

# ELECTROWEAK MULTIBOSON INTERACTIONS IN $Z\gamma jj$

Harry Cooke

*Thesis submitted for the degree of  
Doctor of Philosophy*



UNIVERSITY OF  
BIRMINGHAM

Particle Physics Group,  
School of Physics and Astronomy,  
University of Birmingham.

*May 3, 2023*





---

## ABSTRACT

---

---

## DECLARATION OF AUTHORS CONTRIBUTION

---

---

## ACKNOWLEDGEMENTS

---



*Motto or dedication*





# Contents

1	Introduction	1
2	Theory	2
3	The ATLAS detector at the Large Hadron Collider	3
4	Object reconstruction	4
5	Contributions to the ATLAS Level-1 Calorimeter Trigger	5
5.1	Performance studies of electron and photon algorithms for the Global Event Processor . . . . .	6
5.1.1	Introduction . . . . .	6
5.1.2	Monte Carlo samples . . . . .	7
5.1.3	Software development . . . . .	7
5.1.4	$E_{\text{ratio}}$ algorithm design . . . . .	9
5.1.5	Initial algorithm . . . . .	10
5.1.6	Peak size . . . . .	12
5.1.7	Exclusion region . . . . .	14
5.1.8	Search limit . . . . .	15
5.1.9	Algorithm summary . . . . .	16
5.2	Visualisation of eFEX inputs and algorithms . . . . .	17
5.2.1	Motivation . . . . .	18
5.2.2	Input Data . . . . .	18
5.2.3	User Interface . . . . .	19
5.2.4	Algorithms . . . . .	20
5.2.5	Usage . . . . .	23
5.3	Analysis of early Run-3 data for commissioning . . . . .	23
5.3.1	Data . . . . .	23
5.3.2	TOB and RoI selection . . . . .	24
5.3.3	Results . . . . .	24
6	Search for vector-boson scattering production of a Z boson and a photon	29
7	Search for triboson production of VZ $\gamma$ through its semi-leptonic decay mode	30
8	Conclusions	31

A FIRST APPENDIX	35
------------------	----

## List of Tables

# List of Figures

5.1	Classes, functions, and datasets used in the code, and the relationships between them. NtupleDumper represents the (modified) <b>R3L1Sim</b> software. . . . .	8
5.2	Depiction of the 6 different routes in which the $E_{\text{ratio}}$ algorithm searches for secondary maxima (left) and how the algorithm identifies secondary maxima by tracking energy gradients along each step (right). . . . .	11
5.3	Normalised $E_{\text{ratio}}$ distributions for clusters from signal and background samples (left); the integral of these distributions, showing what fraction of clusters pass a given $E_{\text{ratio}}$ cut (middle); and background rejection obtained with the $E_{\text{ratio}}$ cut corresponding to the given signal efficiency (right). . . . .	12
5.4	Illustration of how adjacent cells are included in the energy measurement for a given cell. The horizontal axis represents $\eta$ and the vertical axis $\phi$ . In all three cases shown the combined energy of the cells contained within the red box is considered to be the energy of the central, bright yellow, cell. . . . .	13
5.5	Background rejection for the $E_{\text{ratio}}$ algorithm with different peak sizes. Left plot shows how background rejection varies with signal efficiency for each of the five peak sizes tested. Right plot gives how background rejection for a fixed 95% signal efficiency varies with the choice of peak size. . . . .	13
5.6	Illustration of how an exclusion region impacts the comparisons made in the $E_{\text{ratio}}$ algorithm and where a secondary maximum can be found. Blue dots mark where energies are compared to calculate gradients. The first comparison on each route is always made against the seed cell. The shaded area shows where a secondary maximum cannot be found assuming that the seed is the highest energy cell in the region. . . . .	14
5.7	Background rejection for the $E_{\text{ratio}}$ algorithm with different exclusion regions. Left plot shows how background rejection varies with signal efficiency for each of the five exclusion regions tested. Right plot gives how background rejection for a fixed 95% signal efficiency varies with the choice of exclusion region. . . . .	15
5.8	Background rejection at 95% signal efficiency versus $\eta$ for $E_{\text{ratio}}$ algorithms using each of the exclusion regions shown. . . . .	16

5.9	Background rejection for the $E_{\text{ratio}}$ algorithm with different search limits Left plot shows how background rejection varies with signal efficiency for each of the six search limits tested. Right plot gives how background rejection for a fixed 95% signal efficiency varies with the choice of search limit. . . . .	17
5.10	Initial interface on launching the Electron Feature Extractor (eFEX) Visualiser program (left) and the default view once data is read from a file (right). . . . .	19
5.11	Demonstration of highlighting used to visualise algorithms, all shown for the same Trigger Object (TOB). Showing (a) initial view without highlighting, (b) highlighting for tau cluster energy, (c) highlighting for tau $R_{\eta}$ , and (d) highlighting for tau $R_{\text{had}}$ . . . . .	21
5.12	Demonstration of highlighting used to visualise algorithms, all shown for the same TOB. Showing (a) highlighting for EM cluster energy, (b) highlighting for EM $R_{\eta}$ , (c) highlighting for EM $R_{\text{had}}$ , and (d) highlighting for EM $w_{s,\text{tot}}$ . . . . .	22
5.13	Match rate for TOBs/Region of Interest (RoI)s as a function of energy as measured by the Cluster Processing Module (CPM). Objects grouped in 20 GeV bins, with the last bin including all overflow. . .	25
5.14	Match rate for TOBs/RoIs as a function of energy as measured by the eFEX. Objects grouped in 20 GeV bins, with the last bin including all overflow. . . . .	26
5.15	Comparison of energies for matched TOBs/RoIs with the energy as measured by the CPM given on the $x$ -axis and as measured by the eFEX on the $y$ -axis. Contains data for all matched objects in Run 423433. The dashed line marks the set of points where the CPM and eFEX energies are equal. . . . .	27
5.16	Comparison of energies for matched TOBs/RoIs with the energy as measured by the CPM given on the $x$ -axis and as measured by the eFEX on the $y$ -axis. Contains data for all matched objects in Run 427885. The dashed line marks the set of points where the CPM and eFEX energies are equal. . . . .	28

---

## DEFINITIONS OF ACRONYMS

---

**CERN** The European Laboratory for Particle Physics

**LHC** Large Hadron Collider  
Superconducting collider occupying the 27 km ring at CERN.

**QCD** Quantum Chromodynamics

**eFEX** Electron Feature Extractor

**ATLAS** A Toroidal LHC Apparatus

**TOB** Trigger Object

**RoI** Region of Interest

**UI** User Interface

**FPGA** Field-programmable Gate Array

**L1Calo** Level-1 Calorimeter

**CPM** Cluster Processing Module

**BCID** Bunch Crossing ID

**LAr** Liquid Argon

**LATOME** Liquid Argon Trigger Optical Mezzanine

**LTA** Long-term Attachment

**GEP** Global Event Processor

# CHAPTER 1

---

## Introduction

---



## CHAPTER 2

---

### Theory

---

## CHAPTER 3

---

### The ATLAS detector at the Large Hadron Collider

---

## CHAPTER 4

---

### Object reconstruction

---

## CHAPTER 5

---

### Contributions to the ATLAS Level-1 Calorimeter Trigger

---

Throughout the course of the PhD, several contributions have been made to the A Toroidal LHC Apparatus (ATLAS) Level-1 Calorimeter (L1Calo) trigger. Initially, this task served as authorship qualification, making a technical contribution to the experiment in order to be considered a qualified author of the collaboration. Work done as part of the qualification task is detailed in Section 5.1.

Once qualification was achieved it made more sense to direct technical work towards more current aspects of L1Calo development, alongside a Long-term Attachment (LTA) working at The European Laboratory for Particle Physics (CERN) enabling direct collaboration with on-site experts and the opportunity to assist with hands-on installation work. Sections 5.2 and 5.3 detail the work done during this time period.

## 5.1 Performance studies of electron and photon algorithms for the Global Event Processor

### 5.1.1 Introduction

The Global Event Processor (GEP), when introduced in the Phase-II upgrade, will aim to improve discrimination in the hardware trigger for many signatures, but notably for  $e/\gamma$  objects. The GEP will be working alongside the eFEX system, introduced in Phase I, but will have access to more information, giving it potential to improve upon decisions made by the eFEX. To realise this improvement, new algorithms will need to be implemented in the GEP to take advantage of the finer granularity information available to it.

Designing algorithms to be used in future hardware systems is achieved through prospective performance studies. Performance studies use simulations of the expected response of a system to evaluate the performance of individual algorithms. These studies benefit from the ease of implementation of algorithms in high-level software but it is still important to consider the complexity of implementation in firmware when designing algorithms. Performance studies are typically the first step in designing a system as evaluating performance in simulations before a system is built can inform the design of the hardware.

This section explores the specific implementation and possible performance of the  $E_{\text{ratio}}$  algorithm in the GEP, expected to significantly improve discrimination for  $e/\gamma$  by making use of fine granularity input information [1, p. 126],  $E_{\text{ratio}}$ . Section ?? details the samples used in... Section ... gives ... Section ...

### 5.1.2 Monte Carlo samples

Simulations used for the performance studies presented here are from two Monte Carlo samples: a  $Z \rightarrow ee$  sample providing signal EM objects that the trigger should be accepting, and a minimum bias QCD sample providing background objects, typically low-energy jets, that the trigger should be rejecting. The signal sample is generated by POWHEG [2] and PYTHIA 8 [3, 4], and the background sample is generated by PYTHIA 8.

After being processed by the typical ATLAS detector simulation (see Section X), additional simulations of the upgraded trigger are performed by the Phase-I offline software in order to simulate the eFEX response to each event.

### 5.1.3 Software development

The software used in preliminary studies, `R3L1Sim`, simulates the planned form of the trigger in the Phase-I upgrade. To be able to use this for performance studies required for the Phase-II upgrade some modifications must be made. The most significant of these is collecting information from individual calorimeter cells in the region around any  $e/\gamma$  candidates identified by the simulated Phase-I L1 trigger, referred to here as seeds. A framework for handling these clusters of cells has been developed, designed for ease of use in Phase-II performance studies.

The goal is to form a cluster of cells around each seed, containing all cells from Layer 1 of the EM calorimeter within a given distance in  $\eta$  and  $\phi$ . Clusters were chosen to have a total width of 0.3 in both  $\eta$  and  $\phi$ . This size should be suitable to contain all activity surrounding detected particles.

Clusters are complicated by the changing geometry of the calorimeter, described in Section ???. For algorithms to be able to access cells sequentially without explicit knowledge of each cell's coordinates, the cluster must know the geometry of the cells in the calorimeter region it occupies. This is easily achievable if a cluster is fully

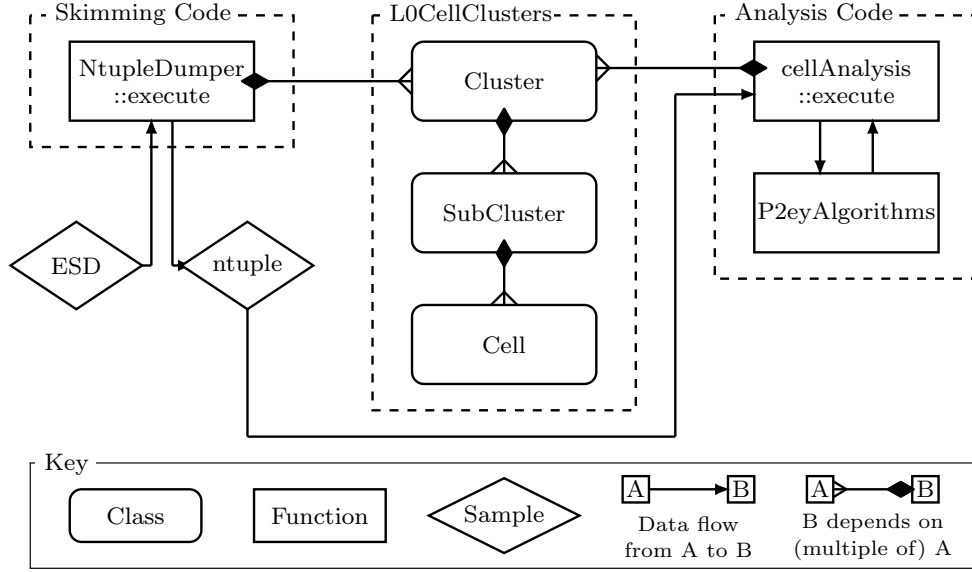


Figure 5.1: Classes, functions, and datasets used in the code, and the relationships between them. NtupleDumper represents the (modified) **R3L1Sim** software.

contained within a single calorimeter region, but becomes difficult if a cluster crosses a region boundary.

The solution employed here is to split each cluster into subclusters, where each subcluster covers the extent of the cluster contained by a single calorimeter region. For instance, a cluster that falls towards the edge of the barrel might have two subclusters, one in the barrel region and one in the barrel-endcap transition region. Individual cells can then be accessed by indexing the subclusters with a coordinate system based on distance from the centre of the cluster.

The cluster structure described above is implemented through a collection of classes. **Cluster**, **SubCluster**, and **Cell** classes contain the core mechanics including constructing the cluster from a seed location, constructing appropriate subclusters, storing cell energies and coordinates if they are within cluster bounds, and indexing and locating cells. Additional classes are used to help with some of the more complicated tasks; Figure 5.1 shows the classes and functions used, and the relationships between them.

To access cell cluster information for trigger analysis, the cluster classes need to

be output to the ROOT [5] file generated by R3L1Sim. Initially this was done by directly saving the `Cluster` class to the ROOT file, by providing dictionaries for each of the defined classes to enable the interpreter to interact with them. This worked at first when executed locally on test data but caused issues when run on the Grid, which is necessary to be able to process the large amounts of data required for the study.

After a large amount of troubleshooting, no cause could be found for these issues. This was solved by taking an alternative approach to saving the data. The class structure can be decomposed to a set of arrays (STL vectors in C++) containing all of the information associated with a cluster, its subclusters, and all contained cells. The classes have the capability to output this decomposed information, and also to be reconstructed from it.

Integrating these classes into R3L1Sim, and adding a few lines of code to the event loop to create and save clusters, provides the information needed for Phase-II studies with Layer-1 strip information. In future this code will likely be expanded to include more calorimeter layers.

#### 5.1.4 $E_{\text{ratio}}$ algorithm design

Performance studies so far have focused on the  $E_{\text{ratio}}$  variable described in Section ?? . This work expands on previous studies which highlighted the potential of the  $E_{\text{ratio}}$  algorithm for the Phase-II trigger [1]. The focus here is on designing and tuning a version of the  $E_{\text{ratio}}$  that could be applied in firmware, as will be required when the upgrade is implemented. The algorithm, combined with the software described above, is then used to obtain estimates for the performance of the Phase-II  $e/\gamma$  trigger.



### 5.1.5 Initial algorithm

Calculating  $E_{\text{ratio}}$  is done for every cluster generated by the modified R3L1Sim software. An initial algorithm was defined, chosen to make each step as simple as possible for firmware implementation, before refinements are made to improve performance.

The cluster contains the location of the seed SuperCell that it receives from the Phase-I simulation. To find the seed for the  $E_{\text{ratio}}$  algorithm, the cells in the SuperCell are compared against each other to find the cell with the highest energy.

Once the seed cell is located, the algorithm searches for possible secondary maxima. This is done by stepping out one cell at a time away from the seed, and along each step calculating the energy gradient:

$$\Delta E = E_{\text{next}}^{\text{cell}} - E_{\text{prev}}^{\text{cell}},$$

where  $E_{\text{next}}^{\text{cell}}$  is the energy of the cell being stepped to and  $E_{\text{prev}}^{\text{cell}}$  is the energy of the cell being stepped from. Initially  $\Delta E$  should be negative, as the seed is the highest energy cell in the cluster. At some point  $\Delta E$  may become positive, indicating that a secondary maximum is being approached. If  $\Delta E$  becomes negative again then the cell before the change is marked as a candidate secondary maximum. If the algorithm reaches the edge of the cluster and  $\Delta E$  is positive then the last cell is taken as a candidate instead.

This iterative process is done in six different routes through the cluster: with the same  $\phi$  coordinate as the seed and one either side, each either in positive or negative  $\eta$ . If the route is one with a different  $\phi$  coordinate to the seed then the first step is out in  $\phi$  from the seed and all subsequent steps are in  $\eta$ . Figure 5.2 illustrates the different routes.

Each of the six routes can return a candidate secondary maximum. The largest of these six is taken to be the secondary maximum and used to calculate  $E_{\text{ratio}}$ , as in

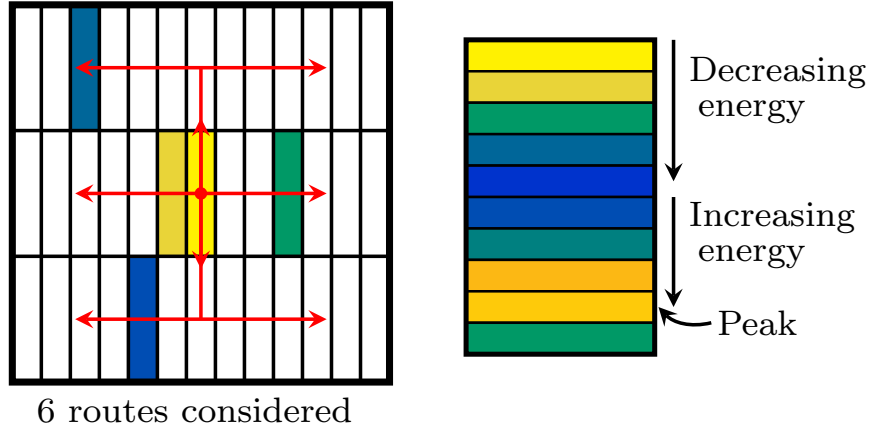


Figure 5.2: Depiction of the 6 different routes in which the  $E_{\text{ratio}}$  algorithm searches for secondary maxima (left) and how the algorithm identifies secondary maxima by tracking energy gradients along each step (right).

Equation ??, with the seed as the primary maximum.

Figure 5.3 shows the  $E_{\text{ratio}}$  distributions for the signal and background samples. The discriminating power of the variable is evident from the difference in shape between the two distributions, requiring candidates to have  $E_{\text{ratio}}$  in the lower end of the range clearly could reject a large amount of background and maintain much of the signal. This is quantified in the figure by showing the fraction of clusters that would be selected by the trigger for a given  $E_{\text{ratio}}$  threshold, for both signal and background. Any suitable  $E_{\text{ratio}}$  threshold is possible to implement in the trigger; the figure of interest is how much of the background can be rejected whilst maintaining a set fraction of signal events. The third plot in the figure shows the background rejection for a range of signal efficiencies. Background rejection is defined as the inverse of the fraction of background events passing the selection.

The primary figure of merit used to evaluate the performance of different  $E_{\text{ratio}}$  algorithms in this study is the background rejection at 95% signal efficiency. This initial algorithm achieves a background rejection of 2.3 at 95% signal efficiency.

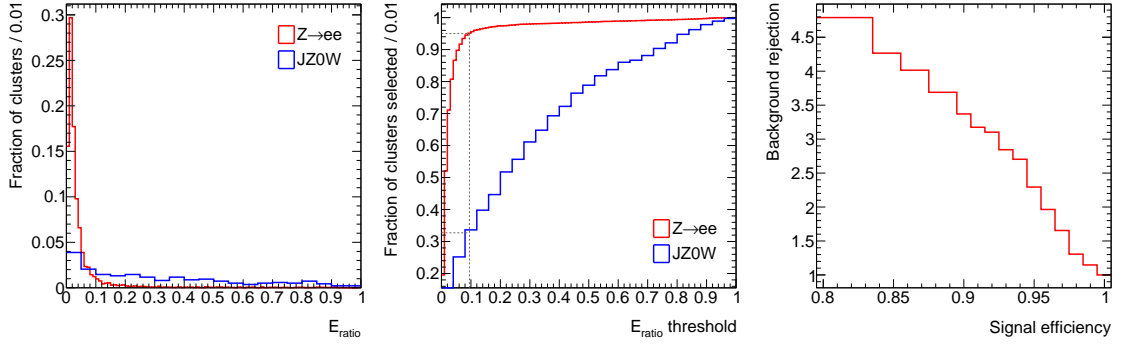


Figure 5.3: Normalised  $E_{ratio}$  distributions for clusters from signal and background samples (left); the integral of these distributions, showing what fraction of clusters pass a given  $E_{ratio}$  cut (middle); and background rejection obtained with the  $E_{ratio}$  cut corresponding to the given signal efficiency (right).

### 5.1.6 Peak size

Several parameters were identified in the initial algorithm that could be adjusted to potentially improve the background rejection. The first of these parameters is the size of the ‘peak’ used to measure energies. In the first form of the algorithm energies were measured in single cells, which is equivalent to a peak size of one. With a larger peak each energy measurement is taken as the sum of the energies of the cell in question and its nearest neighbours in  $\eta$ , such that the number of cells summed over is equal to the peak size. Figure 5.4 shows how cells are selected for different peak sizes.

The  $E_{ratio}$  algorithm was applied to the signal and background samples with a range of different peak sizes. Figure 5.5 shows how changing the peak size impacts the background rejection of the algorithm. It is clear that in the region of interest, particularly at 95% signal efficiency, the original peak size of one cell gives the best background rejection, so no improvement has been identified.

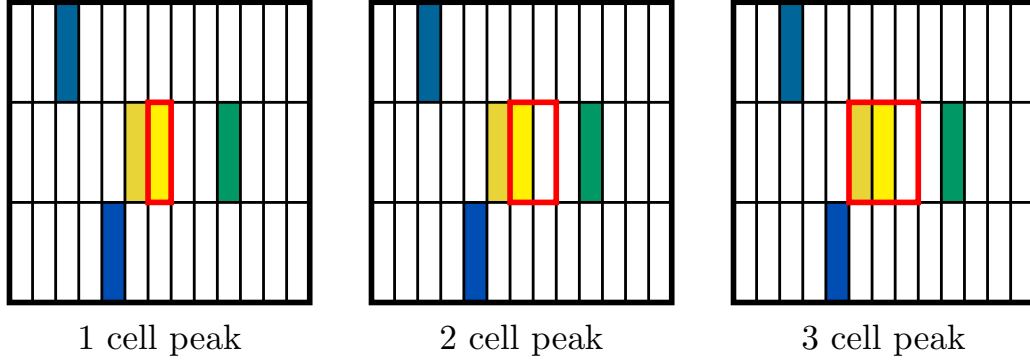


Figure 5.4: Illustration of how adjacent cells are included in the energy measurement for a given cell. The horizontal axis represents  $\eta$  and the vertical axis  $\phi$ . In all three cases shown the combined energy of the cells contained within the red box is considered to be the energy of the central, bright yellow, cell.

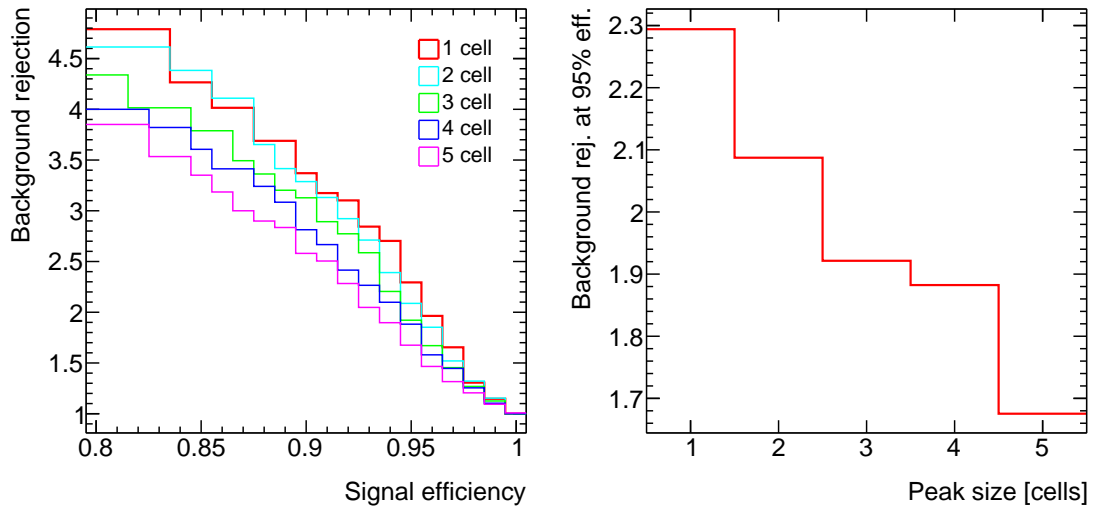


Figure 5.5: Background rejection for the  $E_{\text{ratio}}$  algorithm with different peak sizes. Left plot shows how background rejection varies with signal efficiency for each of the five peak sizes tested. Right plot gives how background rejection for a fixed 95% signal efficiency varies with the choice of peak size.

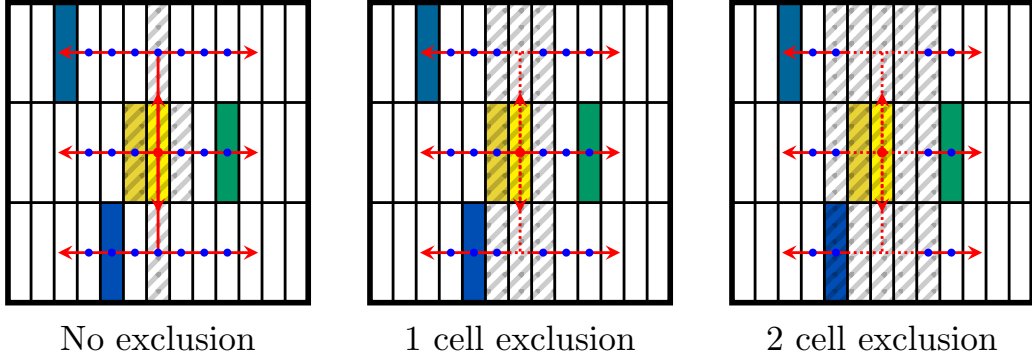


Figure 5.6: Illustration of how an exclusion region impacts the comparisons made in the  $E_{\text{ratio}}$  algorithm and where a secondary maximum can be found. Blue dots mark where energies are compared to calculate gradients. The first comparison on each route is always made against the seed cell. The shaded area shows where a secondary maximum cannot be found assuming that the seed is the highest energy cell in the region.

### 5.1.7 Exclusion region

Another parameter investigated was the exclusion region. The exclusion region is defined as the distance in  $\eta$  from the seed in which secondary maxima are not considered distinct from the primary maximum. The initial algorithm has no exclusion region. An  $n$ -cell exclusion region is implemented by making the first comparison with the cell  $n$  cells in  $\eta$  away from the seed, regardless of  $\phi$  coordinate of the route in question. Figure 5.6 shows the differences between different exclusion regions.

A range of exclusion regions were applied to the  $E_{\text{ratio}}$  algorithms to check their impact on background rejection. Figure 5.7 shows the background rejection with each variation of the algorithm. In this case improvements were seen over the initial algorithm, with a one cell exclusion region giving the best background rejection of 3.05 at 95% signal efficiency.

Given that cell widths vary significantly in different regions of the calorimeter, the exclusion region was investigated over a range of  $\eta$  values. Figure 5.8 compares the performance of the  $E_{\text{ratio}}$  algorithm with different exclusion regions against  $\eta$ . The one cell exclusion region performs best in each bin except the second. The second

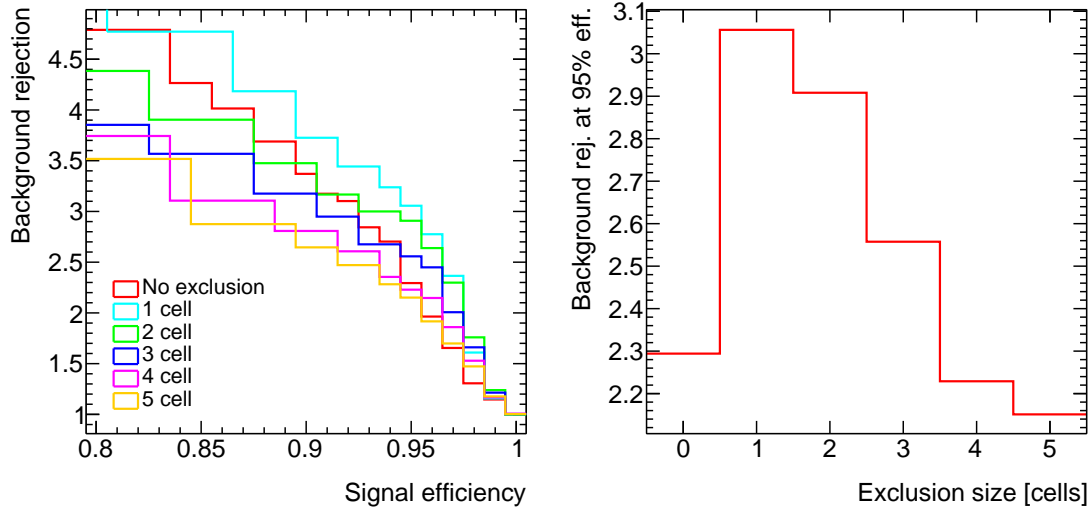


Figure 5.7: Background rejection for the  $E_{\text{ratio}}$  algorithm with different exclusion regions. Left plot shows how background rejection varies with signal efficiency for each of the five exclusion regions tested. Right plot gives how background rejection for a fixed 95% signal efficiency varies with the choice of exclusion region.

bin contains all clusters that touch the transition region of the calorimeter, where the algorithm is not expected to perform sensibly due to the lack of strips, so the drop in performance here is not an issue. This shows that the choice of exclusion region has no significant dependence on the strip width.

### 5.1.8 Search limit

The final parameter to have been investigated so far is the search limit. The search limit places an upper limit on the distance, in  $\eta$ , travelled from the seed in search of a secondary maximum. In the initial algorithm the search is only limited by the width of the clusters, this is equivalent to a search limit of 0.15.

A range of search limits, from 0.15 down to 0.025, were implemented in the  $E_{\text{ratio}}$  algorithm to see if it could improve background rejection. Figure 5.9 shows how the search limit impacts background rejection. No improvement is found over the initial 0.15 search limit. There appears to be a plateau in performance, the maximum background rejection could still be obtained with a search limit of 0.1. This could

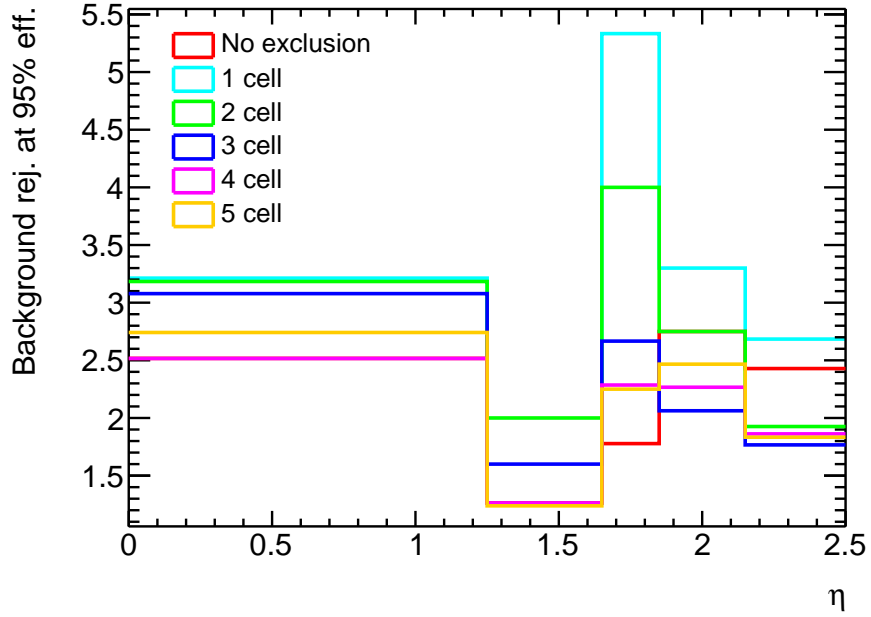


Figure 5.8: Background rejection at 95% signal efficiency versus  $\eta$  for  $E_{\text{ratio}}$  algorithms using each of the exclusion regions shown.

prove useful if the current cluster size is too large to be implemented in firmware.

### 5.1.9 Algorithm summary

Of the three parameters investigated only the exclusion region gave an increase in performance over the initial algorithm. The recommended form of the algorithm is therefore one with a peak size of one cell, an exclusion region of one cell, and a search limit of 0.15 (or anywhere between 0.1 and 0.15).

There are still changes to be made to the current algorithm and simulation. Currently the Phase-II simulation relies on the Phase-I eFEX simulation to seed the algorithm. This is not representative of the hardware plans, discussed in Section ???. The Phase-II simulation needs to be adapted to form seeds without input from eFEX. The studies presented here all use samples with 80 pileup collisions per bunch crossing; this should be increased to around 200 collisions per bunch crossing to properly represent expected Phase-II conditions.

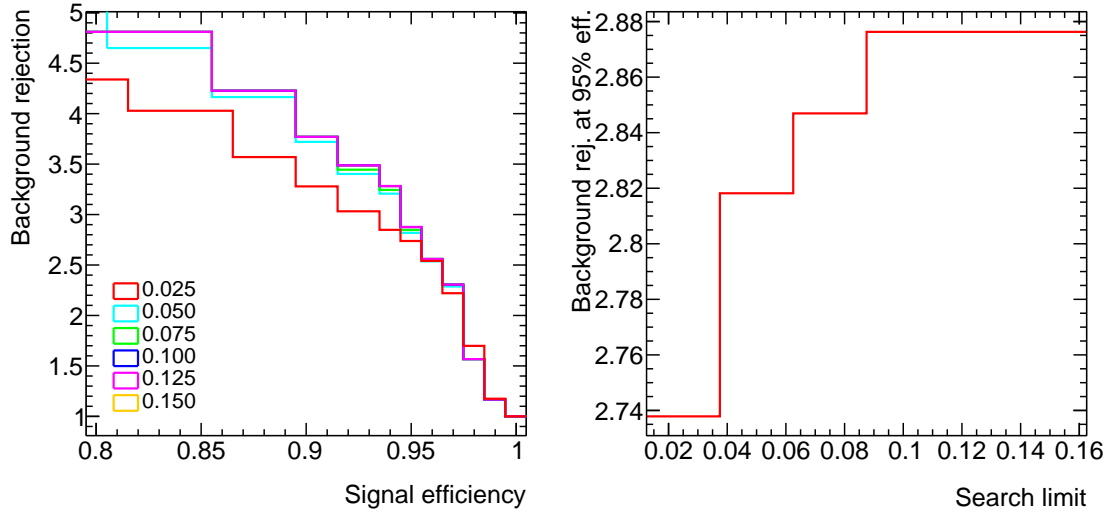


Figure 5.9: Background rejection for the  $E_{\text{ratio}}$  algorithm with different search limits. Left plot shows how background rejection varies with signal efficiency for each of the six search limits tested. Right plot gives how background rejection for a fixed 95% signal efficiency varies with the choice of search limit.

Further studies could include investigating the impact of noise in the calorimeter energies on  $E_{\text{ratio}}$  performance. This could include adding a threshold that energy differences have to exceed to be considered a change in gradient.

## 5.2 Visualisation of eFEX inputs and algorithms

In order to aid in debugging minor differences between different implementations of eFEX algorithms, an algorithm visualisation program was created. The program reads input data and performs aspects of eFEX algorithms whilst also providing a visual representation of what the algorithm is doing and where the result comes from.

The visualiser is written in JavaScript, using Node.js to interface with some server-side C++ scripts, needed to access energy decoders from online software, and Express.js to handle the web-based User Interface (UI).



### 5.2.1 Motivation

During development of algorithms for the hardware trigger, each algorithm is implemented multiple times. First, algorithms will be implemented in offline software to be tested and tuned against simulations or existing data. Then, in order to run on hardware, the algorithm needs to be ported to firmware. Often, to provide closer cross-checks of the firmware algorithms, they are also simulated in online software. Inevitably, due to software and firmware bugs, subtle differences will exist between these algorithms; these differences need to be understood and corrected to have a complete bug-free implementation.

The need to find these subtle differences between algorithms motivates the visualisation software discussed here. Although at first it seems illogical to add an additional, independent, implementation of the algorithm (since the issue is in part due to having multiple different implementations), the added visualisation aspect makes it easier to understand where a particular algorithm implementation might have gone wrong in cases where there are discrepancies. This has been demonstrated through the use of the visualisation software in tests, discussed in Section 5.2.5.

### 5.2.2 Input Data

The visualisation software takes as input the calorimeter energies visible to a single eFEX Field-programmable Gate Array (FPGA). This covers a  $6 \times 10$  area in  $\eta \times \phi$  of trigger towers, with each tower being split into SuperCells across 5 calorimeter layers, as described in Section X. For each event, the input data provides one energy value per SuperCell, encoded with either LAr or Tile energy encoding.

From this input data, a  $3 \times 3$  area of trigger towers, centred on an  $(\eta, \phi)$  coordinate provided by the user, is extracted and displayed on-screen. This area covers all energy values used for eFEX algorithms if the seed of the TOB is located in the central trigger tower.

### 5.2.3 User Interface

The eFEX Visualiser program provides a minimal UI to explore input data and results of the eFEX algorithms. The basic interface is shown in Figure 5.10. It prompts the user to specify an input file,  $(\eta, \phi)$  centre-tower coordinates, and an event number, then on receipt of these inputs it reads the information and displays the requested energies in a grid.

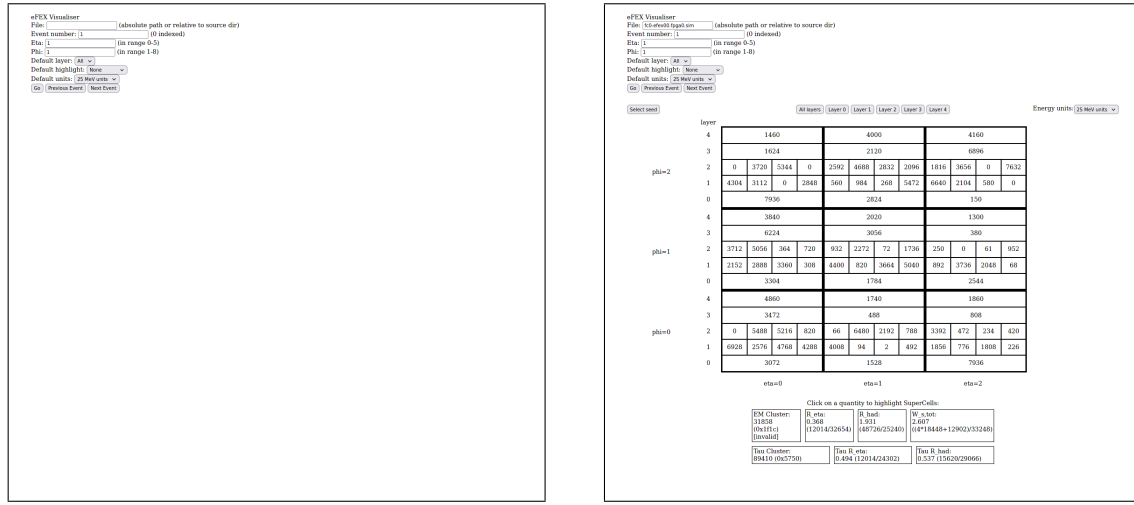


Figure 5.10: Initial interface on launching the eFEX Visualiser program (left) and the default view once data is read from a file (right).

The full interface becomes visible after the grid is displayed. The grid itself is a  $3 \times 3$  area divided by bold lines, with each square representing a trigger tower, and each trigger tower square divided further into SuperCells. The horizontal axis represents the  $\eta$  coordinate of the tower or SuperCell, and the vertical axis represents the  $\phi$  coordinate. These coordinates are labelled with the same indices the user gave as initial input. In order to show all layers simultaneously, in the default view layers are stacked on top of each other, in the phi axis, within each tower. Controls are provided to instead view each layer individually if preferred.

Below the grid, a list of all the quantities calculated for the current TOB is displayed. Clicking on one of these quantities will highlight all of the SuperCells involved in the calculation. The details of how these values are calculated and how the algorithms are visualised are discussed in more detail in Section 5.2.4.

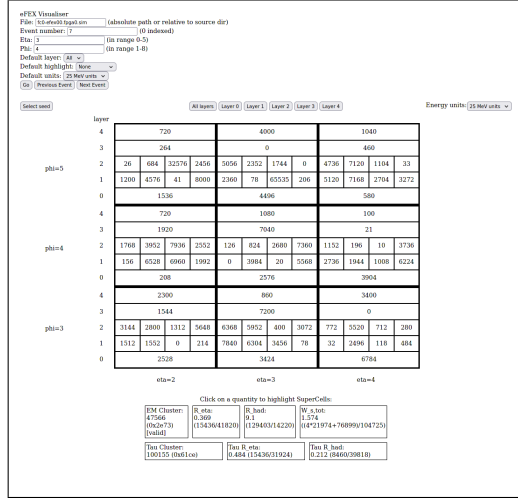
Above the grid, alongside the layer selection buttons, are options to manually set the seed SuperCell and to select the units used to display energies. The unit selection input is a drop-down box that allows the user to choose between 25 MeV (default units in firmware) or GeV units. Changing this option instantly updates all displayed energies. Pressing the “Select seed” button will toggle the layer view to display Layer 2, prompt the user to click on the SuperCell with the highest energy, and then on its  $\phi$ -neighbour with the highest energy. This aids the user in selecting the correct seed for TOB generation, but is not normally necessary as the program will apply these criteria to automatically set the seed as soon as the grid is loaded. The manual override is included in case the automatic selection is wrong, or if looking at algorithms with a different seed may help debugging.

#### 5.2.4 Algorithms

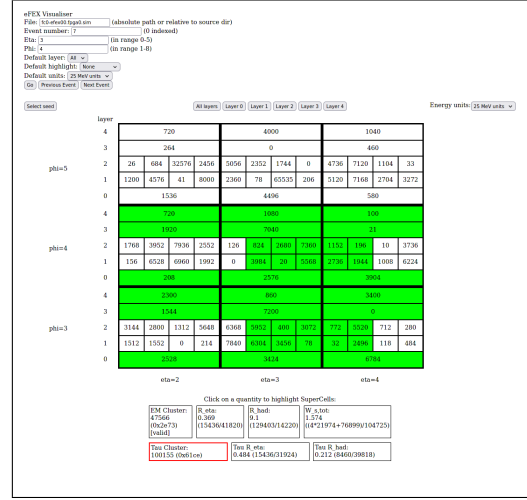
For each TOB processed (i.e. each particular event, coordinate location, and seed), several algorithms are run to calculate the quantities displayed on-screen. These are the same algorithms used by the eFEX to calculate TOB energies and isolations. The following variables are calculated: EM cluster energy, EM  $R_\eta$ , EM  $R_{\text{had}}$ , EM  $w_{s,\text{tot}}$ , tau cluster energy, tau  $R_\eta$ , and tau  $R_{\text{had}}$ . All of these are either sums of SuperCell energies (EM and tau cluster energies), ratios of sums of SuperCell energies ( $R_\eta$  and  $R_{\text{had}}$ ), or a ratio with weighted sums ( $w_{s,\text{tot}}$ ).

The values of these variables are calculated immediately after the data for a given TOB is collected, or once the seed is re-specified, and displayed on-screen below the grid. If the user clicks on a displayed quantity, the SuperCells involved in the sums for the corresponding algorithm are highlighted with colours corresponding to whether those cells are used in the numerator (lime green); the denominator (gold); or, in the case of  $w_{s,\text{tot}}$ , in the numerator with a larger weight (dark green). Figures 5.11 and 5.12 demonstrate the highlighting for all of the algorithms.

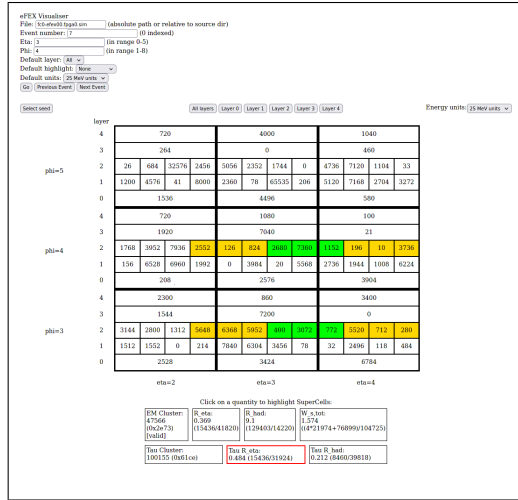
## 21 CHAPTER 5. CONTRIBUTIONS TO THE ATLAS LEVEL-1 CALORIMETER TRIGGER



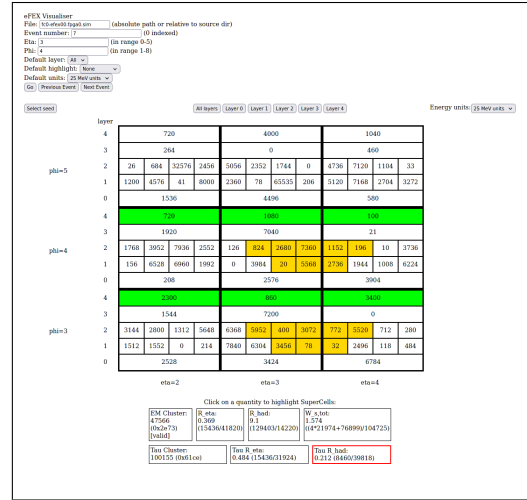
(a)



(b)

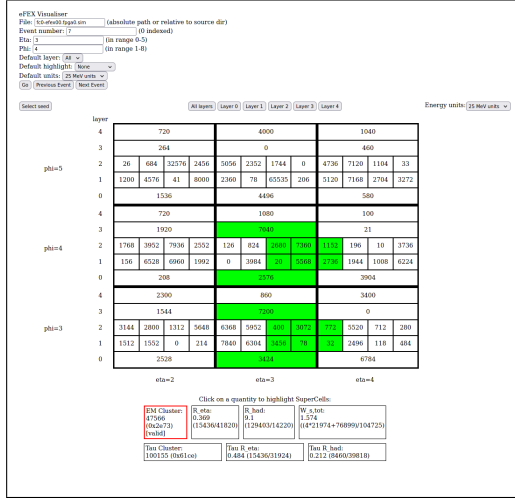


(c)

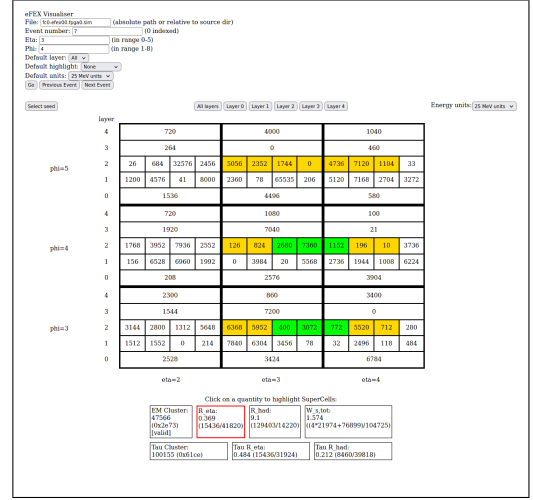


(d)

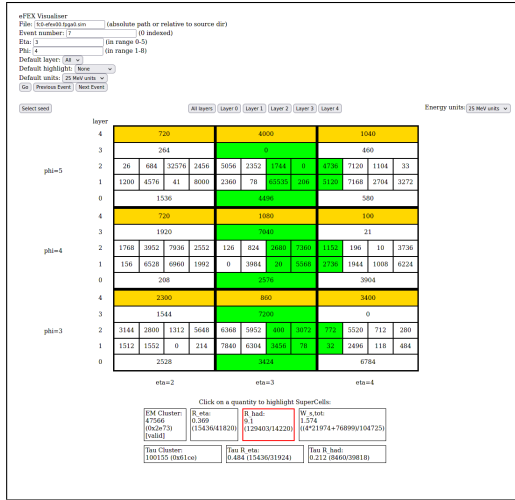
Figure 5.11: Demonstration of highlighting used to visualise algorithms, all shown for the same TOB. Showing (a) initial view without highlighting, (b) highlighting for tau cluster energy, (c) highlighting for tau  $R_\eta$ , and (d) highlighting for tau  $R_{had}$ .



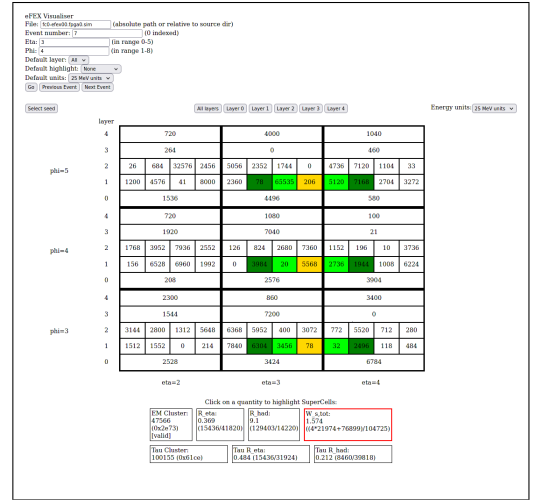
(a)



(b)



(c)



(d)

Figure 5.12: Demonstration of highlighting used to visualise algorithms, all shown for the same TOB. Showing (a) highlighting for EM cluster energy, (b) highlighting for EM  $R_\eta$ , (c) highlighting for EM  $R_\phi$ , and (d) highlighting for EM  $w_{s,tot}$ .

### 5.2.5 Usage

The visualisation tool was used at several stages during commissioning of the eFEX. Primarily it was used to compare firmware algorithms to their implementation in online software. Each time there was a difference found between the two, the event could be checked with the visualiser to help determine which of them was correct and to work out how the other might have gone wrong.

Once there was sufficient confidence in the similarity between online simulation and firmware implementations, the visualiser was again used to help in tests comparing online and offline simulations. The same technique was applied here to help ensure offline simulations were running with the same results as their online equivalent, and to help find errors in cases where they were not.

## 5.3 Analysis of early Run-3 data for commissioning

At the start of Run 3, the Phase-1 L1Calo trigger was being used for the first time, having just been installed in the ATLAS detector. In these early stages, the new Phase-1 system was running in parallel to the legacy system, but the legacy system was still being used in the trigger menu over the new system.

One of the key goals of this time period was validating the Phase-1 trigger system, comparing it to the legacy system to identify any differences which may have arisen from bugs or hardware issues. This section describes analysis of some early Run-3 data contributing to this goal.

### 5.3.1 Data

Two runs were used to provide the data for this analysis: Run 423433, taken on 31 May 2022, and Run 427885, from 10 July 2022. These runs were taken in quite

different conditions, the first with lower intensity beams and no stable beam conditions, and the second with high intensity stable beams. Notably, the second of these runs had bunch trains with 25 ns separation between bunches, whereas the first had only isolated bunches.

Events are taken from the `physics_Main` stream. This stream contains 1,636,636 events for Run 423433 and 107,016 events for Run 427885.

### 5.3.2 TOB and RoI selection

Phase-1 TOBs and legacy RoIs in events are compared to find instances in the same event that have the same, or very similar,  $\eta$ - $\phi$  coordinates. A match is considered to be a pair of objects within  $\pm 1$  trigger tower in both  $\eta$  and  $\phi$ , i.e. a TOB matches an RoI if it falls within the  $3 \times 3$  area of trigger towers centred on the tower containing the RoI. Matched objects are considered to be the same physics object, identified independently by both systems. Instances where there is a TOB or RoI with no analogue in the opposing system are also tracked.

Only the barrel region was considered for this as a preliminary investigation, since it has a simpler geometry and as such it is easier to isolate bugs. Also, at the time of analysing, only half of the eFEX modules were installed so the Phase-1 system had coverage for half of the  $\phi$  range; only RoIs inside of this coverage are accepted.

### 5.3.3 Results

From the 1,636,636 events in Run 423433, 292,498 RoI/TOB pairs are selected. Of these, 271,854 matched in  $\eta - \phi$  coordinates, giving a total match rate of 93%. For Run 427885, 22,337 of 27,973 pairs were matched for a match rate of 80%. Figures 5.13 and 5.14 show the match rate for objects in Run 427885 as a function of energy, using CPM-measured and eFEX-measured energies respectively. This shows that the bulk of the mismatches come from low-energy objects, with a notably higher match

rate at higher energies. At all energies though the match rate is worse here than in the earlier run.

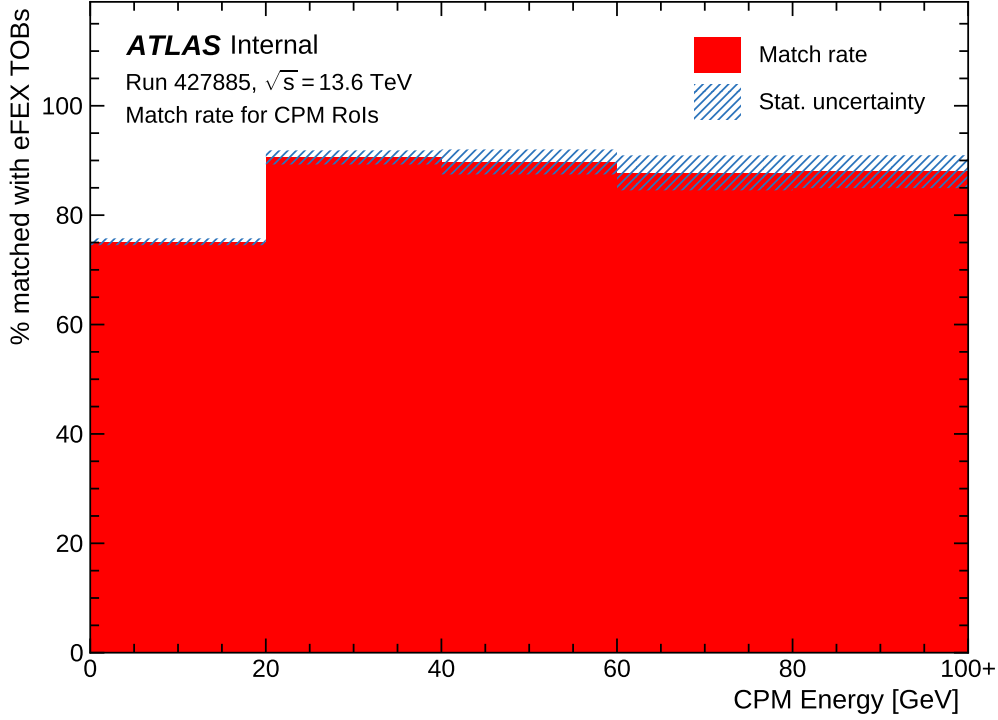


Figure 5.13: Match rate for TOBs/RoIs as a function of energy as measured by the CPM. Objects grouped in 20 GeV bins, with the last bin including all overflow.

Figures 5.15 and 5.16 compare the energies recorded by the legacy and the Phase-1 systems for matched objects in the two runs. In Run 423433 it is clear that the majority of matched objects have approximately the same energy, with an additional cluster where in a few cases the eFEX-measured energy is much lower than the CPM-measured.

In the later run, Run 427885, however, there is no longer such a strong correlation in energies. It seems that in general the eFEX energies are measured to be some fraction of the CPM energies – seen by the gradient of the area containing the majority of objects being less than the equal-energies line. Once again there is another cluster of objects with very low eFEX energies at high CPM energies.

So the general trends are a high but imperfect match rate; and decreased performance in the later run compared to the earlier run, both in terms of match rate



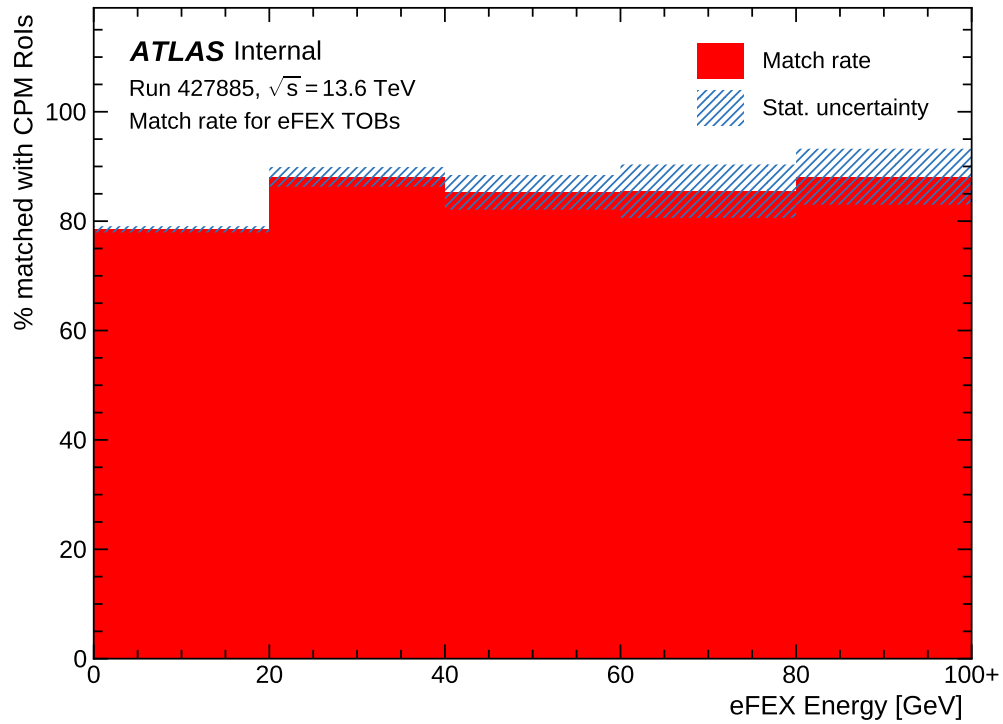


Figure 5.14: Match rate for TOBs/RoIs as a function of energy as measured by the eFEX. Objects grouped in 20 GeV bins, with the last bin including all overflow.

of objects and energy correlation between the two systems. This information was fed back to people working on the systems to help them track down the root issues. Likely the reason for degraded performance in the later run, despite constant improvements to the system in the time between the two runs, is due to the different beam conditions. For example, the bunch trains present in Run 427885 (but not Run 423433) could have caused issues with Bunch Crossing ID (BCID) on the Liquid Argon Trigger Optical Mezzanine (LATOME) modules feeding information to the eFEX.

[PERHAPS ANOTHER SUBSECTION HERE WITH AN UPDATE ON THE SYSTEM AND HOW THESE ISSUES WERE SOLVED]

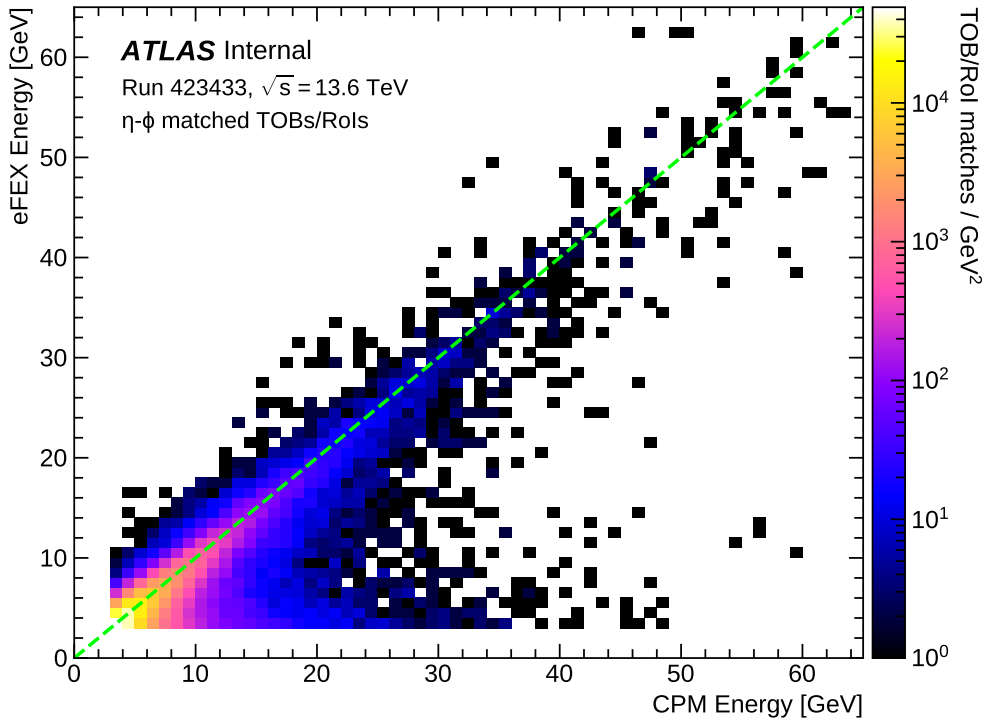


Figure 5.15: Comparison of energies for matched TOBs/RoIs with the energy as measured by the CPM given on the  $x$ -axis and as measured by the eFEX on the  $y$ -axis. Contains data for all matched objects in Run 423433. The dashed line marks the set of points where the CPM and eFEX energies are equal.

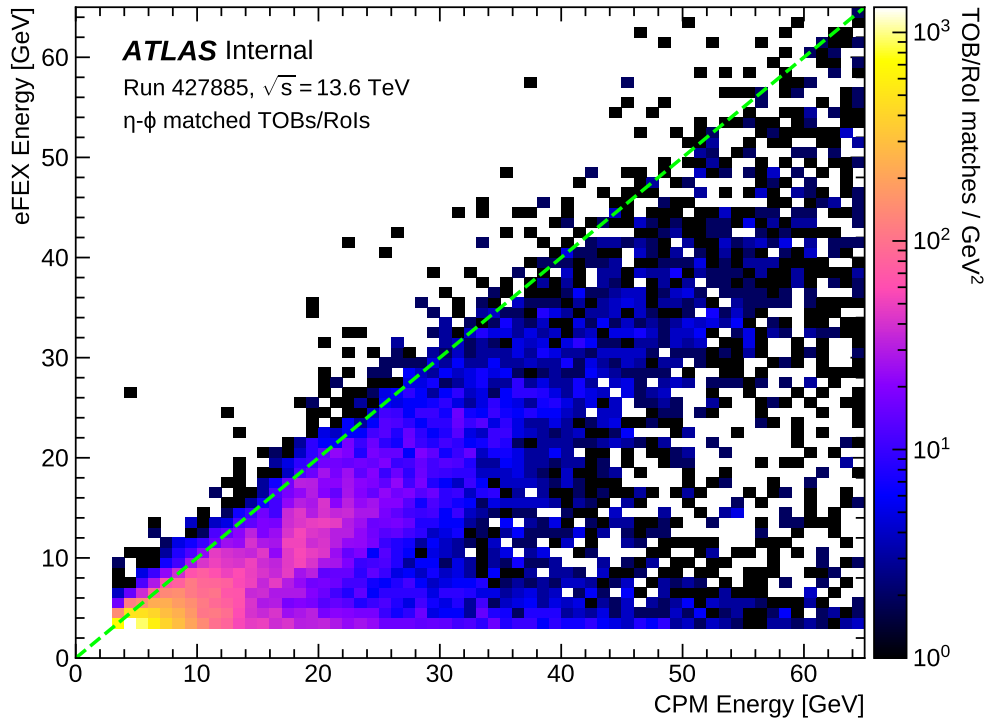


Figure 5.16: Comparison of energies for matched TOBs/RoIs with the energy as measured by the CPM given on the  $x$ -axis and as measured by the eFEX on the  $y$ -axis. Contains data for all matched objects in Run 427885. The dashed line marks the set of points where the CPM and eFEX energies are equal.

## CHAPTER 6

---

Search for vector-boson scattering production of a  $Z$  boson  
and a photon

---

## CHAPTER 7

---

Search for triboson production of  $VZ\gamma$  through its  
semi-leptonic decay mode

---

## CHAPTER 8

---

### Conclusions

---



---

## REFERENCES

---

- [1] The ATLAS Collaboration, “Technical Design Report for the Phase-II Upgrade of the ATLAS TDAQ System,” Tech. Rep. CERN-LHCC-2017-020. ATLAS-TDR-029, CERN, Geneva, sep 2017.
- [2] S. Frixione, P. Nason, and C. Oleari, “Matching NLO QCD computations with parton shower simulations: The POWHEG method,” *J. High Energy Phys.*, vol. 2007, p. 070, nov 2007.
- [3] T. Sjöstrand, S. Mrenna, and P. Skands, “PYTHIA 6.4 physics and manual,” may 2006.
- [4] T. Sjöstrand, S. Mrenna, and P. Skands, “A brief introduction to PYTHIA 8.1,” *Comput. Phys. Commun.*, vol. 178, pp. 852–867, jun 2008.
- [5] F. Rademakers, P. Canal, A. Naumann, O. Couet, L. Moneta, V. Vassilev, D. Piparo, S. Linev, G. GANIS, B. Bellenot, G. Amadio, Wverkerke, E. Guiraud, P. Mato, TimurP, M. Tadel, Wlav, E. Tejedor, A. Gheata, J. Blomer, S. Roiser, Marsupial, A. Bose, CristinaCristescu, R. Iseman, X. Valls, O. Shadura, K. Albertsson, B. P. Bockelman, and O. Zapata, “root-project/root: Patch release of v6.18 series,” sep 2019.





## APPENDIX A

---

### FIRST APPENDIX

---