Forschung mit Lepton Collidern (FLC) Group
Deutsches Elektronen-Synchrotron (DESY)

National Research Nuclear University MEPhI
(Moscow Engineering Physics Institute)

Monitoring of physics performance of ILC Software based on Higgs Recoil Mass

Elena Volkova

Supervisor: Voutsinas Georgios Gerasimos

Abstract

This report is about a part of ILC software development. The goal of this study was to create a code, that allow us to make automated testing of ILC results. The code uses the result of generation, Mokka simulation and Marlin reconstruction of ILC events. The codes task is automated Higgs recoil mass analysis everyday and to compare Higgs recoil mass with one of the previous day result.

Hamburg, 8 September, 2015.

Contents

Chapter 1

Introduction

This report describes the part of ILD detector simulation and modeling. My project has to do with the development and maintenance of the software used in ILC. A key task in software development is the automated testing on different levels of the process: at compile and build time, at runtime and on the results level. The first testing tasks are more technical, while the latter one relies on physics and statistics understanding.

The ILC software will consists of ≈60 packages, developed by different people in different institutes. My task was to write a code, which will allow us to monitor the physics performance of the software.

The physics performance will be monitored via the simulation study of the Higgs boson production for processes in which the Higgs is produced together with a well measurable di-muon system using the current proposal of the ILD detector. The study was conducted based on Marlin and Mokka modeling of ILD.

The goal of this work is to reconstruct the Higgs recoil mass of two independent samples and compare the results. The code can be included in the highly builds, comparing everyday the obtained value of the Higgs recoil mass with the ine of the previous day. This way it can probe differences of the software performance. It can be also used for the ILD detector optimisation. This code will be included in the general programm for checking detector simulation.

Chapter 2

ILC

The International Linear Collider will collide electrons and positrons at energies of initially 500 GeV, upgradeable to 1 TeV(Figure 2.1).
Precision physics at the ILC requires that the beam parameters are known with great accuracy. The interaction region of the ILC is designed to host two general purpose Detectors: International Large Detector and the Si-Detector. ILD will consist of a multy-layer pixel-vertex detector(VTX), Si strip detector(SIT), a large volume time projection chamber(TPC), a highly segmented ECAL and HCAL. In addition, the detector will be equipped with a system of high precision, radiation hard, calorimetric detectors in the very forward region(LumiCAL, BCAL, LHCAL).
Each of the two ILC detectors can be moved into the beam position with a

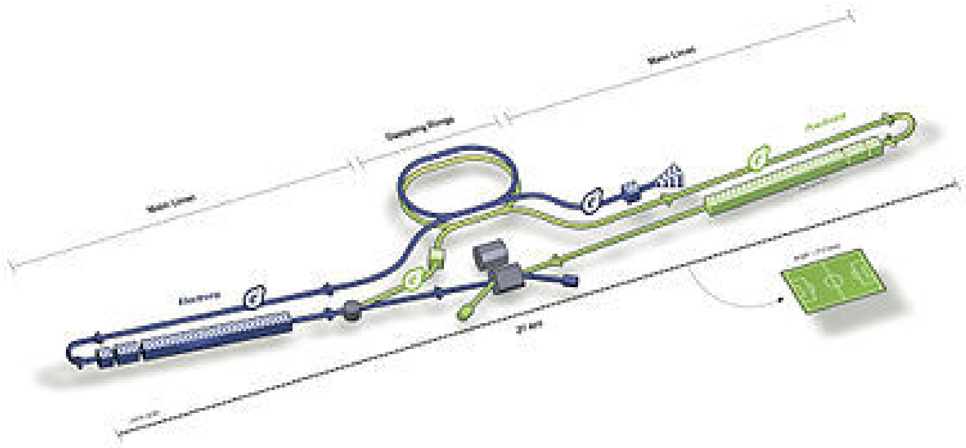

Figure 2.1: The International Linear Collider

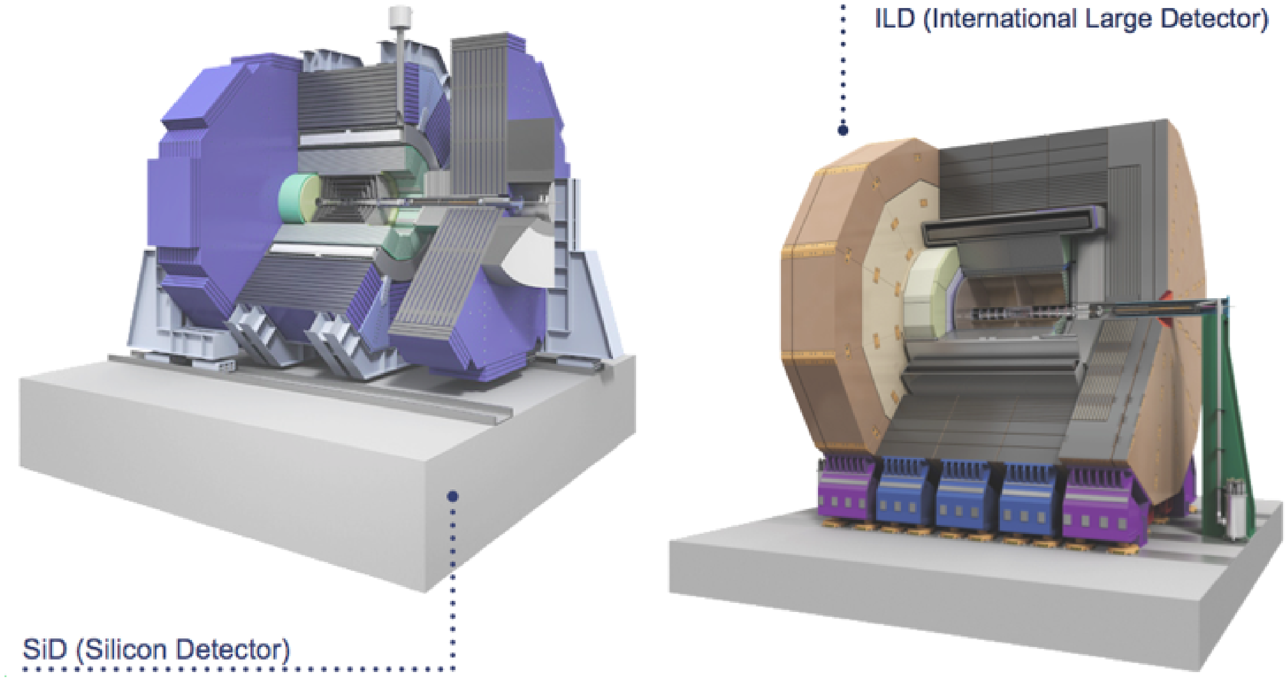

Figure 2.2: The ILD and SiD detectors schimes

"push-pull" scheme. To enable the operation of the detector in a "push-pull" scenario, the complete detector is mounted on a movable platform, which can move sideways out of the beam to make space for the second detector in the interaction region. The platform ensures that the integrity and calibration of the detector is minimally disturbed during the moving process, making the re-commissioning of the detector after the "push-pull" operation easier. The ILD and SiD detectors schemes are shown in Figure 2.2. More information about ILC you can find in [4].

ILD Software.

The software framework of ILD is based on the LCIO persistency format and event data model. The detailed simulation of the detector response is performed by GEANT4 based Mokka application. The overall detector is then built from individual subdetectors, making it relatively straightforward to compare different technology choices.

The Mokka simulated events are processed in Marlin. Marlin is a modular C++ application framework which supports plug-in modules which can be loaded at runtime. This plug-in-based design support the distributed development of reconstruction algorithms and also allows comparison of different algorithms at runtime, e.g. it's possible to run two tracking algorithms producing parallel collections of reconstructed tracks.

Event reconstruction is performed with the MarlinReco package. This consists of a set of models for digitisation, track finding, track fitting, particle flow reconstruction, and flavour tagging.

A key task in ILC software development is the automated testing on different levels of the process: at compile and build time, at runtime and on the results level.

The ILC software consists of ≈60 packages, developed by different people in different institutes. My task was to write a program, which will allow us to monitor the physics performance of the software using Higgs decay data.

Chapter 3

Higgs Recoil Mass

The relevant process for the present study is the recoil reaction $e^+e^- \rightarrow HZ \rightarrow Hf\bar{f}$ (where f=leptons and quarks), also called Higgs-strahlung. I chose the $HZ \rightarrow \mu^+\mu^- X$ because the track quality for muons is better and the Z-boson mass peak is more clear than for electrons [1].

All data analysed in this report have been produced by the GEANT4 generation (E_{cms} = 250 GeV), Mokka detector simulation and Marlin reconstruction. The generated Higgs boson mass is $120 GeV/c^2$. 500 fb^{-1} have been reconstructed with use Marlin software, but in the beginning I worked with part of all statistics, namely with 2000 events. I used two different reconstruction algorithms: DBD tracking and cellular automaton.

The first step of analysis was muon tracks selection and recovery of Z-boson from two muons, and the result is shows in Figure 3.1.

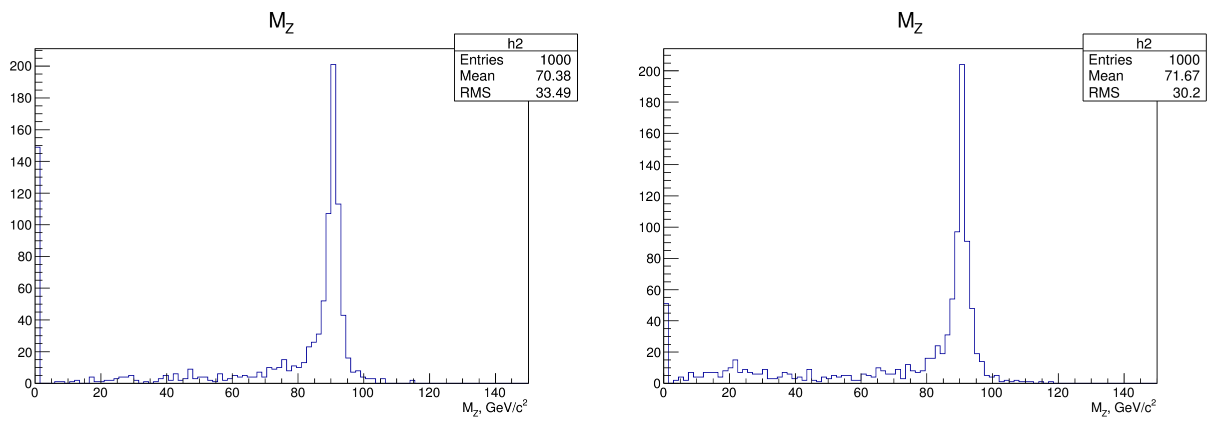

Figure 3.1: Z-boson mass of two data parts

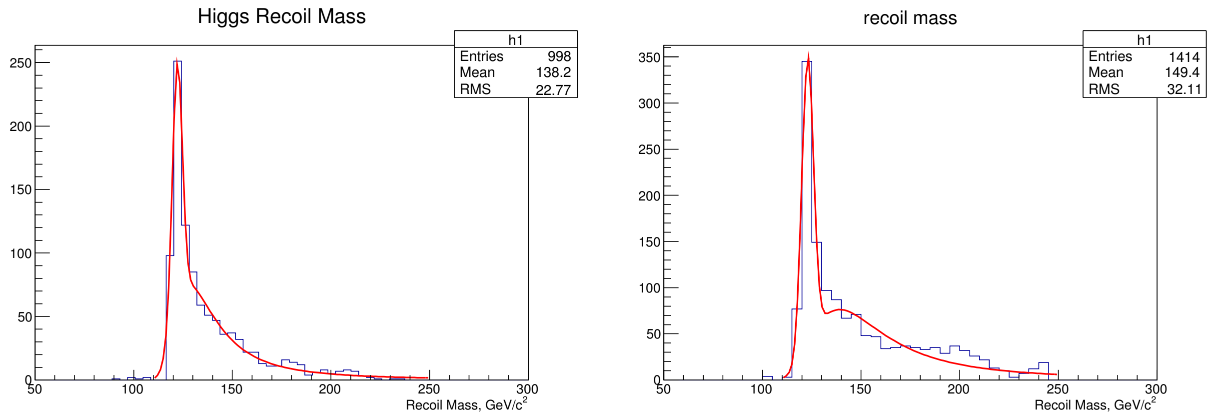

Figure 3.2: Higgs Recoil Mass without Z-boson select

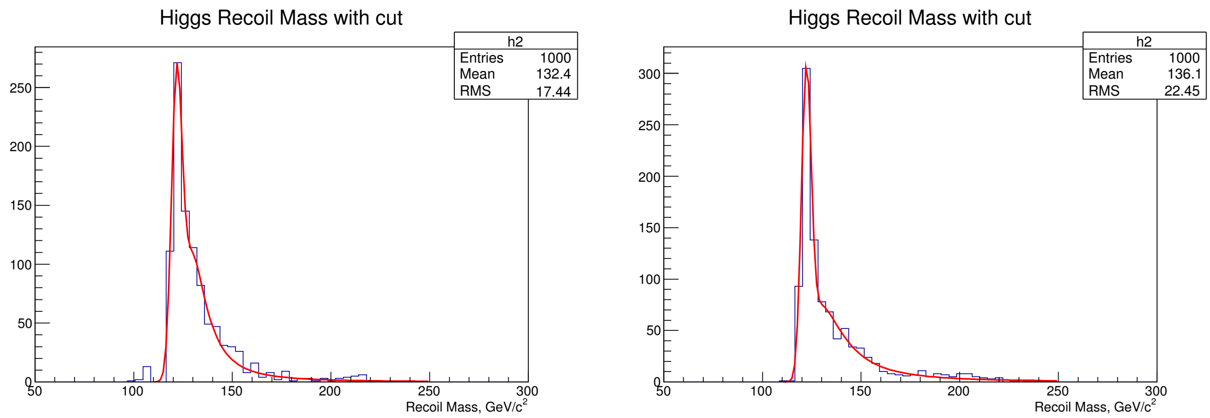

Figure 3.3: Higgs Recoil Mass with Z-boson select

Then I selected one muon pair from each event for peak improvement. The idea of selection is to choose a muon pair, that has the most close mass to the well known Z-boson mass [2]. The results for Z-boson without muons selection are shown in Figure 3.2. And the results for Z-boson mass with muons selection are shown in Figure 3.3.

The Higgs recoil mass was calculated via the relative kinematic equation:

$$M^2_{recoil} = s + M_Z^2 - 2E_Z\sqrt{s}, \tag{3.1}$$

The energy of the incoming beams is smeared with an energy spread of 0.3%, that is a one of the sources of tail. The another one is initial state radiation and beamstrahlung [3].

Fit function for resulting spectra consists of Gaussian for the Peak with Landau

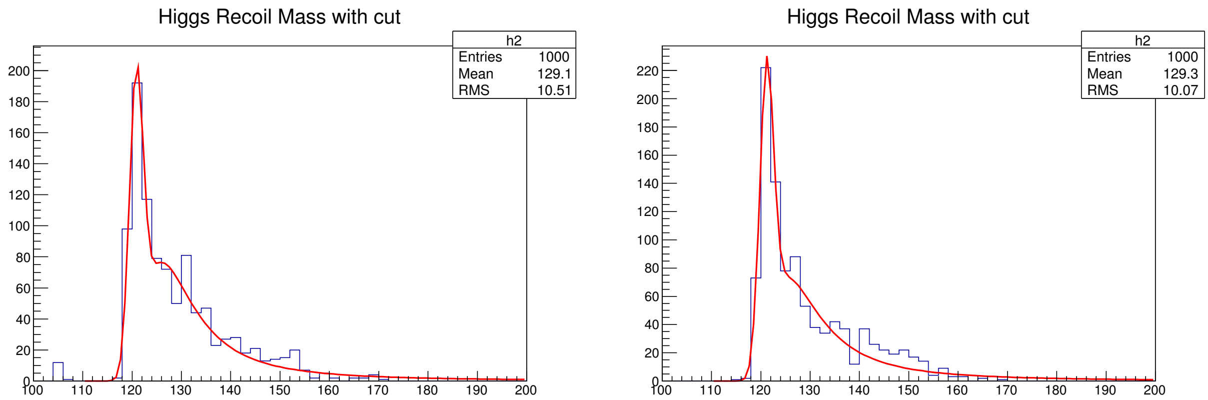

Figure 3.4: M_{reco} =120.8±0.2 for first case, 121.14±0.12 for second case

function for tail. I tried use different functions for tail fit: exponent, polynomial, but Landau function describes better for describes the distribution.

Then for peak improvement I made $70 < M_z < 110$ cut, the result on Figure 3.4.

Chapter 4

Outlook

The code was used to analyse and compare two independent samples. The results of the program is shown on table, total error = 0.5, the difference of values = 0.3.

Mean	Err
121.87 GeV/c^2	± 0.46
122.17 GeV/c^2	± 0.19

The result for parts with 5000 events is total error = 0.11 The next step is to

Mean	Err
121.05 GeV/c^2	± 0.07
121.08 GeV/c^2	± 0.06

improve the fitting of the distribution. After that, the code can be included in the ILC software diagnostic tools.

Bibliography

[1] K. Ito *et al.*, *"HZ Recoil Mass and Cross Section Analysis at ILD"* LC-PHSM-2009-006 arXiv: 1202.1439

[2] J. Beringer *et al.* (Particle Data Group), Phys. Rev. D **86** (2012) 010001

[3] T. Barklow and P. Chen *"Beamstrahlung Spectra in Next Generation Linear Colliders"* SLAC-PUB-5718 Rev. DAPNIA/SPP 92-02

[4] K. Fujii *et al.*, arXiv:1506.05992 [hep-ex].