PROCEEDINGS OF SCIENCE

A Fast Vertex Fitting Algorithm for ATLAS Level 2 Trigger

Dmitry Emeliyanov*
STFC Rutherford Appleton Laboratory,
Harwell Science and Innovation Campus, Didcot, OX11 0QX, United Kingdom
E-mail: `Dmitry.Emeliyanov@cern.ch`

A vertex fitting algorithm developed for the Level 2 Trigger of the ATLAS experiment is presented. The algorithm features a Kalman filter with a decorrelating measurement transformation which reduces the computational burden of the vertex fit. The algorithm has been tested on data produced using a full Monte Carlo detector simulation. Results regarding the precision and speed of the algorithm are presented.

PACS: 07.05.Kf; 29.85.+c

XI International Workshop on Advanced Computing and Analysis Techniques in Physics Research
April 23-27 2007
Amsterdam, the Netherlands

*Speaker.

PoS(ACAT)058

1. Introduction

ATLAS is one of the particle detector experiments currently being constructed at the Large Hadron Collider (LHC) at CERN in Switzerland. When completed, ATLAS will be 46 metres long and 25 metres in diameter, and will weigh about 7,000 tonnes. ATLAS is designed as a general-purpose detector and consists of the Inner Detector, the calorimeters, the muon spectrometer and the magnet systems [1].

ATLAS has a three level trigger system [2] which reduces the initial 40 MHz rate to about 100 Hz of events to be recorded. The first level trigger (LVL1) is a hardware-based trigger which makes a fast decision (with latency 2.5 μs) selecting events which are of interest for further processing. The LVL1 trigger reduces the rate to below 75 kHz and identifies regions of the detector ("Regions of Interest", RoIs) which contain interesting signals (e.g. high-p_T electrons). The LVL1 RoIs are used and refined in the subsequent trigger levels to guide the trigger reconstruction. The High Level Trigger (HLT) is software-based and consists of two levels. At Level 2 (LVL2) the full granularity of the detector is used to confirm the LVL1 results and then to combine information from various sub-detectors within the LVL1 RoIs. This stage of event selection employs fast reconstruction algorithms and has a time budget of about 10 ms (25-30 ms on a 2.4 GHz CPU). The LVL2 output rate is about 1-2 kHz. Finally at the Event Filter (EF), "offline-like" algorithms are used along with better alignment and calibration information to produce a final decision about whether or not an event is accepted. With an execution time of about 1 s, the rate is reduced to 100 Hz.

The experiment has a broad B-physics programme and plans to study CP violation (in $B^0_d \to J/\psi K^0_S$ and $B^0_s \to J/\psi\phi$ channels), B^0_s oscillations (using $B^0_s \to D_s\pi$ and $B^0_s \to D_s a_1$ channels), and to search for rare decays (e.g. $B^0_{d,s} \to \mu^-\mu^+(X)$, $B^0_s \to \phi\gamma$). However, ATLAS is a general-purpose experiment with an emphasis on high-p_T physics and as such has only a limited bandwidth (about 5-10 %) for the B-physics events. Thus accommodating the B-physics programme requires a highly selective trigger with exclusive or semi-inclusive reconstruction of decays already at the LVL2.

An essential part of the B-physics event selection is vertex finding and fitting using tracks reconstructed by the LVL2 tracking algorithms as input. Due to the LVL2 timing constraints a vertex fitting algorithm for the LVL2 application has to be fast. An additional requirement stems from the LVL2 track reconstruction which provides input track parameter errors in form of a covariance matrix. In contrast, vertex fitting algorithms proposed in the literature [3]–[5] assume uncertainties of the input track parameters to be described by weight (inverse covariance) matrices. However, if only track covariance matrices are available, these algorithms require them to be inverted beforehand thus resulting in substantial computing time overhead.

To alleviate this drawback a fast vertex fitting algorithm capable of using track covariance matrices directly (i.e. without time-consuming inversion) has been developed. The specific feature of the algorithm is that track momenta at perigee points rather than "at-vertex" momenta are selected as the fit parameters. Such a choice of fit parameters makes it possible to apply a decorrelating measurement transformation so that the transformed measurement can be partitioned into two uncorrelated vectors – measured momenta and its linear combination with measured track coordinates at the perigee. This linear combination comprises a new 2D measurement model while the measured momenta and the corresponding blocks of the input track covariance matrices are used to initialize a parameter vector and covariance matrix of the vertex fit. This approach provides a mathematically correct and numerically stable initialization of the vertex fit. The reduced size (2D instead of 5D as in the algorithm [4]) of the measurement model makes the proposed Kalman filter very fast and therefore suitable for an online application in the ATLAS Level 2 Trigger.

2. The vertex fitting problem

Let n reconstructed tracks $\{t_k\}$, $k = 1,\ldots,n$ originate from a common point – vertex $\mathbf{V}$ (Fig.1). The problem is to estimate a vector $\mathbf{X}$ of fit parameters – vertex position $\mathbf{R}$ and track momentum vectors $\mathbf{T}_k$, $k = 1,\ldots,n$ using measured track parameters $m_k = (m_k^r, m_k^q)^T$, $k = 1,\ldots,n$, where m_k^r is a measurement of a track position r_k at some reference surface and m_k^q is a measurement of track momentum q_k at the reference surface.

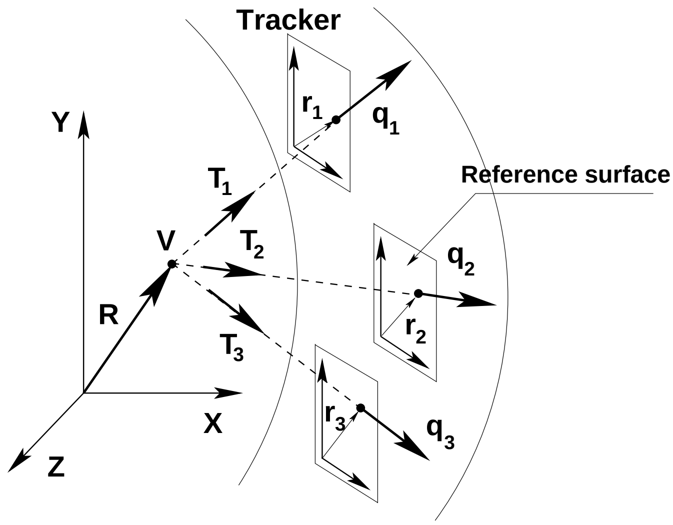

Figure 1: Geometry of a vertex fit.

The dependence of the measured track parameters m_k on the fit parameter vector $\mathbf{X}$ is given by the measurement equation

$$m_k = H(\mathbf{R}, \mathbf{T}_k) + \varepsilon_k, \quad (2.1)$$

where $H(\cdot,\cdot)$ is a 5D nonlinear measurement function and ε_k is the error of track parameter measurement, assumed to be a Gaussian random vector with a 5×5 covariance matrix $\text{cov}(\varepsilon_k) = V_k$.

3. Measurement model reduction

Usually, track momenta *at a vertex* are used as the fit parameters $\mathbf{T}$. Alternatively, track momenta *at a measurement surface* q_k can be chosen as the fit parameters such that $\mathbf{T}_k = q_k$, $k = 1,\ldots,n$. Such a choice reduces the size of the non-linear part of the measurement equation (2.1), which can be then partitioned as follows

$$m_k^r = h(\mathbf{R}, q_k) + \varepsilon_k^r \quad (3.1)$$

$$m_k^q = q_k + \varepsilon_k^q, \quad (3.2)$$

where a 2D vector ε_k^r and 3D vector ε_k^q are parts of the vector ε_k.

Let the Kalman filter formalism be applied to the measurement model given by Eqs. (3.1),(3.2). In principle, the measured track momentum m_k^q can be used as a *prior* estimate $\widehat{q}_k^0$ of the momentum q_k with a covariance matrix given by the V_k^{qq} block of the covariance matrix V_k. However, in the standard Kalman filter, the error of a prior estimate and the measurement error are assumed to be uncorrelated. Obviously, this does not hold for the measurement model (3.1),(3.2) as, in general, ε_k^q and ε_k^r are correlated and the corresponding covariance block V_k^{rq} is non-zero. This difficulty can be rectified by applying a linear *decorrelating* transformation.

We can replace Eqs. (3.1),(3.2) by an equivalent system

$$m_k^r + L_k m_k^q = h(\mathbf{R}, q_k) + L_k q_k + \varepsilon_k^r + L_k \varepsilon_k^q \quad (3.3)$$

$$m_k^q = q_k + \varepsilon_k^q, \quad (3.4)$$

where matrix L_k is such that the vectors ε_k^q and $\varepsilon_k^r + L_k \varepsilon_k^q$ are uncorrelated. The matrix L_k can be written in terms of the V_k blocks as

$$L_k = -V_k^{rq} \left(V_k^{qq}\right)^{-1}.$$

The modified measurement model splits into two uncorrelated parts – Eq. (3.4), which defines a prior estimate of the track momentum, and Eq. (3.3), which is, in fact, a new measurement equation for the Kalman filter.

4. A fast vertex fitting algorithm

Using Eqs. (3.3),(3.4) we can derive equations of a Kalman filter step that adds a new, $k+1$-th track to a vertex already fitted with k tracks. For a k-prong vertex, the fit parameter vector reads

$$\mathbf{X}_k = (\mathbf{R}, q_1, \ldots, q_k)^T \quad (4.1)$$

Let $\widehat{\mathbf{X}}_k$ be an estimate of $\mathbf{X}_k$ with covariance matrix Γ_k. Following the structure of the vector $\mathbf{X}_k$, this matrix can be partitioned into four blocks:

$$\mathrm{cov}\left(\widehat{X}_k\right) = \Gamma_k = \begin{pmatrix} C_k & E_k^T \\ E_k & D_k \end{pmatrix},$$

where C_k is a 3×3 vertex covariance, D_k is a $3k \times 3k$ joint covariance matrix of the track momenta, and E_k is a $3k \times 3$ matrix of mutual "vertex-track" correlations.

If a $k+1$-th track is added to the vertex the vector X_k has to be augmented and its prior estimate (prediction) is

$$\widetilde{X}_{k+1} = \left(\widehat{X}_k, \widehat{q}_{k+1}^0\right)^T.$$

If a prior estimate $\widehat{q}_{k+1}^0$ is defined by Eq. (3.4) the prediction, $\widetilde{X}_{k+1}$, and its covariance become

$$\widetilde{X}_{k+1} = \begin{pmatrix} \widehat{X}_k \\ m_{k+1}^q \end{pmatrix}, \quad \widetilde{\Gamma}_{k+1} = \begin{pmatrix} \Gamma_k & 0 \\ 0 & V_{k+1}^{qq} \end{pmatrix}. \quad (4.2)$$

The Kalman filter updates the prediction as follows:

$$\widehat{X}_{k+1} = \widetilde{X}_{k+1} + K_{k+1} d_{k+1}, \tag{4.3}$$

where K_{k+1} is the Kalman filter gain, d_{k+1} is a 2D residual between the actual measurement $m^r_{k+1} + L_{k+1} m^q_{k+1}$ and its prediction. In accordance with Eq. (3.3), the predicted measurement is $h(\widehat{\mathbf{R}}_k, \widehat{q}^0_{k+1}) + L_{k+1}\widehat{q}^0_{k+1}$, where $\widehat{\mathbf{R}}$ denotes an estimate of the vertex position given by the first three components of the vector $\widehat{\mathbf{X}}$. Note that, if $\widehat{q}^0_{k+1} = m^q_{k+1}$ the $L_{k+1} m^q_{k+1}$ terms enter both actual and predicted measurements. They cancel out and the residual becomes

$$d_{k+1} = m^r_{k+1} - h\left(\widehat{\mathbf{R}}_k, m^q_{k+1}\right). \tag{4.4}$$

A 2×2 covariance matrix S_{k+1} of the residual (4.4) is given by

$$S_{k+1} = A_{k+1} C_k A^T_{k+1} - B_{k+1} V^{qr}_{k+1} - V^{rq}_{k+1} B^T_{k+1} + B_{k+1} V^{qq}_{k+1} B^T_{k+1} + V^{rr}_{k+1}, \tag{4.5}$$

where A_{k+1}, B_{k+1} are matrices obtained by linearizing the measurement function $h(\cdot,\cdot)$ in the vicinity of the prediction $\widetilde{\mathbf{X}}_{k+1}$:

$$A_{k+1} = \left.\frac{\partial h}{\partial \mathbf{R}}\right|_{\widehat{\mathbf{R}}_k, m^q_{k+1}} \qquad B_{k+1} = \left.\frac{\partial h}{\partial q}\right|_{\widehat{\mathbf{R}}_k, m^q_{k+1}}. \tag{4.6}$$

In turn, the Kalman filter gain is given by

$$K_{k+1} = M_{k+1} S^{-1}_{k+1}, \tag{4.7}$$

where a $(3k+6) \times 2$ matrix M_{k+1} denotes the following matrix expression

$$M_{k+1} = \begin{pmatrix} C_k A^T_{k+1} \\ E_k A^T_{k+1} \\ V^{qq}_{k+1} B^T_{k+1} - V^{qr}_{k+1} \end{pmatrix}.$$

An updated covariance matrix of the estimate $\widehat{\mathbf{X}}_{k+1}$ is

$$\Gamma_{k+1} = \widetilde{\Gamma}_{k+1} - K_{k+1} M^T_{k+1}. \tag{4.8}$$

Finally, a χ^2 contribution of the $k+1$-th track to the total χ^2 of the vertex fit is given by

$$\Delta\chi^2_{k+1} = d^T_{k+1} S^{-1}_{k+1} d_{k+1}. \tag{4.9}$$

Since d_{k+1} is a 2D vector, the contribution $\Delta\chi^2_{k+1}$ has two degrees of freedom.

The system (4.2)-(4.9) completely describes a fast vertex fitting algorithm. This algorithm is mathematically equivalent to the algorithm [4] and has a number of computational advantages. Due to using the so-called "gain matrix" formalism [4], track covariance matrices V_k, $k = 1,\ldots,n$ are utilised directly and as soon as the last track is processed the estimate of the full fit parameters vector $\widehat{X}_n$ and its covariance Γ_n are available immediately, i.e. no smoothing pass [4] is needed. Another advantage of the algorithm is that the only matrices to invert are 2×2 symmetrical matrices S_k, $k = 1,\ldots,n$. This feature is especially important since vertex fitting is an iterative procedure

– after processing all tracks the linearizing (4.6) is repeated using the estimated vertex position and the Kalman filter cycle (4.2)-(4.9) is repeated for each track. Typically, a few iterations are required for convergence so that the overall computational cost of the vertex fit can be significantly reduced by using the fast Kalman filter (4.2)-(4.9) in the iteration loop. It is also worth mentioning that although the equations of the filter are derived using the decorrelating measurement transformation, in fact, explicit transformation of the measurements m_k, $k = 1,\ldots,n$ is not needed as their components and blocks of the covariance matrices V_k, $k = 1,\ldots,n$ are used directly in the filter equations.

5. Algorithm validation

The algorithm has been validated on data produced using a full ATLAS Monte Carlo (MC) simulation. The results described in this section have been obtained on $B_s \to J/\psi(\mu^+\mu^-) + \Phi(K^+K^-)$ dataset. Track reconstruction has been performed by the LVL2 tracking algorithms [6],[7]. Muon and kaon tracks have been selected according to the MC truth information and combined into $J/\psi(\mu^+\mu^-)$ and $B_s(\mu^+\mu^-K^+K^-)$ vertices. These vertices have been fitted using the algorithm (4.2)-(4.9) and evaluated using the following performance criteria:

- normalized residuals (pulls) P_x, P_y, and P_z of estimated vertex coordinates. The pull (for example, P_x) is defined as follows

$$P_x = \frac{\widehat{x}_n - x_{MC}}{\sqrt{C_n^{xx}}},$$

  where $\widehat{x}_n$ is an estimated x-coordinate of a vertex, x_{MC} is a vertex coordinate from the MC truth, C_n^{xx} is a corresponding element of a vertex covariance.

- χ^2 probability defined as

$$p_{\chi^2} = \int_{\chi^2}^{+\infty} f_{\chi^2}(z, n_{DOF})\, dz,$$

  where χ^2 is the total χ^2 of a vertex fit, n_{DOF} is the number of degree-of-freedom of the χ^2, $f_{\chi^2}(\cdot,\cdot)$ is a p.d.f. of the χ^2 distribution. If the χ^2 values produced by the vertex fit obeys the χ^2 law the p_{χ^2} will be uniformly distributed between 0 and 1.

- computing time needed to fit a vertex.

The normalized residuals and χ^2-probability distribution for the $J/\psi(\mu^-\mu^+)$ vertices are shown in Fig.2 and their parameters are summarized in Table.5.

Pull r.m.s.			p_{χ^2} distribution parameters	
P_x	P_y	P_z	mean	r.m.s
1.09	1.09	1.09	0.49	0.29

Table 1: Parameters of the vertex pulls and χ^2-probability distribution.

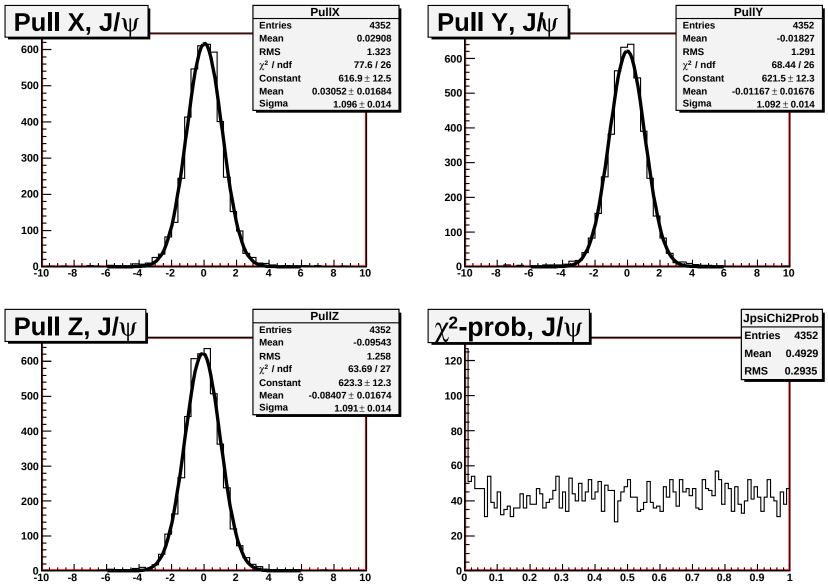

Figure 2: Vertex pulls and χ^2-probability.

The pull distributions in Fig.2 are nearly perfectly Gaussian with an r.m.s close to 1. This indicates that covariance matrices produced by the vertex fit correctly reflect the actual estimation errors of the vertex parameters. Also these results confirm that the decorrelating transformation of the measurement model does not introduce any bias. The χ^2-probability distribution is flat and its parameters are close to those of a uniform distribution between 0 and 1. There is a small (less that 2 %) excess of vertices with a χ^2-probability below 0.01. Typically, these vertices contain poorly reconstructed tracks, e.g. with pixel hit-outliers – hits located near the true trajectory and erroneously included in track fit.

The computing time of the vertex fit has been measured on a Xeon 2.4 GHz processor as a function of track multiplicity n. The number of iterations in the vertex fit has been fixed at 5. The results for $J/\psi(\mu^+\mu^-)$ ($n=2$) and $B_s(\mu^+\mu^-K^+K^-)$ ($n=4$) vertices are shown in Fig. 3 and summarised in Table.2.

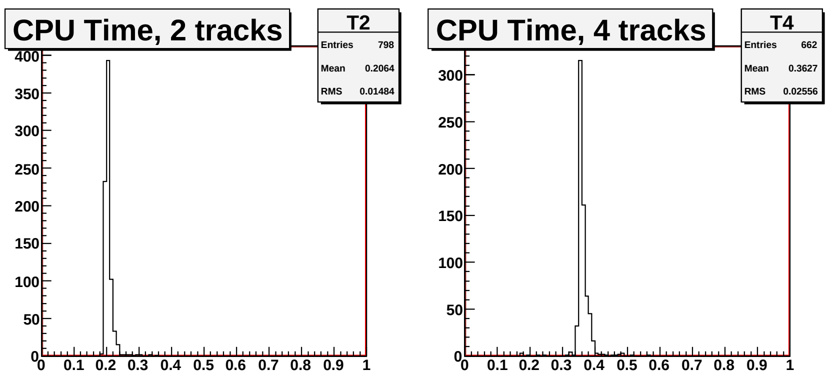

Figure 3: Computing time of the vertex fit.

As can be seen, the vertex fitting algorithm is very fast. Its computing time is negligible

n	2	4
CPU time, ms	0.20	0.36

Table 2: Average computing time per vertex.

with respect to the available LVL2 time budget which is approximately 25-30 ms for a 2.4 GHz processor.

6. Conclusion and outlook

A fast vertex fitting algorithm developed for the ATLAS Level 2 Trigger has been presented. The algorithm features a Kalman filter with a reduced-size measurement model and decorrelating measurement model transformation, which reduce the computational burden of the vertex fit.

The algorithm has been successfully validated on simulated B-physics data and shown to have the required performance. The computing time measurements have shown that the algorithm is fast enough to meet the Level 2 Trigger timing constraints.

Although the algorithm has been developed specifically for fitting secondary vertices of B-mesons decays the algorithm application to fitting primary vertices with high track multiplicity might also be very promising.

References

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

[2] *ATLAS High Level Trigger, Data Acquisition and Controls Technical Design Report*, ATLAS Collaboration, CERN/LHCC 2003-22 (2003).

[3] P. Billoir, R. Frühwirth and M. Regler *Nucl. Instr. and Meth.* **A241** (1985) 115.

[4] R. Frühwirth, *Nucl. Instr. and Meth.* **A262** (1987) 444.

[5] R. Frühwirth et al., *Comput. Phys. Comm* **96** (1996) 189.

[6] N. Konstantinidis et. al., *Nucl. Instr. and Meth.* **A566**, 1, (2006) 166.

[7] D. Emeliyanov, *Nucl. Instr. and Meth.* **A566**, 1, (2006) 50.