Summer Student Report:
Software Validation of the Level 1 Track Trigger for the CMS Phase 2 Upgrade

Tobias Orthen

September 13, 2013

Introduction

The Large Hadron Collider (LHC) is the world's largest and most powerful particle accelerator [1]. Its ring-shape allows scientist at CERN to accelerate two proton beams at almost the speed of light in opposite directions. At four points along the circle the beams cross each other, the protons collide and the physical data of the collisions at these intersections are measured by four different particle detectors: ATLAS, CMS, ALICE and LHCb.

The teams of the two largest detectors ATLAS and CMS chose different technical approaches to meet the same physical goals like the discovery of the Higgs-Boson or hints for the existence of Supersymmetry. ATLAS stands for A Toroidal LHC Apparatus and the acronym CMS means Compact Muon Solenoid.

Since the first run of the LHC in 2008, the amount of the collected data is great enough to have good statistics in terms of data analysis under the given conditions of the experiment. But the potential of the LHC is not yet exhausted.

The energy per particle beam will be increased from 3.5TeV to 7TeV to reach a collision energy of 14TeV. Also the density of the particle bunches will be increased to double the luminosity which will then lead to a higher collison rate. The preparation time of the LHC and the detectors to achieve a high performance and the runs under these conditions themselves cover the Phase 1 Upgrade period from 2015 to presumably 2023.

The Phase 2 Upgrade from 2023 on, aims for increasing the luminosity by a factor of ten. This a completely new order of magnitude which needs new detector designs to deal with rapid increase of radiation which comes along with this upgrade. The expected collision rate will produce a huge amount of data that needs to be filtered to be, at all, able to process and store the physically interesting data.

All detectors at the LHC are divided into several subdetectors facing the detection of different kinds of particles. The Tracker is the one closest to the collision chamber to reproduce to trajectories of the particles. The Tracker is like a camera imaging what happens during the collisions. To distinguish between interesting events and background signals, a complex software package is needed because the interesting events rarely occur and the background is great. The novelty of the Phase 2 Upgrade is that the Level 1 Track Trigger will be hardware based since the data amount would be too big to be processes by software otherwise.

In particle physics, everything what happens while an experiment is running is also simulated with software. The comparison between simulated data and real data is the only way to understand

[1] http://home.web.cern.ch/about/accelerators/large-hadron-collider

what is going on during an experimental run. Every existing part of the detector is geometrically correct implemented in the simulations and all known physical processes are calculated. Thus, the simulations for the CMS Level 1 Track Trigger for the Phase 2 Upgrade are developed right now. This software needs to be validated and it has to be proved if the system of the simulations can be implemented into electronics. My project as a Summer Student at CERN involved both.

Summer Student Project

The Phase 2 Tracker with the BarrelEndcap geometry will consist of 10 layers of silicon detectors in toroidal arrangement of the barrel and 5 layers of silicon detectors for each endcap. Each layer is composed of detector modules. There are two module types existing: The module which is used for the inner four barrel layers is the PS module that is made up of one strip sensor and one pixel sensor. The other layers of the barrel and the endcap consist of 2S modules which are made up of 2 strip modules.

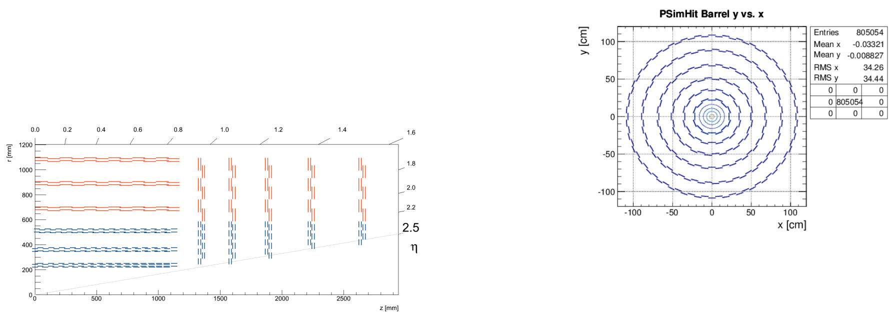

Figure 1: Tracker Geometry BE5D for the Phase 2 Upgrade of the LHC. The profile of the x-y- plane on the right shows the different detector layers of the barrel. The left image is a profile along the y-axis which also shows the endcap of the tracker.

When a particle passes the silicon modules it deposits a charge in strips and pixels surrounding the hit which is recorded by the electronics. These charged areas coming from the passing particle are called clusters. Two clusters from the same particle in one and the same layer are called a Stub.

Since charged particles have bended trajectories because of the magnetic field of the solenoid it is possible that the position of the particle-imposed clusters are displaced in different layers. More important is a displacement between two modules a layer consists of because these clusters form a Stub. The lower the mass of the particle the lower is the transverse momentum P_T of the particle. The lower the P_T the higher is the bending of the trajectory and therefore the displacement increases. Background particles and events of lower interest of a collision typically have low transverse moment. So large displacement of clusters is a indicator for background events.

So the trigger decides which Stubs will be stored for data analysis and which Stubs are deleted. The Level 1 Track Trigger hardware is expected to sort out particles with P_T less than 2GeV.

The first task of my project was to extract information from a simulation to provide them to a PhD student in electrical engineering. In his work he estimates if it is possible to implement the mechanism the Track Trigger works into hardware. If it is possible the results of his calculations should be comparable to the results I extracted from the simulation. I used the CMS software collection (CMSSW) to run a simulation. At first I used the Event Generator and used the output as basis for the next simulation which would give me detector response. Afterwards I modified an existing Event Data Analyzer, written in C++ by Nicola Pozzobon, to produce a text file containing the exact position of Hits and Stubs in the tracker.

This was my first contact with CMSSW as well as the programming language C++ and applied Object Oriented Programming (OOP). Thanks to my course "Systematic Programming" I already new the general concept of OOP but I never applied it before. As a reference to teach myself the skills that were needed I used the CMSSW Workbook, its TWiki pages and for learning the basics of C++ I used the book "Sams Teach Yourself C++: In 21 days" from the CERN library. I also used several other references on the internet.

As a second part I got involved into the validation processes of the Phase 2 Tracker Simulations. As mentioned before we expect many particle trajectories (referred to as SimTracks) with low P_T and less SimTracks for higher P_T. For Stubs we expect a cut at 2GeV. Momenta smaller than 2GeV are not relevant – this is the background we do not want to keep in storage. It has its origin in weak collisions where particles interact but do not collide head-on. Head-on collisions are often decribed as hard collisions.

The distribution for Stubs with $\mathrm{P_T} > 2\mathrm{GeV}$ is expected to have the same shape as the SimTracks have.

Can we see the expected distributions in the simulation data? I compared the plots in Fig.2 using ROOT. Since I did not used ROOT before I used the online Tutorials and the documentation to learn how to use it. I got help and advice from my supervisor as well as from other summer students. To get the distribution of the Stubs I used and existing two dimensional histogram StubPt versus SimTrackPt and did a projection to the y-axis (cf. Fig2). To see if this method works I also projected to the x-axis to get the SimTrack distribution and compared this diagram with an original histogram for the SimTracks which was produced by the C++ file by Nicola Pozzobon from which I started. This comparison is shown in Fig.2.

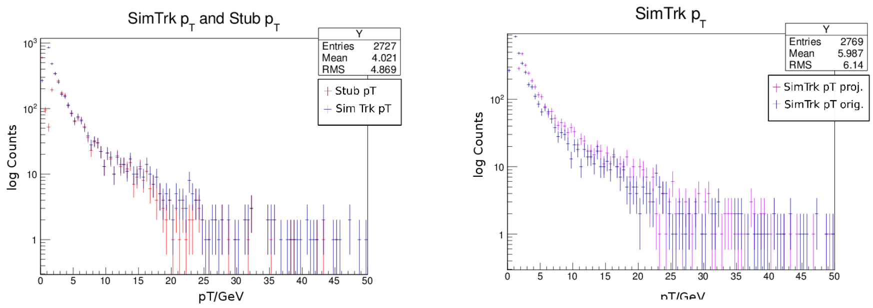

Figure 2: SimTracks and Stubs as they were reconstructed by the validation software. We can see many particle trajectories (SimTracks) in the lower P_T region which come from weak collisions. The distribution decreases for higher P_T since they have their origin in harder collisions and occur less often. The small error bars for low P_T which get larger with increasing P_T is a statistical effect: ROOT calculates the relative error.

The right plot in Fig.2 shows that the method of projection of a two dimensional histogram does not give exactly the same result than producing a one dimensional histogram directly. However it is useful to get a general idea what is going on in the tracker. As expected we see that the number of SimTracks is high for low P_T and decreases with increasing P_T.

The left plot shows a comparable distribution for the Stubs but we can slightly see a decrease of the red distribution. This is the Stub histogram from our projection of a two dimensional histogram which was already produced by Nicola Pozzobon. It needs to be discussed if this rapid decline is the cut we expected to see at 2GeV. An argument against this is the cross for the lowest P_T which is high again. If one takes a look at the two dimensional histogram from Pozzobon in Fig.3 one sees a second bulk of entries next to the y-axis. This could be the reason for the red cross which does not follow the decline in Fig.2.

General Summary

I enjoyed being a Summer Student at CERN. Even if I did not have worked with C++, ROOT or Object Orientated Programming before I got an idea how important simulations are to know what is really going on in a detector. Due to my supervisor I had the possibility to take part in many meetings either personally or

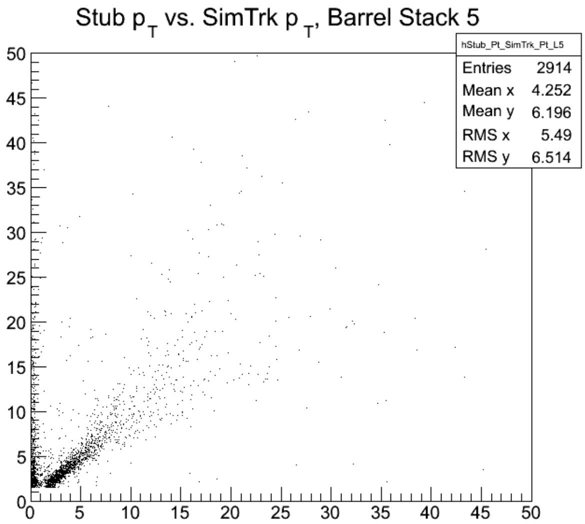

Figure 3: Stub P_T vs. SimTrack P_T from simulation data produced by Nicola Pozzobon. This histogram was the starting point for the projections shown in Fig.2

via VidyoDesktop. This gave me an idea how working in a huge scientific collaboration is structured and organized. I asked many questions and got an more than just satisfying answer almost every time. Even if they were sometimes not project related.

Thank you CERN for organizing this Summer Student Program, thank you Gaelle Boudoul for a lot of patience and good answers to many questions and thank you to the Summer Student Team for always being there when issues occurred.