Vertex Reconstruction and Tracking at Trigger Level for ATLAS

Carlo Schiavi

on behalf of the ATLAS Inner Detector Trigger Group

Vertex 2007

Outline

Tracking and Vertexing in the ATLAS Trigger Scheme

Tracking and Vertexing at Level2

Tracking and Vertexing at the Event Filter

Conclusions

Outline

Outline

Tracking and Vertexing in the ATLAS Trigger Scheme

Tracking and Vertexing at Level2

Tracking and Vertexing at the Event Filter

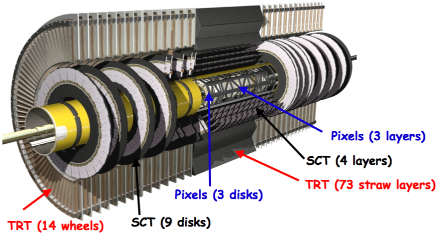
Conclusions

Outline

The ATLAS Tracking System

The ATLAS experiment is equipped with the following tracking subsystems:

- Magnetic field:
  solenoidal magnet; 2T field.
- Pixel detector:
  $50 \times 400\ \mu m^2$ sensors;
  max radius: 15 cm.
- SemiConductor Tracker (SCT):
  80 μm strips;
  max radius: 50 cm.
- Transition Radiation Tracker (TRT):
  tubes with two thresholds;
  max radius: 1 m.



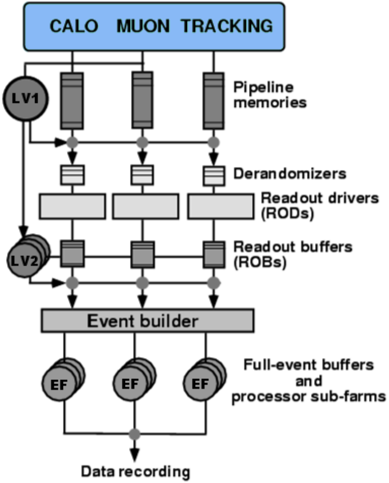
The ATLAS Trigger

LHC interaction rate ($\sim$ 1 GHz) is reduced through three trigger selection steps:

Level1 Trigger:
- hardware based (FPGAs ASICs);
- coarse granularity calo/muon data;
- latency: 2 μs;
- output rate: 75 kHz.

HIGH LEVEL TRIGGER (HLT):
Level2 Trigger:
- detector sub-regions processed;
- full granularity for all subdetectors;
- fast rejection "steering";
- mean execution time: $\sim$ 10 ms;
- output rate: $\sim$ 2 kHz.

Event Filter:
- guided by Level2 result;
- potential full event access;
- off-line quality algorithms;
- mean execution time: 2 s;
- output rate: $\sim$ 200 Hz;
- data storage: $\sim$ 300 MB/s.



The ATLAS Trigger

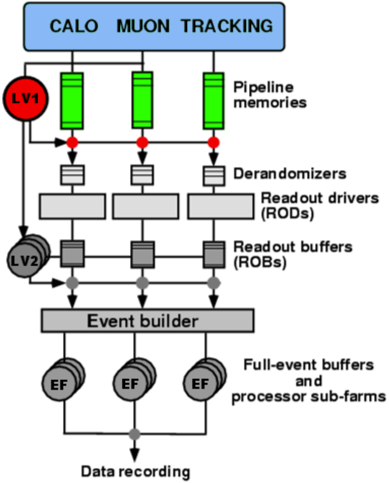
LHC interaction rate (~ 1 GHz) is reduced through three trigger selection steps:

Level1 Trigger:
- hardware based (FPGAs ASICs);
- coarse granularity calo/muon data;
- latency: 2 μs;
- output rate: 75 kHz.

HIGH LEVEL TRIGGER (HLT):

Level2 Trigger:
- detector sub-regions processed;
- full granularity for all subdetectors;
- fast rejection "steering";
- mean execution time: $\sim$ 10 ms;
- output rate: $\sim$ 2 kHz.

Event Filter:
- guided by Level2 result;
- potential full event access;
- off-line quality algorithms;
- mean execution time: 2 s;
- output rate: $\sim$ 200 Hz;
- data storage: $\sim$ 300 MB/s.



The ATLAS Trigger

LHC interaction rate (~ 1 GHz) is reduced through three trigger selection steps:

Level1 Trigger:

- hardware based (FPGAs ASICs);
- coarse granularity calo/muon data;
- latency: $2\ \mu s$;
- output rate: 75 kHz.

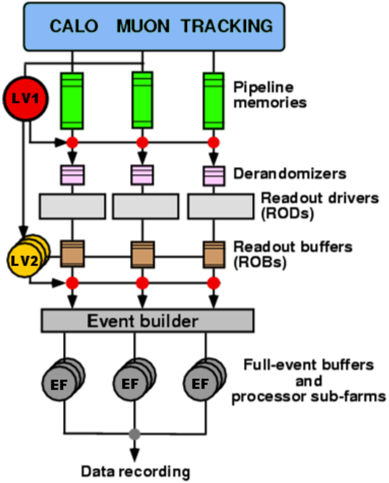
HIGH LEVEL TRIGGER (HLT):

Level2 Trigger:

- detector sub-regions processed;
- full granularity for all subdetectors;
- fast rejection "steering";
- mean execution time: $\sim$ 10 ms;
- output rate: $\sim$ 2 kHz.

Event Filter:

- guided by Level2 result;
- potential full event access;
- off-line quality algorithms;
- mean execution time: 2 s;
- output rate: $\sim$ 200 Hz;
- data storage: $\sim$ 300 MB/s.



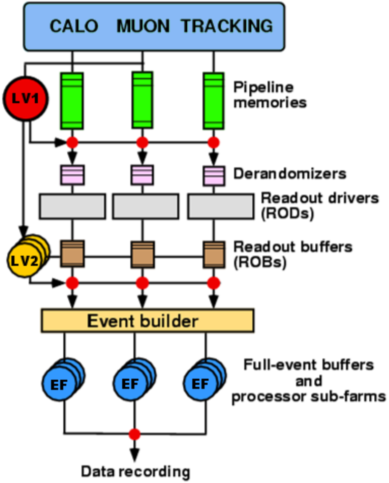
The ATLAS Trigger

LHC interaction rate (~ 1 GHz) is reduced through three trigger selection steps:

Level1 Trigger:
- hardware based (FPGAs ASICs);
- coarse granularity calo/muon data;
- latency: $2\ \mu s$;
- output rate: 75 kHz.

HIGH LEVEL TRIGGER (HLT):
Level2 Trigger:
- detector sub-regions processed;
- full granularity for all subdetectors;
- fast rejection "steering";
- mean execution time: $\sim$ 10 ms;
- output rate: ~ 2 kHz.

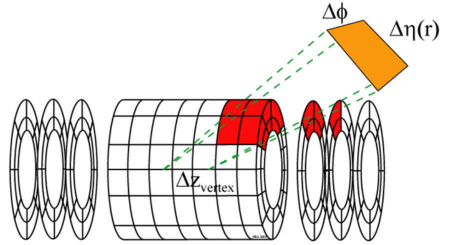
Event Filter:
- guided by Level2 result;
- potential full event access;
- off-line quality algorithms;
- mean execution time: 2 s;
- output rate: ~ 200 Hz;
- data storage: ~ 300 MB/s.



The ATLAS Trigger Data Access

One of the distinguishing features of the ATLAS Software Selection strategy is the reconstruction in Regions of Interest (RoI):

- only a subset of detector data is processed;
- RoI size and position are derived from previous steps;
- mainly exploited at LEVEL2;
- parallel processing;
- goal: minimize processing time and network traffic.



LVL2 Tracking Algorithms

The earliest stage where tracking information can be used in the trigger selections is the LVL2.
Given the tight constraints on the execution times, this forces algorithm development to a very delicate balance between time consumption and the performance in terms of efficiency, track parameter resolution and fraction of reconstructed fake tracks.

LVL2 tracking is used for the definition of the following trigger items:

- selection of high-p_T electrons and muons: tracks reconstructed in the Inner Detector are used to match information from the calorimeters and the muon detector;
- reconstruction of tracks from tau decays: tracks are used for both matching information form outer detectors and to apply cuts on track multiplicity;
- tagging of b-jets: the impact parameters of the reconstructed tracks are used to evaluate the discriminant variables used for flavour tagging;
- reconstruction of exclusive decays relevant for B physics.

LVL2 Tracking Algorithms: SiTrack

The SiTrack algorithm follows a combinatorial pattern recognition strategy, arranging the Pixel and SCT space points in "logical layers", corresponding to sets of detector layers.

SiTrack is implemented as a set of algorithmic tools:

- track seed formation;
- primary vertexing;
- track seed extension;
- extension merging;
- clone removal.

LVL2 Tracking Algorithms: SiTrack

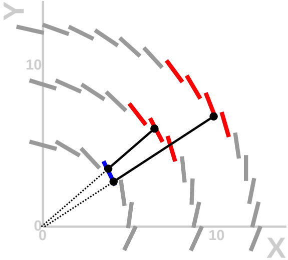
The SiTrack algorithm follows a combinatorial pattern recognition strategy, arranging the Pixel and SCT space points in "logical layers", corresponding to sets of detector layers.

SiTrack is implemented as a set of algorithmic tools:

- track seed formation;
- primary vertexing;
- track seed extension;
- extension merging;
- clone removal.

Track seed formation

- combinatorial formation of SP couples, using SPs coming from the innermost two logical layers;
- straight line fit of the track seed, extrapolated to the beam line;
- cut on transverse impact parameter, which fixes the lowest recontructible p_T value.



LVL2 Tracking Algorithms: SiTrack

The SiTrack algorithm follows a combinatorial pattern recognition strategy, arranging the Pixel and SCT space points in "logical layers", corresponding to sets of detector layers.

SiTrack is implemented as a set of algorithmic tools:

- track seed formation;
- primary vertexing;
- track seed extension;
- extension merging;
- clone removal.

Vertex reconstruction along beam

- used for jet reconstruction, to identify the hard *pp* interaction and filter tracks from soft ones;
- skipped for low multiplicity cases (e.g. single lepton reconstruction);
- longitudinal impact parameter (z0) of seeds used to fill a histogram;
- search histogram maxima;
- more than one vertex candidate can be retained;
- seeds not pointing to a reconstructed vertex are discarded.

LVL2 Tracking Algorithms: SiTrack

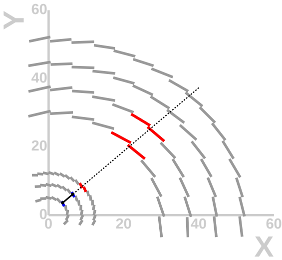
The SiTrack algorithm follows a combinatorial pattern recognition strategy, arranging the Pixel and SCT space points in "logical layers", corresponding to sets of detector layers.

SiTrack is implemented as a set of algorithmic tools:

- track seed formation;
- primary vertexing;
- track seed extension;
- extension merging;
- clone removal.

Track seed extension

- each seed is extrapolated to outer logical layers;
- combinatorial formation of more SP triplets for each seed;
- cut on third space point distance from the extrapolated seed;
- extended triplet fit: straight line in the longitudinal plane, circle in the transverse plane.



LVL2 Tracking Algorithms: SiTrack

The SiTrack algorithm follows a combinatorial pattern recognition strategy, arranging the Pixel and SCT space points in "logical layers", corresponding to sets of detector layers.

SiTrack is implemented as a set of algorithmic tools:

- track seed formation;
- primary vertexing;
- track seed extension;
- extension merging;
- clone removal.

Extension merging

- all the extension triplets found for each seed are merged into one single full track;
- groups of extension triplets with similar track parameters are used to define the full track;
- all the other extension triplets are discarded;
- full track parameters are re-evaluated after merging.

LVL2 Tracking Algorithms: SiTrack

The SiTrack algorithm follows a combinatorial pattern recognition strategy, arranging the Pixel and SCT space points in "logical layers", corresponding to sets of detector layers.

SiTrack is implemented as a set of algorithmic tools:

- track seed formation;
- primary vertexing;
- track seed extension;
- extension merging;
- clone removal.

Clone removal

- two full tracks obtained from different track seeds may sher most of their SPs (clones);
- only the clone track containing the largest number of space points is retained;
- in case more clone tracks contain the same number of space points, the one with the lowest chi squared value is retained.

LVL2 Tracking Algorithms: IDScan

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 SiTrack.
  This is a mandatory step for IDScan, since it provides a reference point for subsequent histogramming;
- Pixel and SCT SPs are then put in a 2D histogram in (η, ϕ); the content of bins with hits in at least four different layers are accepted; all the other SPs are rejected;
- selected SPs are put in another 2D histogram in $(1/p_T, \phi_0)$; bins containing hits in at least four different layers are taken as candidate tracks;
- clone tracks removal is performed for the candidate tracks.

LVL2 Common Tracking Tools: 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; switching between them is done via configuration.

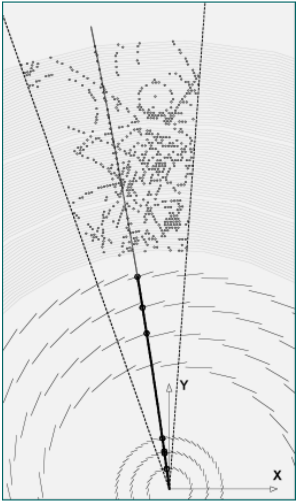
- Distributed Kalman Filter performing a full track fit:
  - uses non-uniform ATLAS B-field map;
  - creates the set of surfaces for the extrapolation;
  - extrapolation performed using a parabolic approximation;
- Distributed Kalman Filter performing a perigee parameter estimation:
  - uses uniform approximation of B-field;
  - surface-to-surface extrapolation is not needed;
  - uses an helical track approximation;
- Karimaki fit:
  - uses non-uniform ATLAS B-field map;
  - performs a straight line fit in the longitudinal direction;
  - performs a circle fit with Karimaki method in the transverse plane.

LVL2 Common Tracking Tools: Extension Tools

Tracks identified in the Silicon detectors can be propagated into the TRT using track following. Tracking in the TRT must be robust, given the high occupancy of the detector (up to 50%).

Currently a TRT extension tool based on the Probabilistic Data Association Filter technique is available:

- single-pass recursive algorithm, computationally similar to the Kalman filter, with almost linear execution time;
- follows a single track, associating at most one hit per layer; all other hits are rejected as background; allows for tracks undetected in some layers;
- consists of the two main blocks: data association and track update.

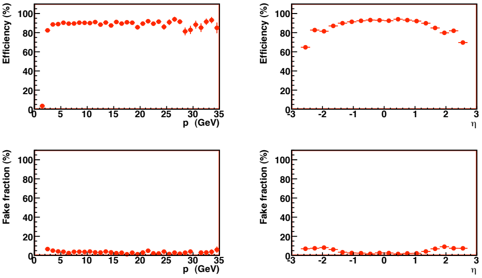


LVL2 Tracking on Simulated Data

Tracking performance is constantly checked on MC samples.
Track reconstruction on *b*-jets from the decay of an Higgs boson ($m_H =$ 120 GeV); no additional pile-up.

Algorithm configuration:

- $\Delta\eta \times \Delta\phi = 0.4 \times 0.4$ RoI;
- primary vertex: high p_T are selected and three vertex candidates are retained, obtaining efficiencies exceeding 95%;
- track selection cuts are tuned to achieve full efficiency down to p_T values around 2 GeV;
- tight track quality selections are applied, to reduce the fake fraction and to provide an high purity track sample.



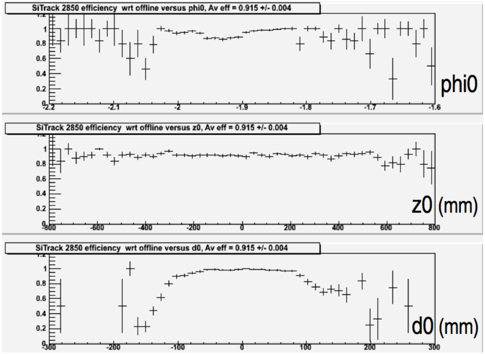
Efficiency	88.2 ± 0.6 %
Fake	4.3 ± 0.1 %

LVL2 Tracking on Real Data

A SCT+TRT surface cosmic run was performed in June 2006.

Cosmics tracks differ from simulated ones:

- large impact parameter w.r.t. beamline;
- recorded by an imperfectly aligned detector;
- no magnetic field; sample populated by low p_T tracks, which suffer from large multiple scattering.



SiTrack reconstructs cosmic tracks with an efficiency exceeding 90% (w.r.t. off-line).
No code adaptation needed; specific tuning for misaligned SPs and displaced vertexes.

LVL2 Vertexing: Secondary Vertex

Vertexing at LVL2 is constrained by track reconstruction: track parameters are estimated at perigee and errors are represented by covariance matrices.

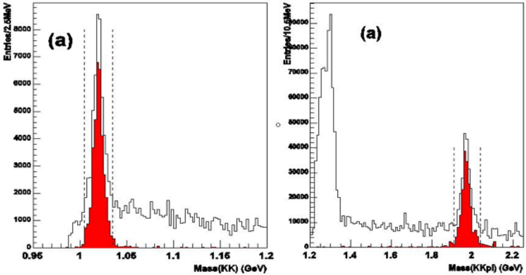
This led to the development of a fast vertex fit algorithm:

- performs a geometrical vertex fit, without mass constraints, equivalent to Billoir's full vertex fit;
- uses a linear transformation of track parameters to reduce track covariance to a block-diagonal form ($2\times2 + 3\times3$);
- this enables to implement a Kalman filter with reduced-size measurement model; no time-consuming matrix inversion;
- produces estimated vertex position (x, y, z), estimated track parameters at vertex and full covariance matrix for the vertexed tracks;
- enables to evaluate invariant mass variance using the full covariance matrix.

LVL2 Vertexing: Secondary Vertex

Vertexing at LVL2 is constrained by track reconstruction: track parameters are estimated at perigee and errors are represented by covariance matrices.

Application to the $D_s \rightarrow \phi(KK)\pi$ B-physics selection



On simulated samples corresponding to a luminosity of $10^{33} cm^{-2} s^{-1}$, this selection reduces the background rate from 5 kHz (after LVL2 muon confirmation) to 200 Hz.

LVL2 Timing

One of the most relevant issues for any LVL2 trigger selection is its timing performance.
Execution times for all the steps of track reconstruction and vertexing at LVL2 are constantly evolving towards lower values, as their optimization is made better and better.

The current status can be summarized as follows; measurements done on a 2.4 GHz Xeon in a 0.1×0.2 RoI on electron/dijet/top events:

- Data preparation for the Si detectors: 2/3/3 ms;
- Track reconstruction execution time: 0.5/0.7/1.0 ms;
- Kalman filter full track fit execution time: $\sim$0.35 ms per track;
- Data preparation for the TRT: 2.0/2.6/4.0 ms;
- Track extension into the TRT: $\sim$3 ms per track;
- Overall timing in the worst case: $\sim$15.5 ms.

EF Tracking Algorithms

The entire Event Filter trigger selection software is based on the idea of reusing as much as possible the code developed for off-line reconstruction.

Obviously, to cope with the limited time budget available at trigger level, some adaptations are mandatory; among them:

- off-line code is executed within "wrapper" algorithms, which can access only data contained in the RoIs; this is a major difference w.r.t. the off-line framework, where reconstruction algorithms can access data from the entire event;
- some tuning of algorithm parameters can be performed, e.g. to reduce number of iterations to save computing power.

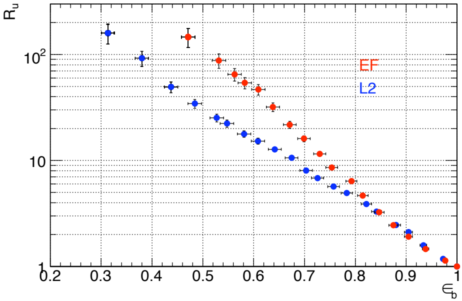
In conclusion, the technical details of the algorithms will not be described in this second part, as they have already been covered yesterday in the talk on off-line tracking software given by W. Liebig.

EF Tracking on Simulated Data

As for LVL2 algorithms, the performance of EF tracking algorithms is constantly checked on MC samples.

Studies on the comparison and the correlation between LVL2 and EF are performed for all the trigger selections using Inner Detector tracking.

A relevant example is on-line *b*-tagging, where both tracking efficiency and track parameter resolution significantly contribute to the rejection power.



EF Vertexing

Vertexing tools are ready in the off-line framework and available for use at Event Filter level. Work is ongoing to include them in trigger strategies.

B-physics semi-inclusive or exclusive selections:

- secondary vertex reconstruction will be used to reject fake combinations which are not corresponding to a common vertex;
- refit of the tracks associated to a given vertex will be used to refine the invariant mass constraints applied;

b-tagging selections:

- primary vertex reconstruction in (x, y, z) will improve the impact parameter resolution for the reconstructed tracks, consequently improving the discriminant power of the impact parameter based variables;
- reconstruction of secondary vertexes will enable to implement additional discriminant variables, uncorrelated with the ones based on impact parameters: e.g. jet multiplicity, invariant mass, fraction of jet energy associated to the secondary vertex particles.

EF Timing

Timing constraints at the EF are more relaxed than at the LVL2; anyway, to be capable of using off-line algorithms without the need for major adaptations, the execution time for all the track reconstruction steps must be constantly checked.

The current status can be summarized as follows; measurements done on a 2.8 GHz Dual P4 in a 0.1×0.2 RoI on electron/dijet/top events:

- Data preparation for the Si detectors: 15/17/15 ms;
- Track reconstruction and fit execution time: 13/165/130 ms;
- Data preparation for the TRT: 14/15/11 ms;
- Track extension into the TRT: 9/22/41 ms;
- Primary vertex reconstruction: 1.0/0.6/2.8 ms;
- Overall timing in the worst case: $\sim$220 ms.

Conclusions

The ATLAS tracking and vertexing trigger software is a mature project, well studied and validated on simulated samples.

It also underwent some first basic tests with real data, but ...

... there are no real conclusions yet ...

as we are currently getting ready to reconstruct cosmics in the pit with the full Inner Detector and we're looking forward to test the tracking software with real data from the first collisions.