Abstract

This note describes the HltHadAlleyMatchCalo tool that performs the confirmation of Level 0 objects in the High-Level Trigger hadronic alley, using information from the Vertex Locator (VeLo). VeLo tracks are matched with calorimeter candidates, allowing the trigger to select tracks that point to a high energy deposition.

Performance tests, matching resolution and track selection efficiencies are discussed. The way the tool is used in the HLT sequence is described.
## Contents

1 Introduction .......................................................... 2

2 Matching strategy ..................................................... 2

3 Matching 2D tracks .................................................... 3
   3.1 Tests with extrapolators ........................................... 4
   3.2 Kick correction ..................................................... 5
   3.3 $\chi^2$ distributions ............................................... 6

4 Matching 3D tracks .................................................... 7
   4.1 Tests with extrapolators ........................................... 7
   4.2 $\chi^2$ distributions ............................................... 7

5 Tool interface .......................................................... 7
   5.1 Tool functions ..................................................... 7
   5.2 Tool options ....................................................... 9

6 Performance ............................................................ 9
   6.1 Offline selection efficiency and retention ....................... 9
   6.2 Impact parameter cut ............................................... 11
   6.3 Acceptance cut .................................................... 11

7 Role of the Tool in the HLT Hadronic Alley ....................... 13
   7.1 Algorithmic sequence .............................................. 13
   7.2 Algorithm options ................................................ 14
   7.3 Timing ............................................................. 14

8 Conclusion ............................................................. 15
1 Introduction

In the 1 MHz readout scheme [1], the High-Level Trigger (HLT, [2]) consists in subtrigger alleys for muons, hadrons, photons and electrons. Each alley starts by confirming the Level-0 (L0) object that triggered it. This L0 confirmation didn’t exist before the 1 MHz readout scheme.

The L0 confirmation in the hadronic trigger is motivated by the fact that about 50% of L0 calorimeter clusters can’t be associated to a single, high-energy track [8]. This corresponds to cases where combinations of low-energy tracks, neutrons and \( K_L \) reach the same calorimeter cell and add up enough energy to trigger L0. Another reason is the bad energy resolution of HCAL clusters. The gain of the confirmation step is to reduce the number of trigger objects and thus the systematics.

This note describes the \texttt{HltHadAlleyMatchCalo} tool that performs the L0 confirmation in the High-Level Trigger hadronic alley. It presents the results shown at the Trigger and Reconstruction meeting in March and May 2006 [4, 5] in an updated and more detailed way.

The code of the tool has been taken from N. Tuning’s \texttt{MatchVeloL0Tool} [3], which was written in 2003. It had the aim of using the L0 information in order to improve the Level-1 trigger, which is historically referred to as Super-Level-1. Although this project is no longer part of the trigger, the hadronic part of the matching can now be used for the L0 confirmation.

2 Matching strategy

The energy deposition at the calorimeter can be used to estimate the amount of bending induced by the magnetic field on the track. By correcting the calorimeter position for this magnetic kick, one can estimate the slope of the track before the magnet, and compare it to the actual slope of the VeLo track. The difference is used to compute a \( \chi^2 \) criterion which is used for selecting tracks. Figure 1 shows an overview of this matching principle.

![Schematic drawing of the matching strategy](image)

Figure 1: Schematic drawing of the matching strategy (not to scale).

The L0 confirmation is done in two steps: 2D and 3D matching. The first step uses...
VeLo 2D tracks reconstructed by the **PatVeloRTracking** algorithm. These tracks have $r$-$\phi$ coordinates deduced from the 8 VeLo $r$-reading sectors, so that $\phi$ only takes 8 possible values (see figure 10 on page 12). The second step uses VeLo 3D tracks reconstructed by the **PatVeloSpaceTracking** algorithm. These tracks have a cartesian geometry.

Both confirmation functions take one track and try to match it with all the **L0CaloCandidates** that triggered L0 ($E_T > 3.5$ GeV). Only the smallest $\chi^2$ is returned. As there is a charge ambiguity, the matching is tried with negative and positive charge, so that each track is actually matched twice to each **L0CaloCandidate**. The charge that is assigned to a track by the tool is not used in the further reconstruction steps; in addition, the tool has no polarity information.

The following sections give a detailed description of these two confirmation steps.

## 3 Matching 2D tracks

The 2D confirmation uses VeLo tracks in $r$-$\phi$ coordinates, so that the calorimeter coordinates have to be translated to $r$-$\phi$ geometry.

Given a VeLo 2D track with slope $t_r = \frac{dr}{dz}$ and azimuthal angle $\phi$, and a **L0CaloCandidate** with position $x, y, z$ and transverse energy $E_T$, the $\chi^2$ is computed in the following way for charge $q = \pm 1$:

$$\chi^2 = \chi_r^2 + \chi_\phi^2$$

$$\chi_r^2 = \left( \frac{\Delta r}{\Delta z} \bigg| _{L0} - t_r \right)^2 \frac{1}{\sigma_r^2}$$

$$\chi_\phi^2 = \left( \phi_{L0} - \phi_{VeLo} \right)^2 \sigma_\phi^2$$

$$\frac{\Delta r}{\Delta z} \bigg| _{L0} = \sqrt{(x - qx_{kick})^2 + y^2} \quad z$$

$$\phi_{L0} = \arctan \left( \frac{y}{x - qx_{kick}} \right)$$ (1)

The kick correction uses a first-order approximation in $\frac{1}{E}$

$$x_{kick} = C_{kick} \frac{z - z_0}{E}$$

$$E = \sqrt{\frac{x^2 + y^2 + z^2}{x^2 + y^2} E_T}$$ (2)

where $C_{kick} = 1.263$ GeV and $z_0 = 5.25$ m is taken as the center of the magnetic field, using field 043. The errors $\sigma_r$ and $\sigma_\phi$ take into account the uncertainty on the calorimeter coordinates and energy, and the uncertainty on the $\phi$ value of the track. The uncertainty on the track slope measurement is small compared to the L0 measurement uncertainty and is therefore neglected.

$$\sigma_r^2 = \left( \sigma \left( \frac{\Delta x}{\Delta z} \right)^2 + \sigma \left( \frac{\Delta y}{\Delta z} \right)^2 \right) \left( \frac{\Delta r}{\Delta z} \bigg| _{L0} \right)^{-2}$$

$$\sigma_\phi^2 = \left( \sigma \left( \frac{\Delta x}{\Delta z} \right)^2 + \sigma \left( \frac{\Delta y}{\Delta z} \right)^2 \right) \left( \frac{\Delta r}{\Delta z} \bigg| _{L0} \right)^{-2}$$
\[
\begin{align*}
\sigma \left( \frac{\Delta x}{\Delta z} \right) &= \frac{1}{\frac{z}{\sigma^2_x + \sigma^2_{\text{kick}}}} \\
\sigma \left( \frac{\Delta y}{\Delta z} \right) &= \frac{1}{\frac{z}{\sigma_y}} \\
\sigma_x &= \sigma_y = \frac{4s}{\sqrt{12}} \\
\sigma_{\text{kick}} &= \frac{x_{\text{kick}}}{E} \\
\frac{\sigma_E}{E} &= 60\% \pm 70\% 
\end{align*}
\] (3)

\(s\) is the half-size of the corresponding cell, given by the \texttt{L0CaloCandidate::posTol()} function. There is a factor 4 because in L0 the cells are read out in clusters of 2 \(\cdot\) 2 cells. The energy error parameters\(^2\) are found by fitting

\[
\frac{(E_{\text{MC}} - E_{\text{Calo}})^2}{E_{\text{MC}}^2} \quad \text{vs} \quad \frac{1}{E_{\text{MC}}}
\]

with a straight line (see figure 3, right plot).

### 3.1 Tests with extrapolators

The code has been tested using information from the Monte Carlo truth. VeLo tracks are updated with the missing momentum information and extrapolated to the Calorimeter, taking into account the magnetic bending. The position of the extrapolated track is compared to the position of the cluster, showing that the matching resolution is dominated by the size of the Calorimeter clusters.

These tests were run using a stand-alone Gaudi application on 3000 \(B_d \rightarrow \pi^+\pi^-\) events, matching each track to \texttt{L0CaloCandidates} with \(E_T > 2.0\) GeV. Only correct matches are kept; a match is considered to be correct if the the VeLo track and the \texttt{L0CaloCandidate} are linked to the same MCParticle.

First the track is linked with its MCParticle and the state is updated with the \(q/p\) from the MCParticle. Then the track is extrapolated iteratively for 100 \(z\) values between the original position and the \(z\) of the matched \texttt{L0CaloCandidate}. The \texttt{TrackParabolicExtrapolator} tool is used for the extrapolation\(^3\). 2D tracks have to be converted to cartesian geometry before extrapolation.

The position where the extrapolated track reaches the HCAL can be compared to the position of the matched \texttt{L0CaloCandidate}, giving the matching resolution. The geometrical distance between the extrapolated track and the hit should be correlated to the \(\chi^2\) value (small distances for small \(\chi^2\)). This can be used for testing the validity of the matching procedure.

\(^{2}\)The \(\sigma_E\) parameterization shown in equation 3 was taken from the original code by N. Tuning. It is actually quite different from the one found in the Calorimeter technical design report; however it seems to work better for the matching. This requires further investigation, as \(\sigma_E\) is an important value for the \(\chi^2\) calculation.

\(^{3}\)attempts were made to use the \texttt{TrackMasterExtrapolator}, but this tool caused DaVinci to crash for some events.
Another test criterion is to compute a $\chi^2_{MC}$ in the same way as the tool does, but using the extrapolated track to compute the $x_{\text{kick}}$:

$$x_{\text{kick}} = |x_{\text{ex}} - z_{\text{ex}} t_x|$$

where ‘ex’ refers to the extrapolated track and $t_x$ is the slope of the original track (see figure 1 on page 2). $\chi^2_{MC}$ should be correlated to $\chi^2$.

Figure 2: Tests for 2D matching. Top left: distance between extrapolated track and Calo hit vs $\chi^2$; top right: $r$-projection of the distance vs $\chi^2$; bottom left: $\phi$-projection vs $\chi^2$; bottom right: $\chi^2$ correlation.

Figure 2 shows the test results for 2D matching. The resolution is rather bad, some good matches (small $\chi^2$) having a distance of 40 cm or more. This clearly comes from the difference in $\phi$, which shows a good correlation with $\chi^2$ but can be as large as 23 degrees. The difference in $r$ shows no correlation with $\chi^2$, due to the imprecision of the $x_{\text{kick}}$ correction. $\chi^2_{MC}$ is well correlated to $\chi^2$.

### 3.2 Kick correction

The effect of the magnetic field is approximated by an instant kick of the momentum vector at the center of the magnet (see [7], section 6.2.3). The center of the magnet is defined by a plane at $z = z_0$ where the integrated field equals half the total value. The kick correction is only applied in the $x$-direction.

The track extrapolation allows one to compute the $C_{\text{kick}}$ constant, using the $x_{\text{kick}}$ value from equation 4. For $E_T > 3.5$ GeV, this gives a mean value of $C_{\text{kick}} = 1.240$ GeV, and the mean value of the correction is 29 cm, which is much more than the matching resolution.
Figure 3 shows $x_{\text{kick}}$ computed from equation 4 as a function of $\frac{z-z_0}{E}$. A linear fit gives $C_{\text{kick}} = 1.130 \text{ GeV}$.

A more precise kick correction has been studied, using a second-order approximation in $x$ and in $y$. Figure 3 shows that the parabolic version of the $x$ kick correction isn’t very different from the linear one, all the more at high energies, while introducing an additional parameter.

The kick correction in the $y$ direction doesn’t significantly improve the matching resolution, as its mean amplitude only is 6 mm. Thus it was decided to only apply a linear correction in the $x$ direction, using just one parameter.

![Graph showing kick corrections and energy error](image1.png)

Figure 3: left: kick corrections in the $x$ direction, with linear and parabolic fit; right: energy error and linear fit

### 3.3 $\chi^2$ distributions

![Graph showing 2D $\chi^2$ distributions](image2.png)

Figure 4: 2D $\chi^2$ distributions for $B_d \rightarrow \pi^+\pi^-$ and minimum bias. In dotted red, all matches; in solid blue, only correct matches.

Figure 4 shows the $\chi^2$ distributions. Beyond $\chi^2 \sim 4$ the contribution from correct matches becomes negligible.
4 Matching 3D tracks

Given a VeLo 3D track with slopes $t_x = \frac{dx}{dz}$ and $t_y = \frac{dy}{dz}$, and a LOCalCandidate with position $x, y, z$ and transverse energy $E_T$, the $\chi^2$ is computed in the following way for charge $q = \pm 1$:

$$\chi^2 = \chi^2_x + \chi^2_y$$

$$\chi^2_x = \left( \frac{x - qx_{\text{kick}}}{z} - t_x \right) \frac{1}{\sigma^2_x}$$

$$\chi^2_y = \left( \frac{y - t_y}{z} \right) \frac{1}{\sigma^2_y}$$

(5)

The kick correction is the same as in the 2D matching. The errors $\sigma_x$ and $\sigma_y$ take into account the uncertainty on the calorimeter hit position and energy. Again, $s$ is the half-size of the corresponding calorimeter cell.

$$\sigma_x = \frac{1}{z} \sqrt{e^2_x + e^2_{\text{kick}}}$$

$$e_x = \frac{4s}{\sqrt{12}}$$

$$e_{\text{kick}} = x_{\text{kick}} \frac{\sigma_E}{E}$$

$$\sigma_y = \frac{1}{z} \frac{4s}{\sqrt{12}}$$

4.1 Tests with extrapolators

The same tests as for the 2D confirmation where performed. Figure 5 shows the results for 3D matching. The difference in $x$ shows no correlation with $\chi^2_x$, due to the imprecision of the $x_{\text{kick}}$ correction. The $y$ difference has a very good correlation to $\chi^2_y$ with the expected parabolic shape. The lower parabola, in red, corresponds to matches in the inner HCAL region\(^4\), with 13 $\times$ 13 cm\(^2\) cells, and the upper parabola, in blue, corresponds to the outer region, with 26 $\times$ 26 cm\(^2\) cells. In both cases, the maximal $y$ difference corresponds to the cell size. Both the calorimeter cell size and the kick correction contribute to the matching resolution. $\chi^2_{\text{MC}}$ is very well correlated to $\chi^2$.

4.2 $\chi^2$ distributions

Figure 6 shows the $\chi^2$ distributions. Beyond $\chi^2 \sim 8$ the contribution from correct matches becomes negligible.

5 Tool interface

All the L0 confirmation tools of the HLT share the same interface IHltMatchTrackCalo.

5.1 Tool functions

This is how to access the tool:

\(^4\)inner region means $|x| < 2000$ mm and $|y| < 2000$ mm
Figure 5: Tests for 3D matching. Top left: distance between extrapolated track and Calo hit as a function of \( \chi^2 \); top right: \( x \)-projection of the distance vs \( \chi^2 \); bottom left: \( y \)-projection vs \( \chi^2 \); red is for the HCAL inner region and blue for the outer region. Bottom right: \( \chi^2 \) correlation.

```cpp
#include "HltInterfaces/IHltMatchTrackCalo.h"
IHltMatchTrackCalo* m_matchL0Hadron =
tool<IHltMatchTrackCalo>("HltHadAlleyMatchCalo");

There is a function to tell the tool which L0CaloCandidates to use, requiring candidates with \( E_T > \) \( \) etCut. The function returns the number of candidates found.

```cpp
double etCut = 3.5*GeV;
int nCandidates;
nCandidates = m_matchL0Hadron->getCandidates(etCut);
```

The confirmation functions take a track and return the best \( \chi^2 \):

```cpp
Track track;
double chi2;
chi2 = m_matchL0Hadron->confirmation2D(track);
chi2 = m_matchL0Hadron->confirmation3D(track);
```

There is a function that returns a vector of pairs (\( \chi^2 \), L0CaloCandidate) for all matches with \( \chi^2 < \) \( 2D, 3D \):

```cpp
std::vector< std::pair<double, L0CaloCandidate*> >
matchedPairs = m_matchL0Hadron->matchedCandidates();
```
This function should be used after 2D or 3D confirmation to see which L0CaloCandidates have been matched with the track, since sometimes there is more than one, and the right one is not necessarily the one with the smallest $\chi^2$.

5.2 Tool options

The following properties can be modified in an option file:

- \texttt{ptkickConstant = 1.263*GeV} constant for kick correction
- \texttt{ptkickZ0 = 5250.0*mm} $z_0$ for kick correction
- \texttt{caloCandidatesName = "Trig/L0/FullCalo"} location of L0CaloCandidates in the Transient Event Store
- \texttt{chi2Cut2D = 7.0} \texttt{chi2Cut3D = 7.0} $\chi^2$ limits for the return values of the \texttt{matchedCandidates()} function

6 Performance

The analysis described in the next paragraphs is performed by a stand-alone Gaudi application running on 3212 $B_d \rightarrow \pi^+\pi^-$ events, 1376 $B_s \rightarrow D_sK$ events and 6000 minimum bias events (of which 4179 are L0-hadron triggered). All events are in RTTC format with a luminosity of $2 \cdot 10^{32}$ cm$^{-2}$s$^{-1}$. The signal events are offline selected events. 3D efficiencies are computed after the 2D confirmation, which is used as a filter, with a loose $\chi^2$ cut at 7.0.

6.1 Offline selection efficiency and retention

Figure 7 shows the signal and minimum bias retention and purity. An event is rejected if there is no track that passes the $\chi^2$ cut; the purity is defined as the fraction of correct matches normalized to all matches. It should be noticed that these results are very similar to those obtained by N. Tuning [3].

Figure 8 shows the average number of tracks per event, after 2D and 3D matching. For minimum bias events, there are in average 70.7 forward tracks per event before 2D matching. Cutting at $\chi^2 < 7$, there are 7.2 tracks left after 2D confirmation, and 3.4 after 3D confirmation. Thus the total L0 confirmation track retention rate is 4.8% (including
Figure 7: Signal and minimum bias retention and purity as a function of the $\chi^2$ cut.
VeLo space tracking efficiency and acceptance cut, see paragraph 6.3). This rate can be further improved by applying an impact parameter cut, as is discussed in the next paragraph. Table 1 shows a summary of the efficiencies for both cuts set to 7.0.

<table>
<thead>
<tr>
<th>Channel</th>
<th>Retention 2D</th>
<th>Retention 3D</th>
<th>Tracks 2D</th>
<th>Tracks 3D</th>
<th>Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>$B_d \rightarrow \pi^+\pi^-$</td>
<td>99.9%</td>
<td>95.7%</td>
<td>6.0</td>
<td>3.0</td>
<td>3212</td>
</tr>
<tr>
<td>$B_s \rightarrow D_s K$</td>
<td>99.7%</td>
<td>92.6%</td>
<td>7.8</td>
<td>4.0</td>
<td>1376</td>
</tr>
<tr>
<td>MinBias</td>
<td>98.2%</td>
<td>78.9%</td>
<td>7.2</td>
<td>3.4</td>
<td>6000</td>
</tr>
</tbody>
</table>

Table 1: Summary of 2D and 3D efficiencies, with both $\chi^2$ cuts set to 7.0

### 6.2 Impact parameter cut

It is possible to further reduce the number of tracks by applying an impact parameter (IP) cut before the 2D confirmation. A negative IP is assigned to tracks coming from a secondary vertex which is behind the primary vertex; thus 50% of background tracks have a negative IP, while this is rare for signal tracks (but possible, due to the IP resolution or vertexing errors, or because the B decays away from the beam-line; in the determination of the sign of the IP it is assumed that the B travels exactly along the beam-line). Cutting out these negative IP tracks reduces the $B_d \rightarrow \pi^+\pi^-$ selection efficiency by 3%, but reduces the number of tracks per minimum bias event to 3.7 instead of 7.2 (see figure 9; only the 2D confirmation efficiencies are considered). These 3% signal events would be lost anyway further down the alley because of timing constraints.

Applying a tighter IP cut (IP > 0.05 or 0.1 mm) saves even more tracks but also degrades the signal retention rate.

### 6.3 Acceptance cut

The VeLo has an acceptance of $\theta < 390$ mrad for all $\phi$, in order to cover the LHCb acceptance ($0.25^2 + 0.3^2 \approx 0.39^2$). This means that a fraction of the 3D reconstructed tracks will be outside the LHCb acceptance (see figure 10). Although these tracks might be useful for vertices, isolation etc., they should not be used for HCAL matching.
Figure 9: 2D confirmation event and track retention rates for various impact parameter cuts.

For minimum bias, without applying the 2D confirmation, the fraction of 3D input tracks that are outside the acceptance is about 10%. Applying the 2D confirmation, this fraction goes down to 4%, while only 0.23% of the correct 3D matches come from tracks outside the acceptance. Thus the 2D confirmation gets rid of most out-of-acceptance tracks. Still, it is reasonable to apply the acceptance cut:

\[
t_x = \tan(\theta_x) < \tan(0.3) \\
t_y = \tan(\theta_y) < \tan(0.25)
\]

where \( t_x \) and \( t_y \) are the track slopes.

Figure 10: Polar angle distributions of 2D and 3D input tracks. The circle shows the VeLo acceptance, while the rectangle represents the LHCb acceptance.
7 Role of the Tool in the HLT Hadronic Alley

The HLT hadronic alley [8] consists of four boxes, each box being a sequencer of algorithms. The L0 confirmation is the second box; it is preceded by the entry point, which takes the decision to enter the alley, and followed by the pre-trigger (old Level-1) and the trigger (old HLT-generic).

This section shows how the matching tool can be used in the L0 confirmation box, and takes a first look at timing.

In addition to the VeLo-Cal0 matching described in this note, the L0 confirmation could be completed by also matching with TT, but this is still under study.

7.1 Algorithmic sequence

Two algorithms call the tool to perform the L0 confirmation, as illustrated in figure 11. Their basic task is to select the tracks based on a $\chi^2$ criterion. Selected tracks are flagged, and can also be copied to a separate location in the PatDataStore. The procedures of these algorithms are very similar:

- retrieve the 2D or 3D tracks from the PatDataStore
- tell the tool to use the L0Cal0Candidates that triggered L0
- filter backward 2D tracks
- apply the impact parameter cut for 2D tracks
- apply the acceptance cut for 3D tracks
- for each track, get the smallest $\chi^2$ from the tool
- if the track passes the $\chi^2$-cut, set a flag and clone the track in another path in the PatDataStore for further reconstruction

Figure 11: Design of the L0 Confirmation box
- filter the events, rejecting events where no track passes the $\chi^2$-cut
- write a HltSummaryBox including the event filter decision, the number of selected tracks and the smallest $\chi^2$.

Since both algorithms are very similar, and in order to avoid code duplication as much as possible, the 3D algorithm inherits the common functions from the 2D algorithm. This also allows for an easier maintenance. Note that this is not done in the tool, as there is very little duplication in the tool code.

### 7.2 Algorithm options

Both algorithms have the same set of options:

- **chi2Cut** $\chi^2$ cut for track selection. This doesn’t have to be the same value as in the tool options.
- **L0EtCut = 3500.*MeV** minimal $E_T$ of the L0CaloCandidates that should be used for matching
- **CopyTracks = true** if set to false, do not copy the selected tracks to OutputTracksName, just flag them
- **InputTracksName** location of input tracks in the PatDataStore
- **OutputTracksName** location of output tracks in the PatDataStore
- **HltSummaryLocation** location of the HLT summary in the TES
- **UseIPCut = true** if set to false, do not apply the impact parameter cut (only for 2D confirmation). The cut value has to be set using the IPMin property of the HltHadTrack2DIPSelection algorithm.

### 7.3 Timing

Table 2 shows the timing for the L0 Confirmation box, running 10'000 minimum bias events on a machine that is about 0.98 times faster than a 2.8 GHz Xeon. The timing has been performed using the standard SequencerTimerTool.

More than half the time is spent for the tracking, while the confirmation algorithms only take about 10% of the time each.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>ms/event</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>HltHadL0Confirmation</td>
<td>4.580</td>
<td>100 %</td>
</tr>
<tr>
<td>PatInitEvent</td>
<td>0.276</td>
<td>6.0 %</td>
</tr>
<tr>
<td>CreateRawEvent</td>
<td>0.140</td>
<td>3.0 %</td>
</tr>
<tr>
<td>PatVeloDecodeRaw</td>
<td>0.412</td>
<td>9.0 %</td>
</tr>
<tr>
<td>PatVeloRTracking</td>
<td>1.344</td>
<td>29.3 %</td>
</tr>
<tr>
<td>PatPV2D</td>
<td>0.203</td>
<td>4.4 %</td>
</tr>
<tr>
<td>HltHadTrack2DIPSelection</td>
<td>0.167</td>
<td>3.6 %</td>
</tr>
<tr>
<td>HadL0Conf2DDecision</td>
<td>0.526</td>
<td>11.5 %</td>
</tr>
<tr>
<td>HltHadVeloSpacePartial</td>
<td>1.659</td>
<td>24.9 %</td>
</tr>
<tr>
<td>HadL0Conf3DDecision</td>
<td>0.477</td>
<td>7.1 %</td>
</tr>
</tbody>
</table>

Table 2: L0 Confirmation box timing. Time spent in each algorithm, in milliseconds per event or as a fraction of the total time spent in the box.
8 Conclusion

The L0 confirmation code for VeLo to HCAL is available in the form of a tool and two algorithms. The code compiles and runs with RTTC data, which is a major step towards DC06. Hence the migration to DC06 should only imply minor changes, and an easy transition can be expected.

Code, documentation, presentations and other material concerning the subject of this note are available on a dedicated web page [9].

The resolution of the matching could be greatly improved by applying the s-curve correction on the calorimeter clusters [6], which improves the position of the L0Calo-Candidate. This correction method is still under investigation.

The selection efficiencies and track retention rates that have been studied show that the L0 confirmation is a very powerful way of selecting tracks in the hadronic alley.

References

[1] T. Schietinger, High-Level Trigger Strategy at 1 MHz, LHCb Collaboration Meeting, CERN, 30th November 2005


