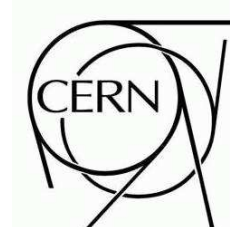




ATLAS NOTE



Monte Carlo Performance of the TileCal Low p_T Muon Identification Algorithm

A. Ruiz-Martínez¹, H. Lim², G. Usai³, J. Abdallah¹, V. Castillo¹,
C. Cuenca¹, A. Ferrer¹, E. Fullana², V. González⁴, E. Higón¹, J. Poveda¹, B. Salvachúa²,
E. Sanchís⁴, C. Solans¹, J. Torres⁴, A. Valero¹ and J. A. Valls¹

ATL-TILECAL-PUB-2008-008
07 July 2008



Abstract

This note describes the TileCal standalone low p_T muon identification algorithm (TileMuId) developed to contribute to the Level-2 trigger. This algorithm is based on the characteristic muon energy deposition inside the calorimeter. The implementation of this algorithm in the core of the Digital Signal Processors (DSPs) in the TileCal Read-Out Drivers (RODs) is also discussed in this paper.

The TileMuId performance with Monte Carlo data from single muons and $b\bar{b}$ events is shown in terms of efficiencies and fraction of fakes for both a fully Level-2 version and a ROD-based version of the algorithm.

¹Departament de Física Atòmica, Molecular i Nuclear and IFIC, CSIC-Universitat de València, Spain

²Argonne National Laboratory, USA

³University of Chicago, USA

⁴Departament de Ingenieria Electrònica, Universitat de València, Spain

Contents

1	Introduction	3
2	Algorithm description	5
3	TileMuId implementation in the ROD DSP	6
3.1	Output ROD data format description	7
4	Performance with Monte Carlo data	11
4.1	Spatial resolution	11
4.2	Efficiency	11
4.3	Fraction of fakes	13
4.4	Summary	13
5	Conclusions	17
	Acknowledgments	17

1 Introduction

This note is devoted to the performance of the TileCal low p_T muon identification algorithm (TileMuId) [1, 2] using Monte Carlo samples of single muon and $b\bar{b} \rightarrow \mu(6)X$ events. This algorithm makes use of the energy deposited in TileCal looking for a deposition pattern compatible with a minimum ionizing particle. The aim of such an algorithm is to build a Level-2 trigger in combination with other subdetectors, especially meant to recover the very soft muons not triggered by the muon spectrometer standalone. In fact, as the ATLAS muon spectrometer is only efficient for reconstructing muons with $p_T > 5$ GeV, TileCal can add lower p_T muons using this algorithm. In addition, a new version of this algorithm which has been implemented in the TileCal Read-Out Driver (ROD) [3] is presented in this note and its results are compared with the existing version fully executed at Level-2. Furthermore, the recent implementation of the algorithm in the TileCal RODs will consume less processing time at Level-2 as all the processing has been done at the ROD level.

For instance, the low p_T di-muon trigger efficiency for the selection of B-physics events can be enhanced with the TileCal information provided by this algorithm. In first place, the Region of Interest (RoI) defined by the muons found by Level-1 in the trigger chambers (with $p_T \geq 6$ GeV) will be confirmed at Level-2 with the muon precision chambers in order to reduce the rate at this stage. After that, TileMuId can make a full scan in the calorimeter to detect a second muon with lower p_T and open a secondary Region of Interest (sRoI), which can be combined with other subdetectors.

In particular, new studies to match the muon energy depositions in TileCal with the tracks found in the Inner Detector (using the algorithm TrigIDSCAN [4]) are currently carried out [5]. This way, the rate from muons identified in TileCal coming from π/K decays can be reduced by matching the corresponding muon track in the Inner Detector. Furthermore, the muon p_T is measured in the tracking system and therefore the muon p_T threshold can be selected as required for the B-physics analysis.

To recover the second muon that has not been triggered by Level-1, a complete search in the calorimeter has to be performed, hence the Level-2 trigger algorithm has to be fast enough to fulfill the tight timing requirements (below ~ 10 ms). Little timing consumption is crucial to make feasible the TileCal contribution on the low p_T muon triggers. The muon identification processing over all calorimeter wedges performed in parallel at the ROD level allows a fast execution at Level-2. The implementation of the muon tagging algorithm in the ROD is discussed in detail in Section 3.

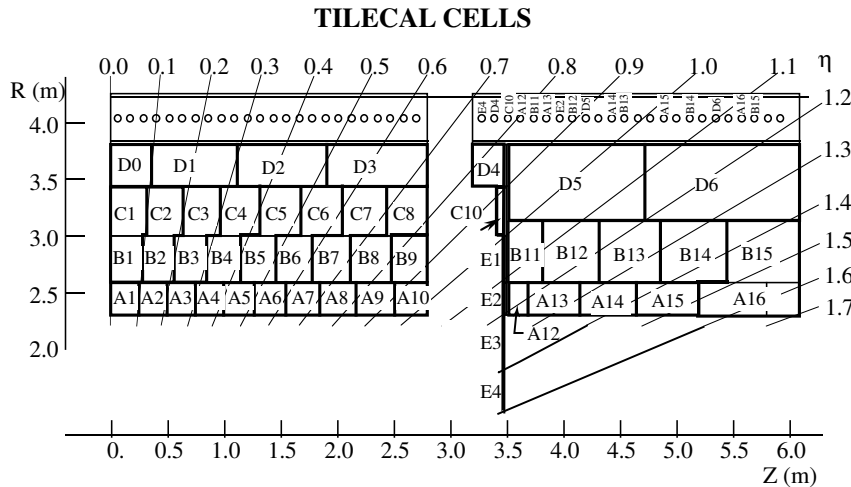


Figure 1: Diagram of the TileCal cells.

TileCal is a sampling calorimeter using iron as the passive material and scintillating tiles as the active medium. It is designed as one Long Barrel (LB) and two Extended Barrel (EB) parts, covering the $|\eta| < 1.7$ region. Each barrel is divided in 64 modules in the ϕ direction. Figure 1 shows the cell structure of the calorimeter for read-out, with 3 longitudinal layers (known as A, BC and D from the innermost to the outermost one, respectively) and η -projective towers. Each cell is read out by two photomultiplier tubes (PMTs). All front-end electronics associated to a TileCal module are placed in compact structures called drawers, located inside the back-beam region of the modules. Two physical drawers are coupled from the electronics point of view forming a new structure called superdrawer. In total, there are 256 superdrawers in TileCal, one per each half barrel module and one per each extended barrel module. From the data acquisition point of view, TileCal is divided in 4 partitions (LBA, LBC, EBA and EBC) each with independent trigger and busy logic.

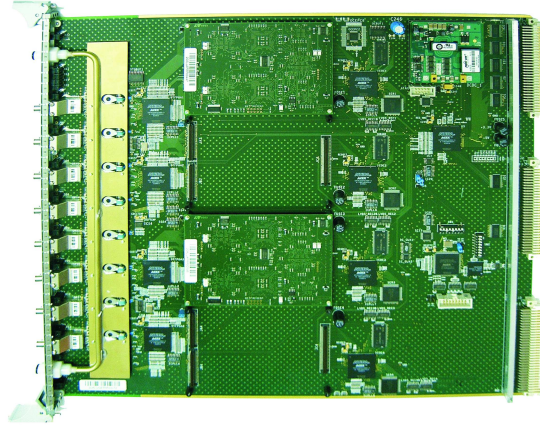
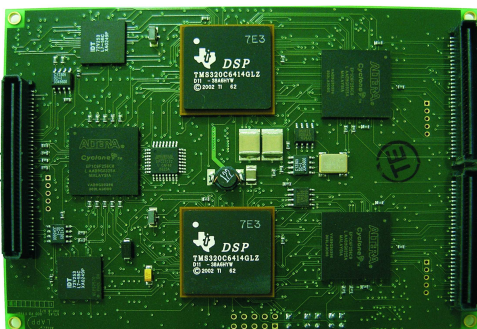
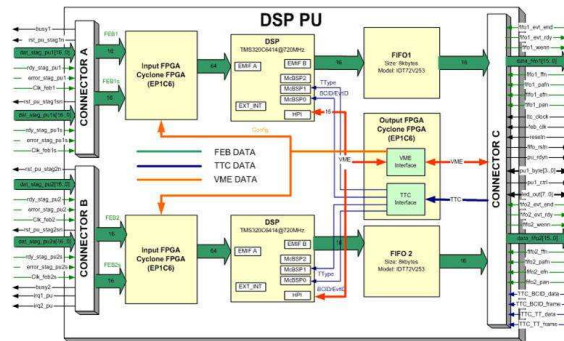


Figure 2: Picture of the TileCal ROD motherboard. Note that there are only 2 PUs in the board, as needed in the default operation mode.



(a) ROD DSP PU picture. Note that the front-end input data come from right to left.



(b) ROD DSP PU scheme. Note the arrows according to the data flow.

Figure 3: ROD DSP PU.

The ROD system is the central element of the TileCal back-end electronics. The ROD is a 9U VME module which can read up to 8 optical fibers from the front-end electronics with the information of a

single superdrawer each. It can have up to 4 mezzanine cards, called Processing Units (PUs), to process the incoming raw data in real time before sending it to the next step in the data acquisition chain. In the current setup, two PUs are placed in the motherboard slots #1 and #3 to process the data. Figure 2 shows a picture of a ROD module equipped with two PUs.

The Digital Signal Processor (DSP) PU [6] is composed of two blocks, each one with an input FPGA Cyclone EP1C6, a TMS320C6414 DSP from Texas Instruments and an external output FIFO. The DSP PU also contains an output FPGA Cyclone EP1C6 used for the VME and TTC [7] interface. A picture of this device and its layout are shown in Figure 3. The input FPGAs and the DSPs can be programmed by uploading the corresponding code through the VME interface. The main functions of the DSP PU are the data flow management, data formatting, TTC reception, buffering and timing synchronization, data processing with online algorithms, online histogramming and error detection.

2 Algorithm description

The aim of this low p_T muon identification algorithm, TileMuId, is to exploit the TileCal capability for detecting muons in the low p_T range as a complement to the muon spectrometer. The basics of the algorithm are to search for muons taking into account the energy deposited in each layer of TileCal and taking advantage of Tile’s segmentation. In order to identify the muons, the energy deposition in each cell is delimited by a higher and a lower thresholds:

$$\text{Thr}_i^{\text{low}} \leq E_i \leq \text{Thr}_i^{\text{high}} \quad i = 1, 2, 3. \quad (1)$$

If this condition is fulfilled in each of the 3 layers in a projective pattern in η , the muon is tagged. This procedure is known as “tight selection”. In order to be efficient on events in which the muon loses a considerable fraction of its energy in one of the layers, muons are also tagged if Eq. (1) is fulfilled in two of the layers and in the other the energy deposition is greater than the higher threshold. This approach is known as “loose selection”.

The lower energy threshold is meant to cut the electronic noise and minimum bias pileup events. In the current version of this algorithm, all cells have the same value for the lower energy threshold, which is taken as $\sim 2\sigma$ of the electronic noise distribution. Although a single value for the lower threshold is used for all cells in the framework of the present note, it can adopt different values for each individual cell. Once the noise level of all the TileCal channels is known in real ATLAS working conditions, the lower threshold for a given cell will be assigned to the quadratic sum of the noise of the corresponding two PMTs in this cell. Anyhow, it can be tuned according to the noise environment at different luminosities or the trigger performance. For instance, by raising this threshold the trigger rate is reduced, because the number of fakes decreases but efficiency decreases as well.

The probability of uncorrelated electronic noise being flagged as a muon is the product of the probability of noise above the lower energy threshold in the three calorimeter layers. With a 2σ threshold level, the estimated probability of fake muons in a tower due to electronic noise is $\sim 1.2 \times 10^{-5}$ (which considering the whole calorimeter leads to a $\sim 4\%$ probability that a fake muon is found due to noise fluctuations in a given event).

The higher energy thresholds are meant to delimit the maximum muon energy deposition while eliminating hadronic showers and tails. These thresholds are determined for each individual cell depending on the pseudorapidity of its trajectory. As the muon goes through different amount of material (iron and scintillating tiles) its energy deposition pattern in the TileCal cells changes. The values of the higher energy thresholds for all cells are summarized in Tables 1 and 2.

Table 1: Higher energy thresholds for LB cells. Note that the $\pm N$ references to the N -th η tower.

A cells	Thr (GeV)	BC cells	Thr (GeV)	D cells	Thr (GeV)
A \pm 1	1.35	BC \pm 1	2.55	D0	1.59
A \pm 2	0.96	BC \pm 2	1.98	D \pm 1	1.11
A \pm 3	0.87	BC \pm 3	1.77	D \pm 2	1.05
A \pm 4	0.75	BC \pm 4	1.89	D \pm 3	1.05
A \pm 5	0.84	BC \pm 5	1.86		
A \pm 6	0.84	BC \pm 6	1.80		
A \pm 7	0.93	BC \pm 7	1.89		
A \pm 8	0.84	BC \pm 8	1.86		
A \pm 9	0.96	B \pm 9	1.11		
A \pm 8	0.90				

Table 2: Higher energy thresholds for EB cells. Note that the $\pm N$ references to the N -th η tower.

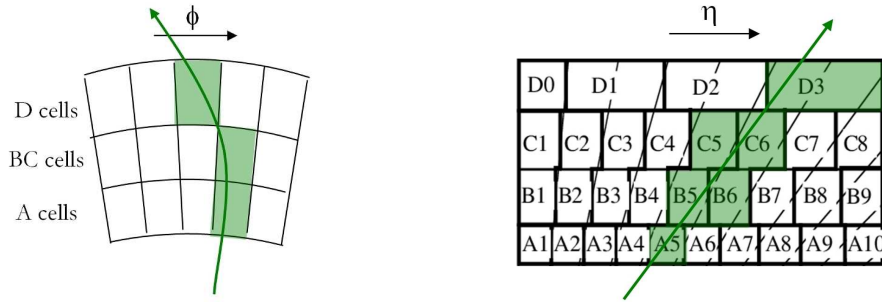
A cells	Thr (GeV)	B cells	Thr (GeV)	D cells	Thr (GeV)
A \pm 12	0.81	C \pm 10	0.39	D \pm 4	0.90
A \pm 13	1.14	B \pm 11	1.05	D \pm 5	2.10
A \pm 14	1.23	B \pm 12	1.38	D \pm 6	2.37
A \pm 15	1.38	B \pm 13	1.44		
A \pm 16	1.68	B \pm 14	1.50		
		B \pm 15	1.68		

3 TileMulD implementation in the ROD DSP

Taking advantage of the high computing capabilities of the ROD DSPs, the TileMulD algorithm has been implemented inside these processors. This way, the algorithm is executed for all the events accepted by Level-1 trigger and its results are encoded in the output data format in a dedicated fragment. Specific software developed inside the Level-2 trigger retrieves the information in those fragments to use it in the High Level Trigger (HLT) environment. Although the implementation of the algorithm in the DSP introduces some limitations, its main advantage is the dramatic reduction in the computing time required at Level-2.

Due to the fact that each DSP processes only the information from 2 superdrawers sequentially, the main limitation of the ROD-based approach of the algorithm is that in principle it will only be possible to execute it separately in each single superdrawer. Also, with the TileCal geometry shown in Figure 1, some towers are split between long and extended barrel. Hence, it is not possible to apply the ROD-based version of the algorithm in these towers.

With the TileMulD algorithm definition and its implementation in the ROD DSP, some problems can appear due to the effect of the ATLAS magnetic field in the muon trajectory and the non-ideal projective geometry of TileCal. In the first case, the magnetic field can bend the trajectory of the muon so that it goes through two TileCal modules, as shown in Figure 4(a). Then the muon cannot be tagged at the DSP level. In the second case, the particle can deposit energy in more than one calorimeter tower, as shown in Figure 4(b), which can lead to a double tag of the same muon. A protection against tagging the same muon twice in two contiguous towers is implemented in the code.



(a) Muon bent by the magnetic field going through several modules.

(b) Muon path going through several towers in η .

Figure 4: Example of muon trajectories which may lead to problems in the TileMuId algorithm.

3.1 Output ROD data format description

After the algorithm processing inside the ROD DSP, their output is formatted according to the requirements described in the ATLAS data format [8] and sent to the next step in the data acquisition chain. Each DSP PU builds a ROD fragment which is formed by several subfragments containing different types of data and framed by a header and a trailer blocks. The subfragments present in a ROD fragment depend on the run type, in particular the output from the muon tagging algorithm is present by default for Physics runs. Each ROD fragment contains the information from up to 4 superdrawers.

Table 3 shows the TileCal ROD fragment [9] used during the detector commissioning phase for cosmic runs. As shown, the present subfragments contain the raw data (Digitizer fragments), the online reconstructed energy (Reco fragments) [10], data quality checks (Status fragments) and the output of the trigger-oriented algorithms for Level-2 (L2 fragments). These trigger-oriented algorithms are TileMuId, explained in this note, and the computation of the total transverse energy per superdrawer. This latter algorithm computes the transverse energy per TileCal superdrawer inside the ROD DSP and sends these data in the byte stream to the Level-2 trigger. At this stage, the transverse energy in the event would be computed with the information from both Tile and LAr calorimeters and used to build a missing E_T Level-2 trigger. All the subfragments contain the information corresponding to a single superdrawer, except the L2 fragments which hold the data from the 2 superdrawers processed by the same DSP to allow a faster data access in HLT environment.

In the default DSP operation mode, called staging mode, the data coming from 2 superdrawers are processed by a single DSP. There is also the possibility to work in the so-called full mode, where the data from only one superdrawer are processed by a single DSP. Therefore, depending on whether both trigger-oriented algorithms are selected to be executed or just one of them and the ROD operation mode (staging or full), there are 4 different types of L2 fragments where TileMuId is executed:

- Fragment type 0x10: TileMuId and transverse energy computation enabled, staging mode.
- Fragment type 0x11: TileMuId and transverse energy computation enabled, full mode.
- Fragment type 0x12: TileMuId enabled, staging mode.
- Fragment type 0x13: TileMuId enabled, full mode.

These fragments will only be present in the output data if the corresponding algorithms are enabled to be executed online. They are variable-size fragments sending two 32-bit words per muon found. If the

transverse energy computation is also enabled, an additional word per superdrawer with this information is added before the muon data words. Note that the fragments which only store the information from the muon tagging algorithm (0x12 and 0x13) are not present in the byte stream (not even their header) unless muons are actually found. Hence, if one muon is found in the superdrawers read out by one DSP, the fragment will contain 5 words. Table 4 shows the structure of one of those fragments.

Table 3: TileCal ROD fragment data format during commissioning for Physics runs.

Header
Reco fragment (type 0x4) #1
Digitizer fragment (type 0x0) #1
Status fragment (type 0xa) #1
Reco fragment (type 0x4) #2
Digitizer fragment (type 0x0) #2
Status fragment (type 0xa) #2
L2 fragment (type 0x12) #1 #2
Reco fragment (type 0x4) #3
Digitizer fragment (type 0x0) #3
Status fragment (type 0xa) #3
Reco fragment (type 0x4) #4
Digitizer fragment (type 0x0) #4
Status fragment (type 0xa) #4
L2 fragment (type 0x12) #3 #4
Trailer

Table 4: Description of a L2 fragment (only TileMuId enabled).

Fragment header (0xff1234ff)
Fragment size ($3+2 \times n$)
Fragment identifier
Muon 1, data word 1
Muon 1, data word 2
...
Muon n , data word 1
Muon n , data word 2

Tables 5 and 6 show the bit field for the fragment identifier word for staging mode and full mode fragments respectively. In both cases the 16 most significant bits (MSB) are dedicated to encode the fragment type (0x10 to 0x13) and the 16 least significant bits (LSB) are used for the TileCal partition (1 for LBA, 2 for LBC, 3 for EBA and 4 for EBC) and the 1 or 2 superdrawers processed. Note that in staging mode, the 6 LSB are used for the first superdrawer and the next 6 bits for the second superdrawer processed by the same DSP.

Table 5: Bit field for the fragment identifier 32-bit word for staging mode (0x10 and 0x12).

Fragment type	Partition	Drawer #2	Drawer #1
bits 31:16	bits 15:12	bits 11:6	bits 5:0

Table 6: Bit field for the fragment identifier 32-bit word for full mode (0x11 and 0x13).

Fragment type	Partition	Drawer
bits 31:16	bits 15:8	bits 7:0

Table 7: Description of the first muon data word.

QF	D	Muon pattern	Energy in 3rd layer
bit 31	bit 30	bits 29:25	bits 24:0

Table 8: Description of the second muon data word.

Energy in 2nd layer	Energy in 1st layer
bits 31:16	bits 15:0

The bit fields for the two 32-bit muon data words are shown in detail in Tables 7 and 8. The variables encoded in the muon data words are the following:

- **Quality factor (QF):** this variable is set to 0 if the muon has been tagged following the tight selection criteria and is set to 1 if tagged following the loose selection criteria, as defined in Section 2. The most significant bit in the first muon data word is used to encode this quality factor in the output data format.
- **Superdrawer (D):** since each DSP processes 2 superdrawers, one bit in the muon data words is used to identify in which superdrawer the muon has been found. This bit is set to 0 if the muon has been tagged in the first superdrawer processed by the DSP (i.e., in the Drawer #1 specified in the fragment identifier, see Table 5) or it is set to 1 if found in the second superdrawer (Drawer #2 in the fragment identifier, see Table 5).
- **Muon pattern:** taking into account the geometrical segmentation of TileCal, a muon produced at the interaction point can go through the calorimeter following several trajectories. Furthermore, as the projectivity in the cell segmentation is not ideal, cells in different towers can be hit by the same muon. Therefore, we define different projective patterns as a combination of 3 cells, one per layer, as shown in Tables 9 and 10. The pattern in which the muon has been found is encoded using 5 bits in the first muon data word. During the fragment decoding, the muon η coordinate is

computed as the average of the η coordinates from the 3 cells in the pattern. This way, knowing the superdrawer and the muon pattern, the (η, ϕ) position of the muon is perfectly determined.

- **Energy deposited by the muon in each layer:** in addition, this fragment contains the energy released in all the 3 cells used to tag the muon using 25 bits in the first muon data word and the whole second muon data word. The energy stored in those words is expressed in MeV and scaled by a factor 2 in order to have a precision of 0.5 MeV using a fixed-point processor.

Table 9: Projective patterns defined in the LB superdrawers. Note that the $\pm N$ references to the N -th η tower.

Muon pattern	D cell	BC cell	A cell	$\langle \eta \rangle$
0	D0	BC \pm 1	A \pm 1	0.033
1	D \pm 1	BC \pm 2	A \pm 2	0.167
2	D \pm 1	BC \pm 3	A \pm 3	0.233
3	D \pm 2	BC \pm 4	A \pm 4	0.367
4	D \pm 2	BC \pm 5	A \pm 5	0.433
5	D \pm 2	BC \pm 6	A \pm 6	0.500
6	D \pm 3	BC \pm 6	A \pm 6	0.567
7	D \pm 3	BC \pm 7	A \pm 7	0.633
8	D \pm 3	BC \pm 8	A \pm 8	0.700

Table 10: Projective patterns defined in the EB superdrawers. Note that the $\pm N$ references to the N -th η tower.

Muon pattern	D cell	BC cell	A cell	$\langle \eta \rangle$
0	D \pm 5	B \pm 11	A \pm 12	1.067
1	D \pm 5	B \pm 12	A \pm 12	1.100
2	D \pm 5	B \pm 12	A \pm 13	1.133
3	D \pm 5	B \pm 13	A \pm 12	1.133
4	D \pm 5	B \pm 13	A \pm 13	1.167
5	D \pm 5	B \pm 13	A \pm 14	1.200
6	D \pm 6	B \pm 11	A \pm 12	1.133
7	D \pm 6	B \pm 12	A \pm 12	1.167
8	D \pm 6	B \pm 12	A \pm 13	1.200
9	D \pm 6	B \pm 13	A \pm 12	1.200
10	D \pm 6	B \pm 13	A \pm 13	1.233
11	D \pm 6	B \pm 13	A \pm 14	1.267
12	D \pm 6	B \pm 14	A \pm 13	1.267
13	D \pm 6	B \pm 14	A \pm 14	1.300
14	D \pm 6	B \pm 14	A \pm 15	1.333
15	D \pm 6	B \pm 15	A \pm 14	1.333
16	D \pm 6	B \pm 15	A \pm 15	1.367

4 Performance with Monte Carlo data

Currently two different implementations of the TileMuId algorithm are used within the ATLAS collaboration. One of them, known as TrigTileLookForMuAlg, is fully executed at Level-2 and the second implementation, TrigTileRODMuAlg, executes the TileMuId algorithm inside the ROD DSPs and its results are accessed at Level-2.

The performance of the TileMuId algorithm in its two implementations has been studied with single muon samples and the inclusive $b\bar{b} \rightarrow \mu(6)X$ process, where there is at least one muon with $p_T > 6$ GeV in the final state.

The geometrical acceptance of the muon tagging algorithm is limited by the coverage of the TileCal third layer, $|\eta| = 1.4$. For this reason, in this analysis only generated muons with $|\eta| < 1.4$ are considered.

4.1 Spatial resolution

The algorithm spatial resolution has been defined using a sample of single muon events with $2 \text{ GeV} \leq p_T \leq 15 \text{ GeV}$. In order to compare the reconstructed muon ϕ coordinates with the information from the Monte Carlo truth, which is defined at the vertex, the latter is extrapolated at the TileCal radius using the following parametrization:

$$\phi_{\text{Tile}}(\mu^\pm) = \phi_{\text{Truth}}(\mu^\pm) \mp 0.000123 \mp \frac{0.507}{p_T(\mu^\pm)} \quad (2)$$

where p_T is expressed in GeV and ϕ in rad. This parametrization is extracted using single muon events at different transverse momenta. The numerical factors in Eq. (2) are obtained from a fit of $(\phi_{\text{Tile}} - \phi_{\text{Truth}})$ as a function of the muon p_T . The average shift with p_T in the distribution of residuals $(\phi_{\text{Tile}} - \phi_{\text{Truth}})$ is thus cancelled.

The final distributions of residuals, $(\eta_{\text{Tile}} - \eta_{\text{Truth}})$ and $(\phi_{\text{Tile}} - \phi_{\text{Truth}})$, for muons with different momenta are shown in Figure 5 for both algorithms, TrigTileLookForMuAlg and TrigTileRODMuAlg. These distributions are used to define the spatial resolution of the algorithm.

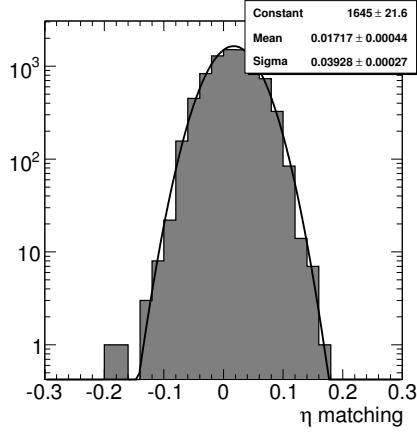
Note that the η distribution for TrigTileLookForMuAlg is biased toward positive values due to an unexpected feature of the algorithm, that results in an asymmetry between the positive and negative side. The muons tagged at $|\eta| \simeq 1.4$ are split among two search paths due to the coarse granularity of the detector and the lack of projectivity in the segmentation. Since the direction in the detector scan is fixed from negative to positive η , we pick up mostly one rather than the other of the two different (but equivalent, since the splitting) paths in the two detector sides. This feature is being corrected for the new releases.

The distributions are fitted with a Gaussian to define an acceptance region (4σ) that will be used to characterize the performance of the algorithm with physics events. Hence, we define an acceptance cone of $\Delta\eta \times \Delta\phi = 0.16 \times 0.12$ for TrigTileLookForMuAlg and of $\Delta\eta \times \Delta\phi = 0.2 \times 0.12$ for TrigTileRODMuAlg. For this analysis and to be able to compare their performance, we use a same size cone of $\Delta\eta \times \Delta\phi = 0.2 \times 0.12$ for both algorithms.

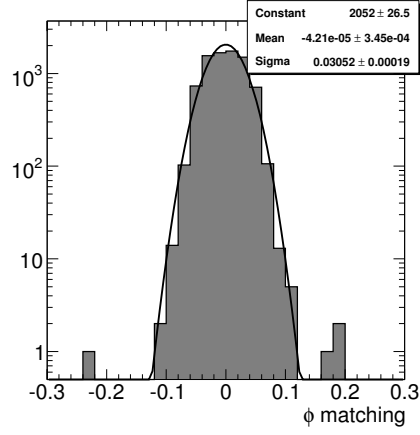
4.2 Efficiency

The μ -tagging efficiency is simply defined as the ratio between the number of tagged muons which match a Monte Carlo muon (N_{tag}) over the number of generated muons (N_{gen}):

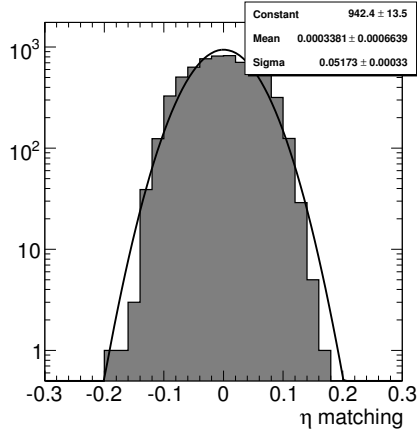
$$\varepsilon = \frac{N_{\text{tag}}}{N_{\text{gen}}}. \quad (3)$$



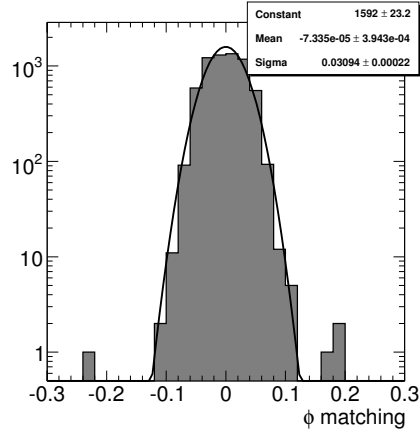
(a) Distribution of residuals in η for TrigTileLookForMuAlg.



(b) Distribution of residuals in ϕ for TrigTileLookForMuAlg.



(c) Distribution of residuals in η for TrigTileRODMuAlg.



(d) Distribution of residuals in ϕ for TrigTileRODMuAlg.

Figure 5: Distribution of residuals between the coordinates of the muons identified by both TileMuId algorithms and the truth muons in the Monte Carlo.

The cells in the towers split between TileCal long and extended barrels belong to different partitions and hence are processed by different ROD DSPs. In consequence the muon tagging algorithm cannot be applied at the ROD level for these particular towers. For this reason, as Figure 6(a) shows, the TrigTileRODMuAlg efficiency is lower than TrigTileLookForMuAlg at $0.8 \leq |\eta| \leq 1.1$ and therefore its average efficiency is slightly smaller as well. Different performance is also observed for $\eta \sim 0$, as the 2 PMTs reading out the central cell in the outermost layer (D0 cell) are processed by different ROD DSPs. Except for the gap and central regions, both algorithms show an efficiency of $\sim 85\%$ with good agreement. Since TileCal is homogeneous in ϕ , the efficiency is uniform as a function of this coordinate, see Figure 6(b).

As shown in Figure 6(c), the efficiency decreases as the muon p_T decreases for $p_T < 3$ GeV and is

about 58% (42%) for TrigTileLookForMuAlg (TrigTileRODMuAlg) at $p_T = 2$ GeV. In fact, most of the muons with $p_T \leq 2$ GeV are stopped in TileCal. For $p_T \geq 4$ GeV, the efficiency is constant in p_T with a value of 75% (60%) for TrigTileLookForMuAlg (TrigTileRODMuAlg).

Figure 7 shows the comparison between the efficiencies of both algorithms for a $b\bar{b} \rightarrow \mu(6)X$ sample, using the tight selection criteria, as a function of η and p_T . Similarly to the single muons case, differences between the efficiency of the algorithms are found in the gap region, leading to a smaller average efficiency in the case of TrigTileRODMuAlg.

Figure 8 shows the efficiency as a function of the η coordinate and the muon p_T for the TrigTileRODMuAlg algorithm distinguishing the cases when the tight and loose selection criteria are used. As expected, the efficiency increases when applying the loose selection, since the muons with large depositions in a single layer are also identified in this approach. This way, in the region $0.2 \leq |\eta| \leq 0.6$ the efficiency reaches 90%, with no special dependence on the p_T of the particle.

4.3 Fraction of fakes

The fraction of fakes (misidentified muons) given by the algorithm is computed as:

$$\text{Fraction of fakes} = \frac{\text{Number of misidentified muons}}{\text{Number of events}}. \quad (4)$$

Figure 9 shows the fraction of fakes as a function of η and ϕ for the inclusive process $b\bar{b} \rightarrow \mu(6)X$, for both algorithms and tight muon identification criteria. Both algorithms present an average fraction of fakes less than 0.12% per tower in the long barrel region ($|\eta| < 0.7$). The largest fraction of fakes corresponds to the extended barrel region, where the cells are bigger and the projectivity is worse. In particular, TrigTileLookForMuAlg shows a significant contribution coming from the gap region. As expected, the fraction of misidentified muons as a function of ϕ is a flat distribution.

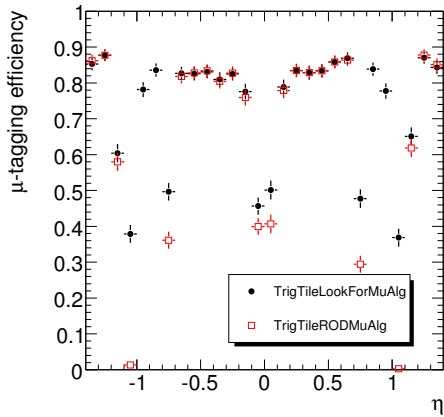
Figure 10 shows the fraction of fakes for TrigTileRODMuAlg for the loose and tight selection criteria as a function of the η and ϕ coordinates. Larger fraction of fakes is found with the loose selection in all calorimeter towers and modules, but especially in the extended barrel region. In consequence, the improvement in the efficiency obtained with the loose selection is shown to be associated with an increase in the fraction of fakes.

4.4 Summary

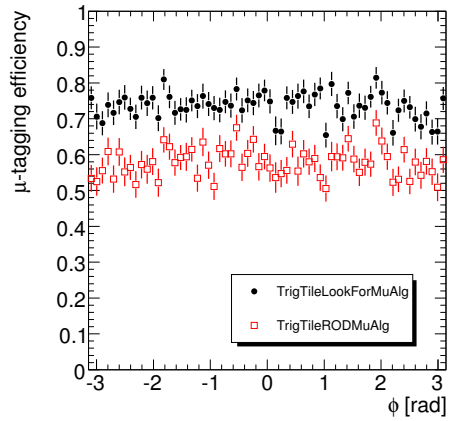
The overall performance of the Tile muon tagging algorithm has been presented in this section for both implementations of the algorithm (TrigTileLookForMuAlg and TrigTileRODMuAlg) and the two selections defined (tight and loose) using Monte Carlo samples of single muons and the inclusive $b\bar{b} \rightarrow \mu(6)X$ process. Table 11 summarizes the efficiencies and fraction of fakes found for this B-physics process for both algorithms.

Table 11: Performance of both algorithms for the inclusive process $b\bar{b} \rightarrow \mu(6)X$.

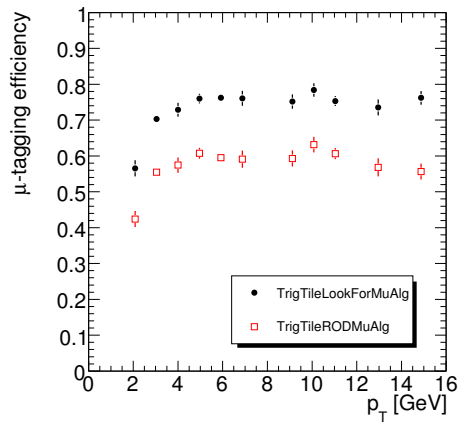
	Efficiency [%]		Fraction of fakes [%]	
	tight selection	loose selection	tight selection	loose selection
TrigTileLookForMuAlg	71.8 ± 0.4	82.5 ± 0.3	4.08 ± 0.14	5.96 ± 0.17
TrigTileRODMuAlg	56.9 ± 0.4	61.6 ± 0.4	2.74 ± 0.11	3.14 ± 0.12



(a)

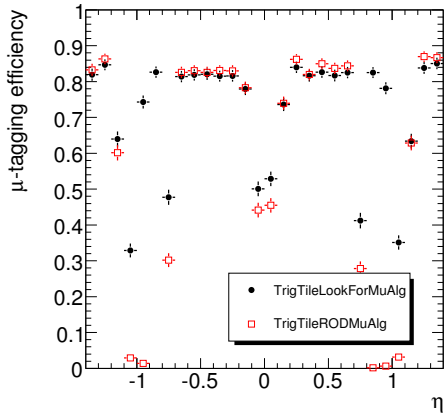


(b)

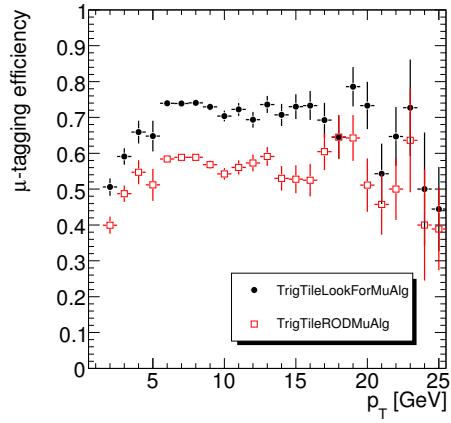


(c)

Figure 6: Muon tagging efficiency (tight selection) as a function of η , ϕ and p_T for single muon events. The performance of TrgTileLookForMuAlg (filled circles) and TrgTileRODMuAlg (open squares) is shown.

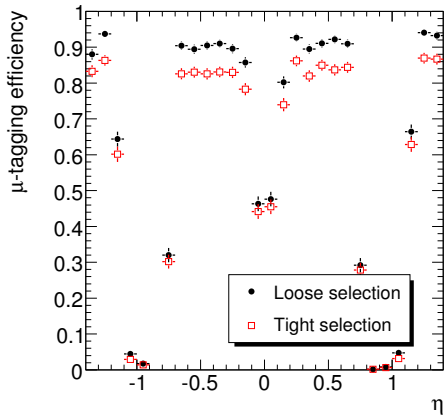


(a)

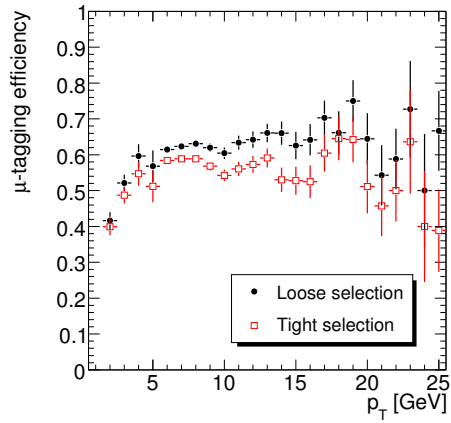


(b)

Figure 7: Muon tagging efficiency (tight selection) as a function of η and p_T for $b\bar{b} \rightarrow \mu(6)X$ events. The performance of TrigTileLookForMuAlg (filled circles) and TrigTileRODMuAlg (open squares) is shown.

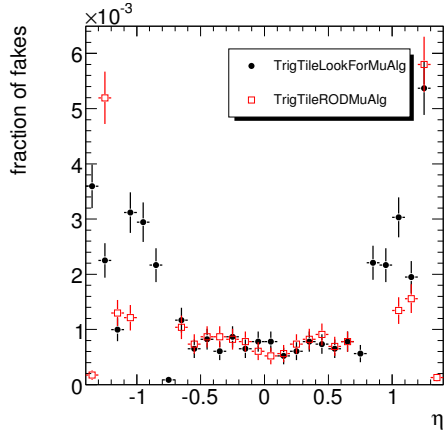


(a)

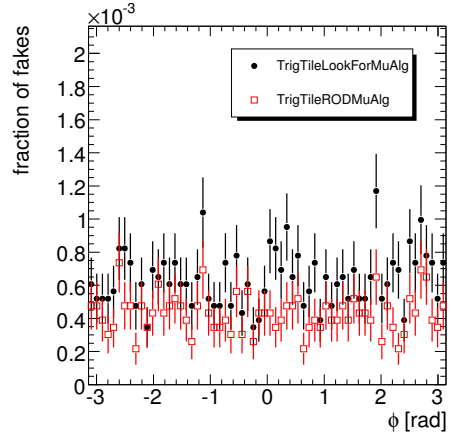


(b)

Figure 8: Muon tagging efficiency (TrigTileRODMuAlg) as a function of η and p_T for $b\bar{b} \rightarrow \mu(6)X$ events. The performance of the loose selection (filled circles) and the tight selection criteria (open squares) is shown.

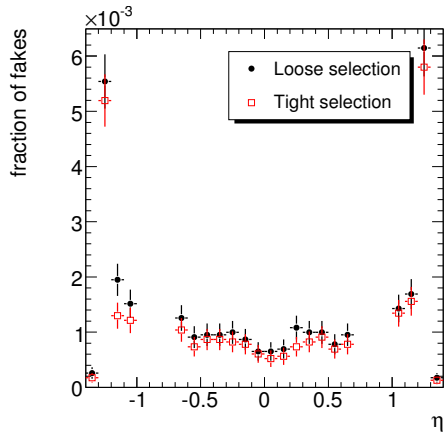


(a)

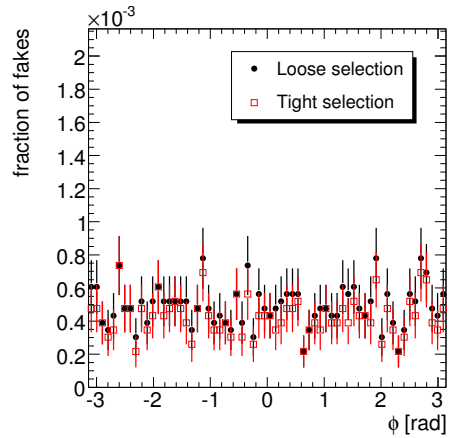


(b)

Figure 9: Fraction of fakes (tight selection) as a function of η and ϕ for $b\bar{b} \rightarrow \mu(6)X$ events. The comparison between TrigTileLookForMuAlg (filled circles) and TrigTileRODMuAlg (open squares) is shown.



(a)



(b)

Figure 10: Fraction of fakes (TrigTileRODMuAlg) as a function of η and ϕ for $b\bar{b} \rightarrow \mu(6)X$ events. The comparison between the loose selection (filled circles) and the tight selection criteria (open squares) is shown.

5 Conclusions

This note presents the latests results about the performace of the low p_T muon identification algorithm based on the energy deposition in the different layers of the Tile Calorimeter (TileMuId). This algorithm is meant to contribute to the Level-2 trigger and play an important role tagging soft muons ($p_T < 6$ GeV), where the muon spectrometer efficiency decreases significantly. The muon information from TileCal provided by TileMuId will contribute to build a soft di-muon trigger together with other ATLAS sub-detectors, whose main application will be the selection of B-physics events characterized by two low p_T muons in final state.

The aim of TileMuId is to find η -projective towers in TileCal with energy deposition compatible with a minimum ionizing particle, as delimited by an higher and a lower energy threshold. This method has been used in the past, but only recently has been implemented in the core of the ROD DSPs for its online execution. Although the ROD-based version of the algorithm has small performance limitations due to the fact that the different parts of the calorimeter are processed separately, its main advantage relies on its fast execution at Level-2 since most of the processing is done at the ROD level. The ROD DSP implementation has been discussed in detail in the present note, together with the specific ROD data fragments created to contain the Tile muon tagging information provided by the ROD which are accessed at Level-2 trigger.

The performance of the algorithm with Monte Carlo data, expressed in terms of efficiency and fraction of fakes, has been shown here comparing both implementations: TrigTileLookForMuAlg and TrigTileRODMuAlg (ROD-based approach). The results show that TileCal is able to identify muons with good efficiency down to $p_T = 2$ GeV with a small fraction of fakes. Parallel studies with real data, i.e. with cosmics data acquired during detector commissioning, are also being performed [5].

Dedicated studies to consolidate a new trigger menu devoted to B-physics channels combining the muons identified in TileCal with TileMuId and the Inner Detector muon hits are in development and will contribute to enhance the ATLAS overall trigger performance.

Acknowledgments

The authors would like to thank very especially Alexander Solodkov for all his help in the software aspects inside the athena framework needed for the algorithm development as well as his interest and advices. Also we would like to thank Margherita Primavera, Alessandro di Mattia and Michela Biglietti for their support in the Muon Trigger Slice group.

References

- [1] G. Usai, Feasibility study of a level two low p_T di-muon trigger based on Tile Calorimeter, January 2003, ATLAS Note ATL-DAQ-2003-017.
- [2] G. Usai, Trigger of a low p_T muons with the ATLAS hadronic calorimeter, Nucl. Instrum. Methods A 518 (2004), 36-38.
- [3] J. Castelo *et al.*, TileCal ROD Hardware and Software Requirements, 2005, ATLAS Note ATL-TILECAL-2005-003.
- [4] N. Konstantinidis *et al.*, A Fast Tracking Algorithm for the ATLAS Level 2 Trigger, Nucl. Instrum. Methods A 566 (2006), 166-169.
- [5] H. Lim *et al.*, ATLAS Note in preparation.

- [6] J. Prast, The ATLAS Liquid Argon Calorimeters Read Out Driver (ROD). The TMS320C6414 DSP Mezzanine board, 2005, ATLAS EDMS document ATL-AL-CD-0009.
- [7] B. Laforge and G. Perrot, TTC System Implementation for ATLAS Liquid Argon Detectors (V.2.2.2), 2004, ATLAS Note ATL-ELEC-2004-003.
- [8] C. P. Bee, D. Francis, L. Mapelli, R. McLaren, G. Monarcchi, J. Petersen and F. J. Wickens, The raw event format in the ATLAS Trigger and DAQ, 1998, Technical Report ATL-DAQ-98-129.
- [9] TileCal Valencia group, TileCal Read Out Driver Input/Output Data Format Studies, ATLAS Note in preparation.
- [10] B. Salvachúa *et al.*, Online energy and phase reconstruction during commissioning phase of the ATLAS Tile Calorimeter, 2008, ATLAS Communication ATL-COM-TILECAL-2008-004, waiting for approval as ATLAS Note.