The ZEUS Global Tracking Trigger Barrel Algorithm

M.R.Sutton*
University College London, Gower Street, London WC1E 6BT, United Kingdom
on behalf of the ZEUS GTT group

Abstract

The current design, implementation and performance of the ZEUS global tracking trigger barrel algorithm are described. The ZEUS global tracking trigger integrates track information from the ZEUS central tracking chamber (CTD) and micro vertex detector (MVD) to obtain a global picture of the track topology in the ZEUS detector at the second level trigger stage. Algorithm processing is performed on a farm of Linux PCs and, to avoid unacceptable deadtime in the ZEUS readout system, must be completed within the strict requirements of the ZEUS trigger system. The GTT plays a vital rôle in the selection of good physics events and the rejection of non-physics background within the very harsh trigger environment provided by the upgraded HERA collider. The GTT greatly improves the vertex resolution and the track finding efficiency of the ZEUS second level trigger while the mean event processing latency and throughput are well within the trigger requirements. Recent running experience with HERA production luminosity is briefly discussed.

INTRODUCTION

The silicon Micro Vertex Detector (MVD) [1] upgrade of the ZEUS experiment, installed during the HERA luminosity upgrade shutdown, adds 2-4 planes of high spatial resolution tracking information close to the interaction point. Together with the information provided by the existing Central Tracking Detector [2] and the Straw Tube Tracking Detector (STT) [3] this provides high resolution data to allow the detailed reconstruction of tracks and event topology with resolutions better than previously possible in the ZEUS Second Level Trigger (SLT).

Implementing a global tracking algorithm to run at the SLT is particularly challenging, since the environment is very harsh for detailed tracking – the events themselves are not clean physics events since there is a significant contribution from high occupancy, beam gas events which require considerable processing, and the time available for both the algorithm processing and data transfer to and from the GTT processors must be kept within ∼10ms latency to avoid increasing the dead time in the ZEUS trigger chain.

Detailed pattern recognition with multiple global passes over the data to resolve the many pattern recognition ambiguities in the data is not possible and approximations and assumptions have to be used to break ambiguities as soon as possible to keep the required processing to a minimum. In addition, since spacepoints are not available to the algorithm, the event reconstruction must be done using the raw datector information and full complex detector geometry.

THE GTT SYSTEM

The hardware systems of the GTT are discussed in detail elsewhere in these proceedings [4] so only a summary of the algorithm processing environment will be given here. The GTT Algorithms run on a farm of 12 commodity dual CPU 1GHz PC's running Linux. Each algorithm processor runs a multi-threaded algorithm environment for the reconstruction of complete single events using data sent from the CTD, MVD and STT with separate threads for data recieving and independent Barrel (CTD+MVD barrel) and Forward (STT+MVD forward wheels) Algorithms.

The Barrel algorithm is fully integrated into the ZEUS DAQ and Trigger chain. The Forward algorithm is under development but has been successfully tested online.

BARREL ALGORITHM DESIGN

The design of the Barrel algorithm builds heavily upon the existing CTD-SLT algorithm [6]. However, before May 2004 the GTT barrel algorithm the did not have access to the CTD z-by-timing information and instead made use of the data from the CTD stereo superlayers for z reconstruction. Additional hardware to make the z-by-timing data available was added in May 2004, and the algorithm code modified to incorporate this data in the event reconstruction.

After segment finding in the CTD and MVD hit decoding, the barrel algorithm, proceeds in three logical stages;

- r-ϕ track finding, (CTD tracks; MVD r-ϕ hit matching.)
- z-track finding, (CTD z-by-timing and stereo segment matching; MVD z hit matching.)
- Primary vertex identification.

When information on the event vertex and z-tracks are available, a second pass of the z-matching stage and revertexing is performed. A secondary vertexing stage is under development.

CTD segment finding

The axial segment finding in the CTD provides the basic structure from which all the tracks are found. The algorithm used is identical to that used in the current CTD-

[*] sutt@mail.desy.de

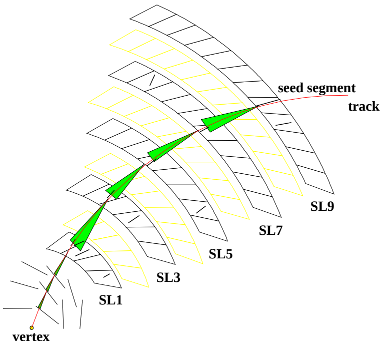

Figure 1: The axial track finding algorithm.

SLT [6]. To save time in the stereo matching section of the algorithm, segment finding is also performed on the hits from the stereo layers.

The segment finder looks for linear combinations of three or more hits in each of the cells of the CTD using the *unsigned* integer drift times and wire numbers. Because the drift time information is unsigned it is not known which side of the wire plane the hits lie, each hit contributing both "real" and "ghost" hits.

The CTD cell geometry has the wire plane for each cell oriented at 45° with respect to the cell radius so that after hits have been assigned to a segment, identifying the real segment is simplified by taking the segment candidate pointing more closely to the beam line so avoiding the additional latency needed were both real and ghost segments to be considered. This has a high efficiency for identifying high p_T tracks, but because of the ϕ asymmetry of the CTD geometry leads to a charge asymmetry for lower p_T tracks.

Identifying tracks in r-ϕ

The initial r-ϕ (axial) track finding is essentially the same as in the CTD-SLT with the addition of matching hits from the MVD. The algorithm, illustrated in Fig. 1, searches for tracks starting with a seed segment in the outer superlayer where the occupancy is lowest. Using this seed segment, the expected azimuthal position of the hit in the next innermost axial superlayer is calculated, and segments consistent with this are matched to the track. The segment last matched is then used as a fresh seed and the matching proceeds again into the next inner axial superlayer until at least one segment is found in superlayer 1. Once the segment matching is complete for a track, the track parameters are calculated using a fast circle fit in r-ϕ constrained to the beam line to aid subsequent hit matching in the MVD.

Once the track has been found in the CTD, the algorithm goes on to find hits in the MVD. Since the MVD hits from both the r-ϕ and z wafers within an MVD half module are multiplexed together, it is not known *a priori* which are z and which r-ϕ hits. Consequently all hits must be considered as potential r-ϕ hit candidates. The algorithm first matches MVD hits in the outermost barrel layer, refiting the track, and looking for hits in turn, in the inner layers.

When a track has at least 2 MVD hits, the track can be refitted without the beam line constraint, in preparation for secondary vertex finding.

Identifying tracks in z-s

After the $r - \phi$ fit has been performed, the algorithm matches hits from the z-by-timing system, matching hits from the z-by-timing system on wires used in the $r - \phi$ fit where the drift times of the hits are constsent between the two systems. Since the z-resolution of the z-by-timing hits is of the order of 7cm the data are then used primarily to provide a guide for the subsequent matching of CTD stereo information.

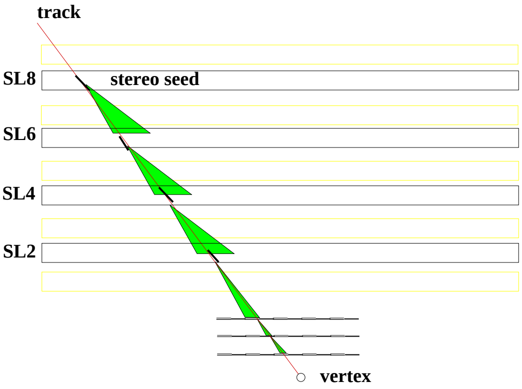

Figure 2: The CTD stereo matching algorithm

Since each stereo wire in the CTD spans a large angle (4 cells) it does not provide very precise information on the axial position of the hit upon it. A z position is only available when the r-ϕ position of the hit on the track has been calculated. Since each stereo hit may be assigned to any track passing through the large range of ϕ spanned by the wire, the r-ϕ and z positions of each hit must be calculated for each possible track candidate within its r-ϕ range. For the inner stereo layers, this range is reasonably large – in superlayer 2 the angle corresponding to 4 cells is as large as 36 degrees – which presents a significant problem since the track occupancy nearer the interaction region is high and the degree of matching ambiguity which must be resolved is large.

The intersection of the track with the hit must be calculated considering the drift displacement of the hit with respect to the wire. This is done using an iterative algorithm [7] and provides the ϕ position of the hit swum to the track, from which is obtained the wire ϕ position. The fraction of the length along the wire is then trivially extracted to provide the z position of the hit.

Solving the track intersections in r-ϕ with the stereo wires is the most costly step in terms of the processing latency so to keep the processing time within acceptable limits, segments are found in the stereo layers using the same algorithm as for the axial layers.

The stereo matching algorithm proceeds as follows:

- Starting in the outer stereo superlayer for this track where the spatial separation of tracks is highest, all possible segments are considered as seed segments.
- The segment end points are swum to the track and a linear fit in z and the transverse path length along the track, s, is performed including the z-by-timing hits in the fit to constrain the track nearer the beamline. The z-track parameters – gradient and track-vertex (z position of the track closest to the beam line) – are calculated and the track extrapolated into the next stereo layer.
- The z positions of each segment in the next inner layer are then calculated by matching to the track as above, continuing in each successive stereo layer, with the fit recalculated at each layer, until a track with segments in each stereo layer is found, or no matching segment is found.

This is illustrated in Fig. 2. For very busy events, the high occupancy in the inner layers means that each inner cell can contain many segments belonging to several tracks. To improve the resolution and efficiency in this case, the algorithm searches for all possible track candidates selecting the candidate with the best fit.

Once CTD stereo segments have been matched to an axial track, the algorithm goes on to match MVD z-hits in essentially the same way as the r-ϕ matching. Since MVD r-ϕ positions are already known, the algorithm looks for unmatched hits only in the corresponding z-wafers of the modules with r-ϕ hits. Each track is assigned a weight based on the number of CTD stereo segments, z-by-timing hits and MVD hits assigned to it. The track-vertex and the weight from the fit are stored for use in the primary vertex fit.

The primary vertex algorithm

Beacuse of the large degree of residual beam gas contamination in the event spectrum for events accepted at the First Level Trigger, the presence of a clean ep interaction vertex cannot be guaranteed. The primary vertex algorithm is intended to make a fast estimation of the presence of a possible vertex and to calculate its likely position in z.

A binning algorithm with overlapping 13cm bins is used, looping over all tracks, binning the track-vertex intersections from the z-s fit with the square of the track weight to automatically take account of the track quality and the different spacial resolutions of the MVD and CTD.

The most probable bin (MPB) – the bin with the highest number of weights – is found and from the tracks in this bin an initial vertex position is calculated using

$$z_{\text{initial}} = \frac{\sum_{i\in\text{MPB}} z_i w_i^2}{\sum_{i\in\text{MPB}} w_i^2}.$$

All tracks within ± 9cm of this initial vertex are then used to calculate the event vertex, again using the weighted mean. This is found to be very stable against the present of incorrectly fitted or assigned tracks.

The stereo z-segment matching and vertex fit is then repeated using the event vertex to increase the z-segment matching efficiency.

ALGORITHM PERFORMANCE

Luminosity data taking

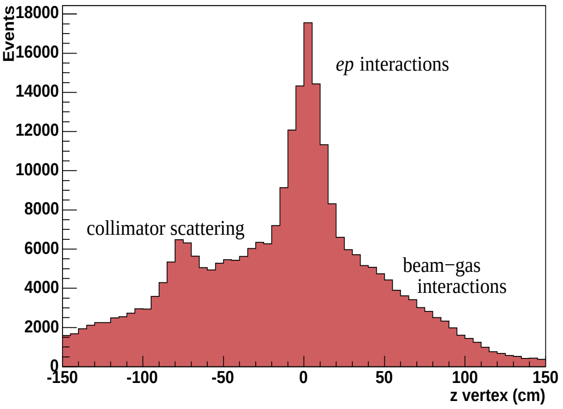

Figure 3: The online CTD-only GTT vertex

The GTT and barrel algorithm, have been in operation stably since the HERA upgrade was completed in 2002. Following additional modifications to the HERA machine in 2003, the GTT has been running under production luminosity conditions, with the results of the algorithm being used by the physics filters to select events online with over 40 pb^{-1} currently on tape. The HERA beam gas related background is found to be significantly larger than before the upgrade. To compensate, the CTD had to be operated at only 95% of the nominal high voltage setting leading to a small loss in chamber performance. In addition, although the MVD hit matching is satifactory for normal physics events, the high occupancy in beam gas events causes problems for the current MVD matching algorithm which biases the vertex distribution in beam gas events. As such, for the 2003-04 running period the MVD hit matching was disabled.

The performance and stability of the algorithm and GTT system as a whole, was well within the expectation during this period. The GTT event vertex available online is illustrated in Fig. 3 and clearly shows events from ep interactions in a vertex peak within ± 25cm, on the large proton-beam gas background, together with secondary scattering events from the collimator at -80cm.

Selecting photoproduction events offline to eliminate any beam gas contamination, the event vertex residual with respect to the offline vertex (with a resolution of around 2mm) is illustrated in Fig. 4. The data have been fitted with a sum of two Gaussians with widths of 2cm and 6cm respectively.

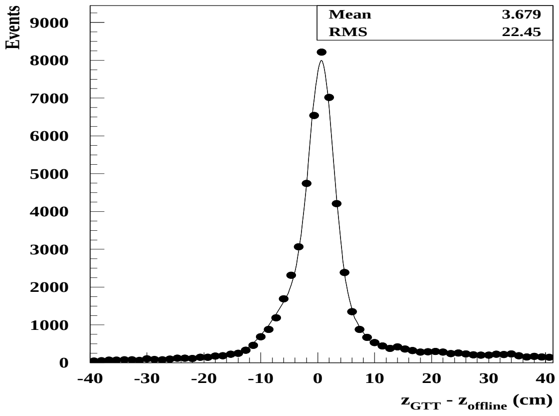

Figure 4: The online CTD-only GTT vertex resolution with respect to the offline vertex

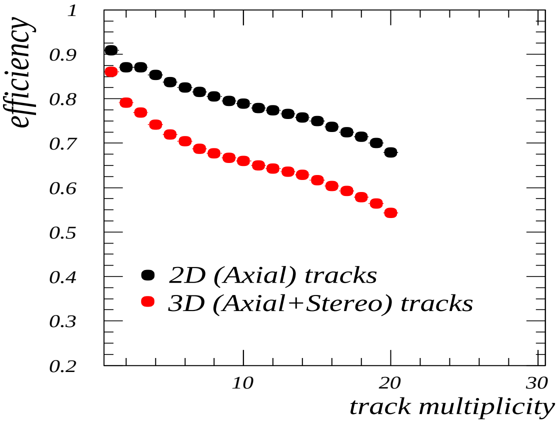

Figure 5: The online track efficiency versus offline track multiplicity.

The efficiency for finding the vertex online within ±60cm of the nominal interaction point is seen to be greater than 90% for vertices with more than 5 tracks.

Detailed studies of the online performance in data are underway, comparing with the offline reconstructed tracks. To summarise briefly, the track resolutions for full length tracks are found to be 0.07GeV^{-1}, 12mrad and 0.05 for p_T, ϕ and η respectively.

The track finding efficiency is around 75% for finding full length tracks in r-ϕ and and 65% when including information in z. Because of the more complex pattern recognition at high multiplicities, the efficiency falls steeply with multiplicity and is shown in Fig. 5.

Algorithm latency

To avoid introducing dead time at the Global First Level Trigger (GFLT) the latency for the complete GTT readout and event reconstruction has to be within around 10ms with reasonably short tails.

The mean overall GTT latency seen by the ZEUS GSLT during 2003-04 running was typically around 10ms, within that required, with a short tail extending to around 40ms.

The algorithm processing latency is a monotonically falling distribution, since the algorithm runs as a single process, with a mean of around 2ms and a tail extending to around 15ms for busy events. The overall latency is dominated by the data transfer time of the large detector data volumes which has a mean of around 7ms for the CTD data.

During 2003-04, the dependence of the mean overall latency on the output rate of the GFLT was seen to be small. always lying within 15ms for all rates up to the design rate of 500Hz.

SUMMARY AND OUTLOOK

The GTT barrel algorithm performed well during the 2003-04 production luminosity running period with high stability and latencies well within those required by the ZEUS DAQ and trigger systems with up 40pb^{-1} written to tape.

Modifications to the GTT and algorithm to use the CTD z-by-timing data were performed in May 2004 to improve the track finding efficiency. In order to use the MVD hits online, further modifications are being made to the barrel algorithm MVD hit matching which should improve the vertex finding efficiency and resolution and should be in place soon after the HERA restart in October 2004.

To allow the inclusion of MVD data in the online algorithm and improve the tracking and vertexing efficiency and resolution are ongoing and should be ready when the HERA machine restarts in September of 2003.

REFERENCES

[1] ZEUS Collaboration; “A Microvertex Detector for ZEUS”, ZEUS internal note ZN-97-006 (1997) (unpublished).

[2] N.Harnew *et al*; Nucl. Inst. Meth. **A279** (1989).
B.Foster *et al*; Nucl. Inst. Meth. **A338** (1994) 254.

[3] ZEUS Collaboration; “A Straw-Tube Tracker for ZEUS”, ZEUS internal note ZN-98-046 (1998) (unpublished).

[4] M.R.Sutton; “The Architecture of the ZEUS Second Level Global Track Trigger”, these proceedings.

[5] M.Sutton and C.Youngman; “The Global Tracking Trigger System and Architecture”, ZEUSinternal note ZN-99-074 (1999) (unpublished).

[6] A.Quadt *et al*; “The design and performance of the ZEUS Central Tracking Detector Second Level Trigger”, Nucl. Inst. Meth. **A438** (1999) 472.

[7] G.Hartner; “VCTRAK Briefing: Program and Math”, ZEUS internal note ZN-98-059 (1998). (unpublished)

[8] R.Hall-Wilton, M.R.Sutton and B.J.West; “The CTD-BMVD version 3.3 Global Tracking Trigger Algorithm and Performance”, ZEUS internal note ZN-01-039 (2001) (unpublished).