of the tracking detectors and for the p_T values shown. This is also evident from table 1, where the overall tracking efficiency is reported along with the fraction of fake tracks.

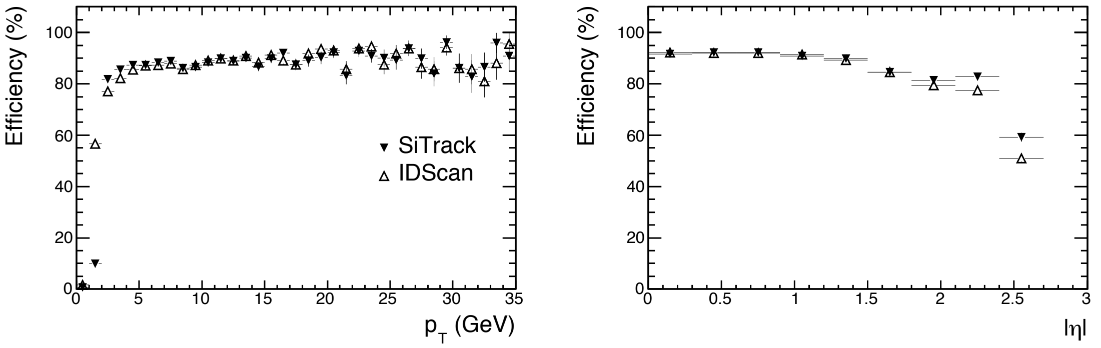

Figure 2: Track reconstruction efficiency in jets, relative to LVL1 selection, as a function of transverse momentum and pseudorapidity η for SiTrack (full triangle), IDScan (open triangle); the η plot refers to tracks above 3 GeV.

	SiTrack	IDScan
Efficiency (%)	86.7 ± 1.1	84.7 ± 1.1
Fake fraction (%)	3.7 ± 0.2	1.7 ± 0.1

Table 1: Reconstruction efficiency and fake fraction for the SiTrack and IDScan algorithms, evaluated on a simulated $H \to b\bar{b}$ jet sample over $|\eta| < 2.5$ and for tracks above 3 GeV.

3.3 Performance on real data

During June 2006 a surface cosmic run was performed; during this test, a fraction of the already assembled SCT and TRT barrel sectors were operated, while the LVL1 trigger was provided by a logic based on scintillators.

This type of studies, apart from gathering useful track samples for the detector alignment procedures, is particularly meaningful as it enables to verify the performance of tracking algorithms in an environment which is far from ideal. First of all, the tracks to be reconstructed are characterised by a large impact parameter with respect to the nominal beam line position. Furthermore, tracks had to be reconstructed without knowing the alignment constants to be applied to the detector geometry.

In Fig. 3 the tracking efficiency of the SiTrack algorithm with respect to reconstructed off-line tracks is shown as a function of the track impact parameters; the tracking efficiency, measured over different runs, always exceeded 90%.

The most encouraging result is that this satisfactory efficiency value was achieved without any code adaptation for the SiTrack algorithm. To cope with misaligned SP and large impact parameters, the algorithm was simply configured with a looser set of cuts for the track seeds formation and extrapolation and including more physical detector layers into the definition of each logical layer.

On the contrary, as the IDScan algorithm relies on the reconstruction of the primary vertex along the beam line, it was not possible to run it directly on the recorded data. An additional algorithm was thus introduced and executed before IDScan for each RoI; the aim of this algorithm is to shift the coordinate frame of the SP so that the cosmic tracks appear to point to the beam line and IDScan

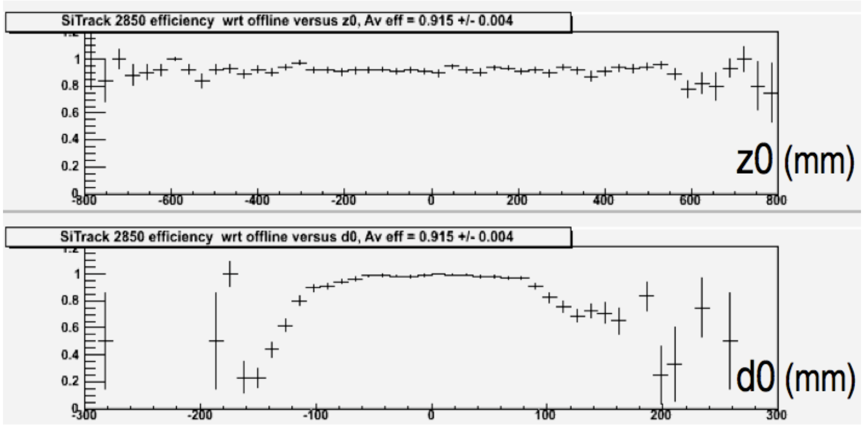

Figure 3: Cosmic tracks reconstruction efficiency of the SiTrack algorithm, as a function of the impact parameters with respect to the nominal beam line: along the beam direction (z_0) and in the transverse plane (d_0).

can run in an operation mode similar to the one foreseen for physics data taking. Anyway, despite this additional step in the reconstruction sequence, the overall efficiency for the reconstruction with IDScan resulted lower than the one obtained with SiTrack, most probably due to inefficiencies of the SP shifting rather than to the IDScan algorithm itself.

3.4 Timing performance

One of the most relevant issues for any trigger algorithm, especially at LVL2, is its timing performance. This is the reason why a lot of effort is put into code optimisation making execution times for all the steps of track reconstruction and vertexing constantly evolve towards lower values. The current status for the LVL2 trigger ingredients is summarised in table 2, where the execution time per RoI is given on electron, QCD background and top events. All the measurements were done on a 2.4 GHz Xeon CPU. It can be noticed that the main contribution to the overall timing is given by the data preparation, i.e. by the translation of raw detector data into high level object such as space points in the silicon detectors or drift circles in the TRT. No distinction was made between the two available tracking algorithms as their timing performance is practically the same.

Event type	electron	QCD	top
Data preparation for Pixel+SCT (ms)	2	3	3
Track reconstruction (ms)	0.5	0.7	1.0
Data preparation for TRT (ms)	2.0	2.6	4.0

Table 2: Timing for the LVL2 tracking ingredients, measured on different types of simulated events.

The overall timing for a single RoI, evaluated in the worst case and taking into account also the track fit and extrapolation, amounts to ∼15.5 ms. Considered that the expected average number of RoIs per event is ∼1.5, the overall LVL2 reconstruction timing fits well within the trigger requirements.

Similar conclusions can be drawn from the corresponding numbers obtained for EF algorithms, reported in table 3.

Event type	electron	QCD	top
Data preparation for Pixel+SCT (ms)	15	17	15
Track reconstruction (ms)	13	165	130
Data preparation for TRT (ms)	14	15	11
Track extension to the TRT (ms)	9	22	22

Table 3: Timing for the EF tracking ingredients, measured on different types of simulated events.

4. Conclusions

The general principles on which the ATLAS online track and vertex reconstruction are based were discussed and their many applications to trigger selections were summarised.

A detailed description of the algorithms currently available for the second level trigger was given and their performance was characterised on simulated samples and on data from surface cosmic runs. In both cases the tracking efficiency was demonstrated to be satisfactory, while the execution time was within the trigger specifications.

Furthermore, the first studies on real data demonstrate that the LVL2 tracking can be made robust enough to cope with non-ideal conditions, as the ones expected for first collisions. Along this direction, further development is ongoing to ensure that the algorithms presented here will have the required robustness and flexibility to operate in the early phases of data taking as the LHC will start its operation in 2008. In fact they will provide a significant contribution to the commissioning and calibration of the ATLAS detectors and to the optimisation of the HLT selection strategies towards the completion of the ATLAS physics program.

References

[1] ATLAS Collaboration, *ATLAS Detector and Physics Performance Technical Design Report*, CERN/LHCC/99-14, ATLAS-TDR-014 (1999).

[2] ATLAS HLT/DAQ/DCS Group, *ATLAS High-Level Trigger, Data Acquisition and Controls Technical Design Report*, CERN/LHCC/2003-022, ATLAS-TDR-016 (2003).

[3] ATLAS Inner Detector Community, *Inner Detector Technical Design Report*, CERN/LHCC/97-16, ATLAS-TDR-4 (1997).

[4] Cornelissen, T. *et al*; *Concepts, Design and Implementation of the ATLAS New Tracking*, CERN-ATL-COM-SOFT-2007-002 (2007).

[5] D. Emeliyanov; Nucl. Inst. Meth. **A566** (2006) 50.

[6] P. Billoir and S. Qian; Nucl. Inst. Meth. **A311** (1992) 139; Nucl. Inst. Meth. **A350** (1994) 624.