Università di Pisa

Facoltà di Scienze Matematiche, Fisiche e Naturali

Corso di Laurea Specialistica in Scienze Fisiche

Anno Accademico 2008/2009

Tesi di Laurea Specialistica

Improving Online Track Reconstruction at High-Luminosity Colliders: the Gigafitter Upgrade at CDF

Candidata:
Martina Bucciantonio

Relatore:
Prof. Mauro Dell'Orso

Contemporary physics is sometimes based
on concepts similar to the smile
of a cat that is not there.

Contents

Introduction

Experiments in hadron collider high energy physics (HEP) have grown to be very ambitious in recent years searching answers to many open questions of fundamental physics. To reach these goals the new experiments deal with very high energy collisions, very high event rates and extremely precise and huge detectors. The data coming from the detector have to be extracted by a complex set of systems that must be able to select a rich variety of interesting but extremely rare events from an overwhelming background and to decide whether or not that event has to be recorded on tape. This kind of system is called "trigger". Because the collision rate is much higher than the rate at which data can be stored on tape, the role of the online trigger is to efficiently select the most interesting physics events.

Charged particle tracking is a very rich source of information, in fact it's a major technique in offline analysis where most sophisticated algorithms were developed. In the capter 1 I will introduce the tracking in HEP to understand how critical is this task.

The SVT (Silicon Vertex Trigger) at CDF at Tevatron has been a pioneer in this field allowing the collection of high quality data. I will show the trigger system at CDF in the chapter 3, after a description of the Tevatron accelerator complex and the CDF experiment itself (chapter 2). As the Tevatron performances improved, SVT showed the need of more powerful computing power. The high degree of organization and standardization inside the CDF trigger and the SVT system allowed a very quick upgrade.Infact, SVT has proven that a properly designed hardware can be flexible enough to prospect upgrades in order to further enhance its capabilities and to cope with increasing detector occupancy.

In order to overcome SVT limits during the final period of CDF data taking,the very last upgrade is again necessary: the Gigafitter (GF), the object of this thesis. The Gigafitter consists in a single board that replaces a complex system made by 16 boards of SVT. The GF allows the SVT processor to deal with the highest luminosity of the Tevatron collider increasing the track reconstruction efficiency. The new system must be also faster than the old one, especially at high luminosity when events are complex and many candidate tracks must be

fitted and evaluated.

During my thesis work I have written the basic firmware of the Gigafitter, that is the part executing the fit operation, and I have contributed to test the GF system for commissioning it at CDF. In chapter 4 I will show which is the logical structure of the Gigafitter in all its parts, I will describe the hardware and the functions that I have implemented in the firmware, also presenting some of the simulation that has been performed to certify the proper functioning of the system.

I have also analyzed the effects and performances of SVT with the Gigafitter board in order to show how the SVT system benefits from the GF board, as shown in chapter 5. At first, the Gigafitter is being installed with exactly the same constant and pattern sets of the current system, therefore I have checked its performance, against the present SVT track fitter system, in terms of the track parameters quality, resolution, purity, efficiency, and timing.

Chapter 1

Tracking in High Energy Physics

Experiments in hadron collider high energy physics (HEP) have grown to be very ambitious in recent years. The upgraded Tevatron at Fermilab and even more the Large Hadron Collider (LHC) at CERN are in an excellent position to give conclusive answers to many open questions of fundamental physics: for example the existence of supersymmetric particles, of the Higgs boson, and the origin of the CP asymmetry. To reach these goals the new experiments deal with very high energy collisions, very high event rates and extremely precise and huge detectors.

Along with the development of new accelerators and detectors, the algorithms and processors for selecting, collecting, and analyzing data and extract useful information also need to evolve and to become more powerful. Developing algorithms for online selection of events is a crucial step to fully exploit the potential of new experiments.

At the CDF (Collider Detector at Fermilab) experiment the Tevatron collision rate is about 2 MHz. With an instantaneous luminosity of 3×10^{32} cm^{-2}s^{-1} the average number of interactions per bunch crossing (pileup) is 6 [1]. The rate at which events can be written on tape is about 100/s.

At the new LHC experiments the problem is even harsher: the collision rate is 40 MHz and, at an instantaneous luminosity of 10^{34} cm^{-2}s^{-1}, there are 25 pileup interactions, while the rate at which events can be written on tape is still about 100/s (the storage technology is faster at LHC but events are bigger).

The complex set of systems, which analyzes the data coming from the detector, extracts the useful information, and makes the decision about whether or not that event is to be recorded on tape, is called the "trigger". The trigger system must perform a very stringent selection, reducing the rate of events of several

orders of magnitude. This selection must be as sophisticated as possible in order to suppress background events while saving sparse signal events.

To understand how critical is this task, we can look at the figure 1.1: the total rate of produced events at LHC baseline luminosity is about 10^9 every second and only 100/s can be written on tape. However, we must be sure to write, among them, a possible Standard Model (SM) Higgs of 115 GeV decaying in two photons that is produced roughly every hour, as well as the rarer SM Higgs of 180 GeV decaying into four leptons, and so on. The trigger must be able to select a rich variety of interesting but extremely rare events, each one with its peculiar detector data, while rejecting the overwhelming amount of uninteresting events, which can mimic these detector responses.

A very important part of the trigger is the one that reconstructs charged-particle trajectories in the tracking detector: with this knowledge it is possible to make very sophisticated and powerful selections. Often, the information from the tracking detector is not used at its best at trigger level because of the big amount of data to process (the tracking detector is usually the one that produces most of the data) and the difficulty of trajectory reconstruction. Actually, modern hardware along with clever algorithms allow us to fully exploit tracks in the trigger environment: indeed, one of the purposes of this thesis is to give further support to the claim.

1.1 Tracking at CDF

For triggers based on reconstructed tracks, the CDF experiment followed the strategy to employ powerful dedicated processors in order to make complex and precise algorithms fast enough to be put in the first two levels of trigger, the most critical in event discrimination, thus allowing the collection of high quality data. The SVT processor at CDF has been a pioneer in this field.

1.1.1 The SVT development

SVT was installed in 2000 providing in "world premiere" an hadron collider experiment with offline quality tracks for the trigger decision algorithm.
In general, dedicated hardware is considered powerful but usually difficult to upgrade and not flexible. SVT has instead proven that a properly designed hardware can also be flexible enough to prospect upgrades in order to further enhance its capabilities and to cope with increasing detector occupancy. In 2003, when the Tevatron performances started to improve, SVT showed the need of greater

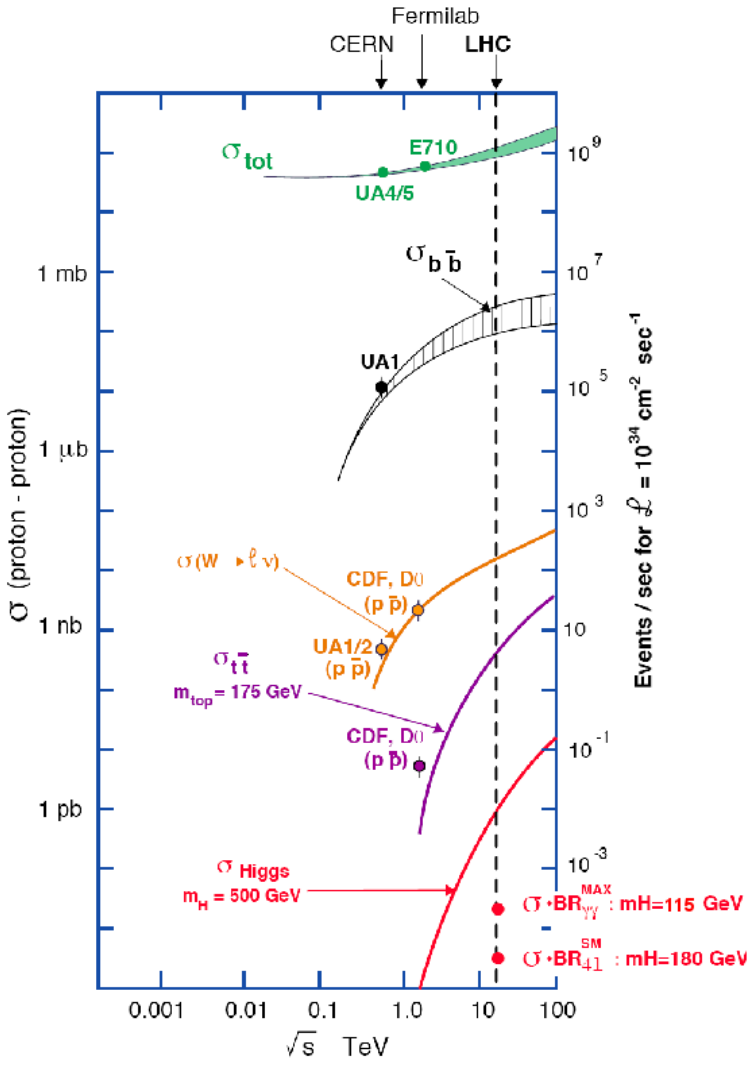


Figure 1.1: LHC Cross sections of various signals. It's shown the expected rate of events and the relative cross sections between various kind of signals and background events at the LHC baseline luminosity of 10^{34} cm^{-2}s^{-1}.

computing power. No SVT upgrade was planned in advance. Potentially, difficulties of an unpredicted upgrade could deprive CDF of its most succesfull trigger capability, which was quickly becoming obsolete. However, the high degree of organization and standardization inside the CDF trigger and SVT system allowed a very quick upgrade: in spite of the very complex function it performs, SVT was upgraded in just 2 years. The SVT upgrade commissioning took place in the summer 2005 while the experiment was taking data. A phased installation

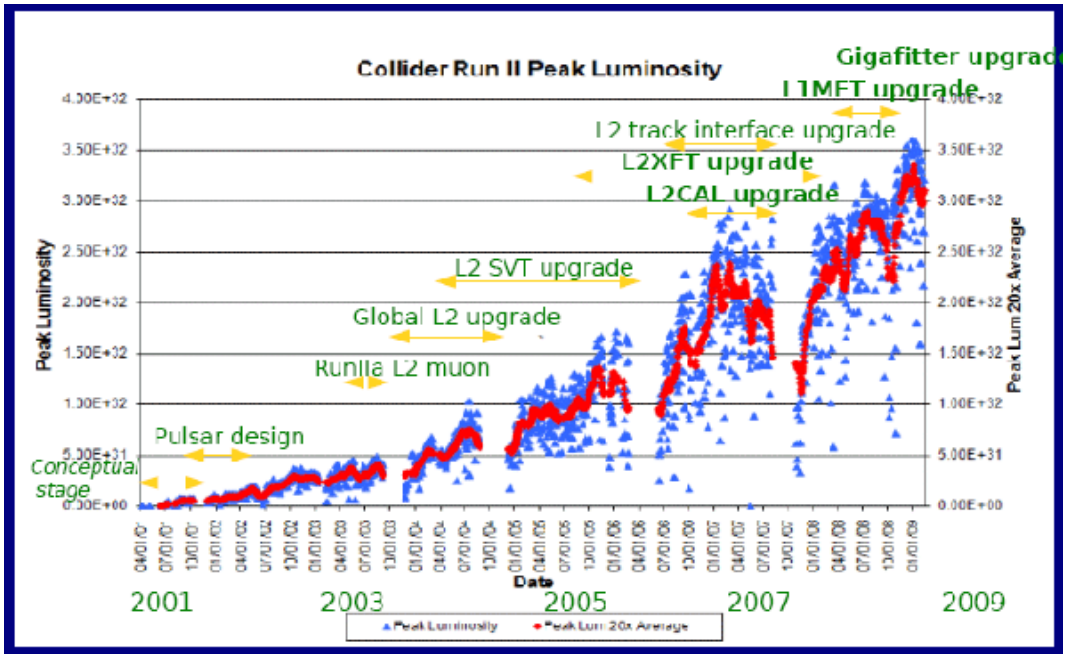


Figure 1.2: This figure shows the various upgrades programs that CDF made at the trigger system to adapt to increasing luminosity. The performance of the accelerator has steadily increased over time and it's foreseen to be able to beat the 3.5×10^{32} cm^{-2}s^{-1} record before the end of operations in 2011/2012. Many of these upgrades were unpredicted and exploited the experience and method used during the successful SVT upgrade of 2006.

was chosen: boards were replaced gradually, exploiting the short time between stores[1]. This phased procedure allowed for quick recovery if there were failures, since each small change was immediately checked before going ahead.
The power added to the experiment without any risk for the data taking convinced the collaboration to proceed with other important unpredicted trigger hardware upgrades, in order to fix problems caused to the trigger by the increasing Tevatron performances (shown in figure 1.2). The very last upgrade is again for SVT and consists of the Gigafitter, the object of this thesis.
The Gigafitter will allow the SVT processor to deal with the highest luminosity of the Tevatron collider.

The effect of each upgrade can reliably be estimated by comparing SVT pro-

[1]A store is the period when the Tevatron accelerator is making collisions for HEP experiments. Its duration depends on initial luminosity of the store and accelerator status. During the SVT upgrade was about a day with few hours between stores, now it's typically 10-12 hours with less than one hour between stores.

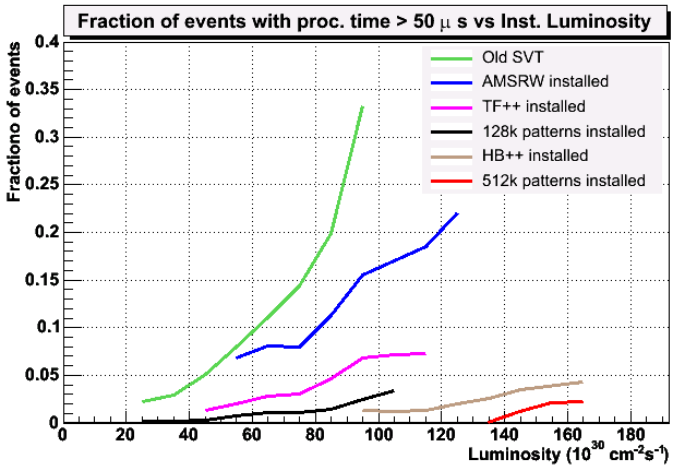


Figure 1.3: The phased installation of new SVT hardware allowed to contain the tails of the SVT processing time along with the Tevatron peak luminosity increase.

cessing time before and after the upgrade at the same luminosity, with the same trigger path mixture. Figure 1.3 shows the improvement on the fraction of events with processing time above 50 μs as a function of luminosity, for different stages of the SVT upgrade. This fraction of events is interesting because over-threshold events directly and substantially contribute to the trigger dead-time.

The first improvement, installing the AMS/RW boards [2], reduced processing time by reducing the number of track fit candidates and reducing the pattern recognition time. The second upgrade, installing the upgraded Track Fitter (TF++) board [3] (described in sec. 3.2.6), significantly reduces the fraction of over-threshold events by speeding up the track fitting process with faster clocks and a six-fold increase in the number of fitting engines on the new board. Next, the use of 128K patterns reduces the number of fit combinations per recognized pattern. The upgraded Hit Buffer (HB++) [4] further increased the processing speed by virtue of the faster clock speed on the upgraded board. Finally, the full power of the upgrade is visible after enabling all 512K patterns [5]. The fraction of events over threshold is well below 5% at the highest luminosities available for these tests. Data taking without an upgraded SVT system at these luminosities would clearly suffer huge rate penalties, as the corresponding fraction of events over threshold is roughly 25% at half the maximum tested luminosity, with a steeply rising tendency. The SVT upgrade has been a clear success in the working range of luminosities. Since the fully upgraded system shows a very little

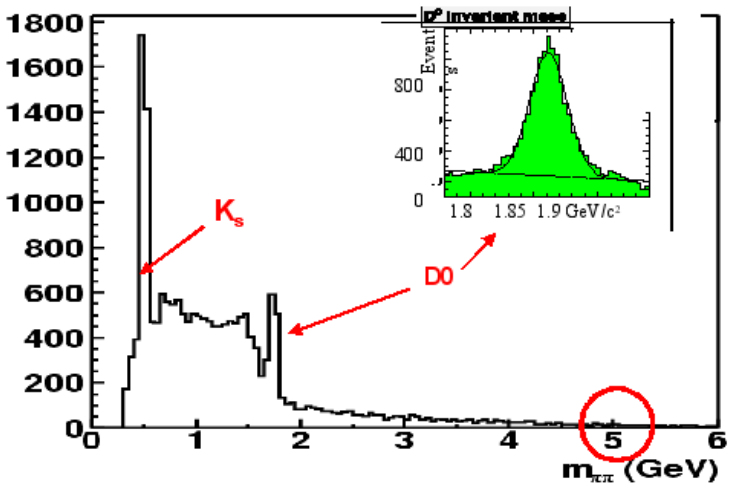


Figure 1.4: Collecting data with Two Track Trigger (SVT), the D^0 mass peak can be used online to monitor the trigger.

dependence on the instant luminosity (the dependence is very near to be flat), the final system later proved effective also for higher luminosity.

1.2 Tracks in trigger: an example

SVT had an extremely significant impact on the CDF physics program. It has been essential for the B_s mixing frequency measurement [6], and for the first observation of the rare charmless decay modes of the B_s [7] and Λ_b that complement the existing "Beauty Factories" information on B_d charmless decays [8]. These extremely challenging measurements and first observations would have been by far out of the CDF reach without the SVT. Severe constraints on several extensions of the SM already arise from the new CDF measurements and will become even more stringent when the precision of the theoretical calculations will reach the current level of experimental accuracy.

1.2.1 Two-Track Trigger

The most revolutionary use of tracks ever seen at an hadron collider, at both level 1 and level 2, is certainly the CDF Two-Track Trigger (TTT). It works on tracks above 1.5-2 GeV. Figure 1.4 shows the Two Track Trigger power.

It shows the online invariant mass distribution of track pairs with large impact

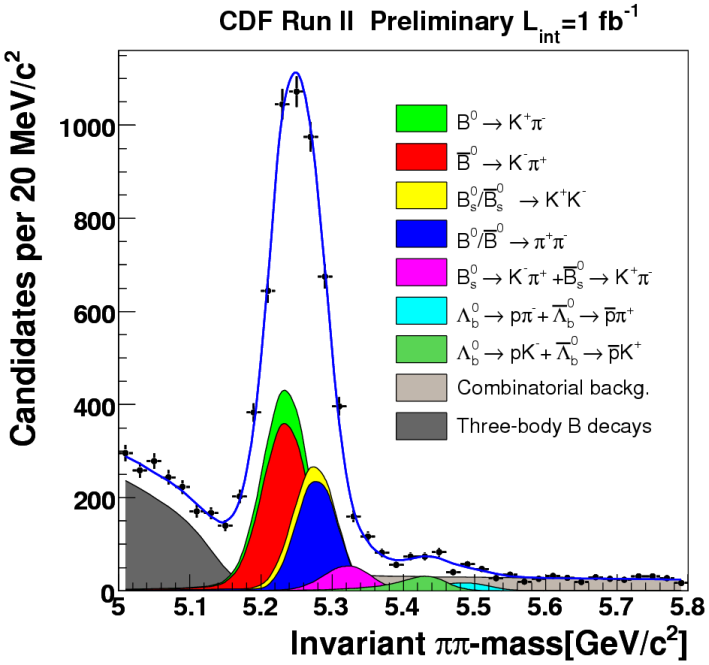


Figure 1.5: The figure shows the $\pi\pi$ invariant mass distribution with reconstructed $B^0 \to hh$ decays. Data was collected using SVT based trigger.

parameter with the contributions from known particle dacays. It is worth to point out that, before SVT were designed, 2-track invariant-mass peaks were universally considered unobservable by offline analyses of hadron collider data, let alone the online. Nowadays, we even online monitor the SVT efficiency run by run with the online reconstructed D^0 signal. The 5 GeV/c^2 region shows a very low background level. With 1 fb^{-1} of data CDF has reconstructed a striking $B^0 \to hh$ signal (figure 1.5) [9]: an excellent example of the concrete possibility of reconstructing rare and "background-looking" signals, when a high-performance

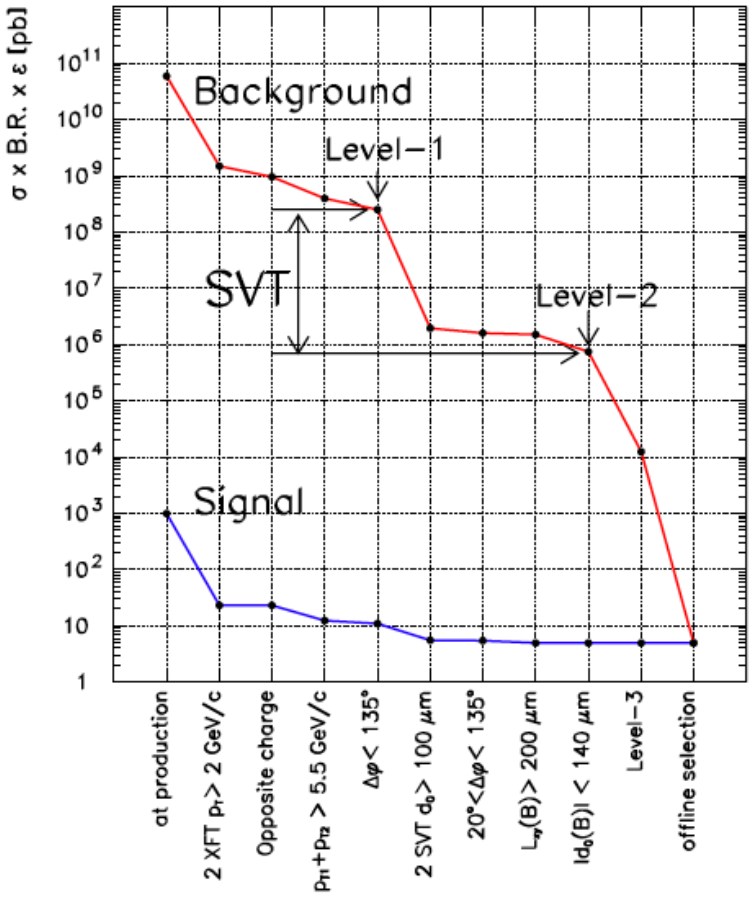


Figure 1.6: The effect of the various selection cuts applied for the $B^0 \to hh$ analysis on both signal (red) and background (blue).

trigger and a sophisticated offline analysis are combined. The plot in figure 1.6 shows how much background would cover the Ks, D^0 and B peaks if the CDF tracking detectors were not used for the trigger selection. The plot shows the background (blue, measured on data) and the $B^0 \to hh$ signal (red) cross section as a function of the applied selection criteria [10]. The request of two XFT tracks at L1 and of two SVT tracks with large impact parameter at L2 reduces the level of background of several orders of magnitude, while keeping the efficiency on the $B^0 \to hh$ signal at a few percent level. The purity of the selected sample is enormously increased. Since the B-physics has a limited rate budget, the better purity allows CDF to increase by several orders of magnitude its efficiency for the hadronic B decay modes.

Historically, B-physics events have been selected at hadron colliders by trig-

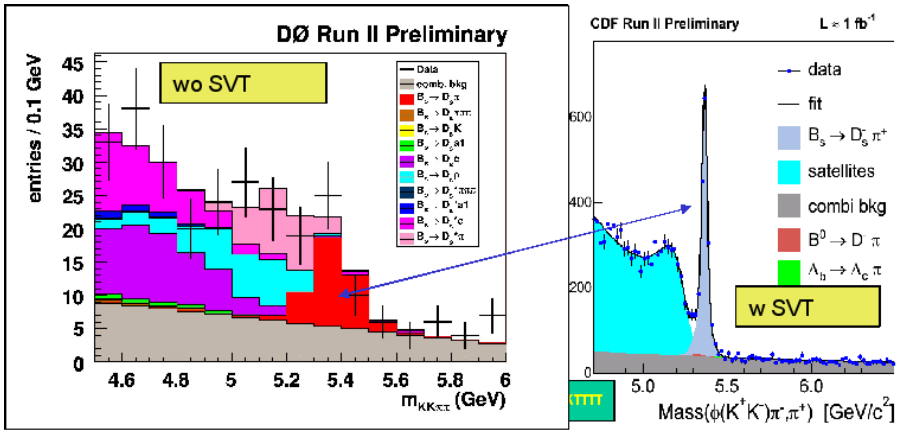


Figure 1.7: *DØ* and CDF, both Tevatron experiments, have published an analysis on B_s oscillation. CDF had the advantage of much more events collected by its trigger.

gers based on lepton identification. Trigger selections based on the reconstruction of secondary decay vertices increase the b-quark identification efficiency and allow collecting otherwise inaccessible hadronic decay modes. The availability of the hadronic decay modes at CDF determined the different quality of the CDF and DØ Bs mixing measurements [11] (see Figure 1.7).

Chapter 2

The Accelerator complex and the CDF detector

The Tevatron [12] is a proton-antiproton synchrotron accelerator hosted at the Fermi National Accelerator Laboratory and producing $p\bar{p}$ collisions at the center-of-mass energy of $\sqrt{s} = 1.96$ TeV. The final proton and antiproton beams are the results of a complex acceleration apparatus which involves different stages, spanning from proton and antiproton production, their acceleration and transfer towards different sub-systems, to their actual collision in designed interaction points where the CDF and DØ particle detectors are installed.

The machine let us examine dimensions up to 10^{-15} m, looking at hadron constituents, the quarks. Two major components of the Standard Model were observed at Fermilab: the bottom quark (May-June 1977) and the top quark (February 1995). In July 2000, Fermilab experimenters announced the first direct observation of the tau neutrino.

In the following sections, after a briefly overview of Fermilab accelerating complex, I will describe the CDF experiment, but first I will spend a few words on a very important accelerator parameter, luminosity.

2.1 Instantaneous and integrated Luminosity

While building an accelerator, a fundamental construction parameter is the design luminosity that needs to be achieved; in fact luminosity is a resource directly related to the computation of the probability $W_{i \to f}$ for a generic process $i \to f$, where i and f are the initial and final states, respectively. In the case of Tevatron, the initial state is made up by two particles, a proton and an antiproton, while the final state is composed by a generic number N of particles. Taking into account the overall four-momentum conservation, the probability amplitude for

the $p, \bar{p} \to f$ process has the following general structure:

$$\langle f|T|p;\bar{p}\rangle = (2\pi)^4\delta^{(4)}(P_f - p_p - p_{\bar{p}})\langle P_f|M|p_p;p_{\bar{p}}\rangle \tag{2.1}$$

We made the following assumptions regarding each particle a in the initial and final states: a is described by a narrow wave packet that obeys[1] the on-shell mass condition, the Klein-Gordon equation and that is peaked around a four-momentum p_a. We obtain the following equation (were we hided all remaining quantum numbers):

$$\mathcal{F}_p^a(x) \equiv \langle x|a\rangle = \frac{1}{(2\pi)^{3/2}}\int d^4q\ \theta(q_0)\delta(q^2 - m_a^2)\hat{f}_p(q)e^{-i\,qx} \tag{2.2}$$

Integrating the square modulus of Eq. 2.1 over its space dependencies and after other manipulations that use approximations allowed by the narrowness of the wave packets, assuming that protons and antiprotons are grouped in bunches, we end up with the following transition probability $W_{i\to f}$:

$$W_{i\to f} \approx (2\pi)^4\delta^4(P_f - p_p - p_{\bar{p}})|\langle P_f|M|p_p;p_{\bar{p}}\rangle|^2\frac{1}{4\omega_p\omega_{\bar{p}}}\int d^4x\ \rho_p(x)\rho_{\bar{p}}(x) \tag{2.3}$$

ω's denote energies. ρ's have the meaning of probability density of particle location and are the time component of conserved four-currents given by:

$$i(\mathcal{F}^*\partial_\mu\mathcal{F} - \mathcal{F}\partial_\mu\mathcal{F}^*) \tag{2.4}$$

The square amplitude in Eq. 2.3 is what can be computed by means of the Standard Model theory. What appears in the integral depends on the experimental setup; the integral itself has dimension of an inverse cross section and is a measure of the probability that incoming protons and antiprotons have to come in interaction. We can assume that the densities ρ are Gaussian near the collision points and that, for simplicity, the collisions themselves are head-on; then, parameterizing the bunches path by s and calling $x(s)y(s)$ a plane orthogonal to the path in s, we can write approximately:

$$\rho^{\pm}(x(s), y(s), s \pm vt) = \frac{N^{\pm}}{(2\pi)^{3/2}\sigma_x\sigma_y\sigma_s}\ exp\left(-\frac{x^2}{2\sigma_x^2} - \frac{y^2}{2\sigma_y^2} - \frac{(s \pm vt)^2}{2\sigma_s^2}\right) \tag{2.5}$$

where $\pm$ refers to proton/antiproton, N is the number of particles in a bunch, v is the speed of the bunches and σ's denote the radii of the portion of the crossing

[1] TO DO

bunches that effectively overlap. In Eq. 2.3 we have consequently:

$$\begin{aligned}\int d^4x\ \rho_p(x)\rho_{\bar{p}}(x) &\equiv \nu\Delta \int dx\ dy\ ds\ dt \rho_p(x,y,s+vt)\rho_{\bar{p}}(x,y,s-vt) \\ &= \nu \frac{N_p N_{\bar{p}}}{4\pi\sigma_x\sigma_y}\frac{\Delta}{2v} \\ &\equiv \mathcal{L}\frac{\Delta}{2v}\end{aligned} \tag{2.6}$$

where Δ is the whole lasting of the data taking, long with respect to the duration of each effective crossing of the colliding bunches, ν is the frequency of the crossing of the proton and anti-proton bunches, and (the lab reference frame is also the center of mass frame in our case)

$$v = \frac{|\vec{p}|}{\omega}, \quad |\vec{p}| = |\vec{p_p}| = |\vec{p_{\bar{p}}}|, \quad \sqrt{m_p^2 + \vec{p}^2} = \omega = \omega_p = \omega_{\bar{p}} \tag{2.7}$$

Thus we have

$$\begin{aligned}\frac{dW_{i\to f}}{dt} &\approx \delta^4(P_f - p_p - p_{\bar{p}})\frac{(2\pi)^4}{2\omega|\vec{p}|}|\langle f|M|p_p;p_{\bar{p}}\rangle|^2\mathcal{L} \\ &= \frac{(2\pi)^4\delta^4(P_f - p_p - p_{\bar{p}})}{\sqrt{(p_p\cdot p_{\bar{p}})^2 - m_p^2 m_{\bar{p}}^2}}|\langle f|M|p_p;p_{\bar{p}}\rangle|^2\mathcal{L} \\ &\equiv \sigma_{int}\mathcal{L}\end{aligned} \tag{2.8}$$

$\mathcal{L}$ is usually called (instantaneous) luminosity, while its integral over time L is called integrated luminosity. The bigger the luminosity, the bigger the probability to observe an interaction. For this reason the Tevatron has undergone a series of improvements during its lifetime in order to increase this fundamental parameter.

In the first period of data taking, from 1992 to 1996 (*RunI*) the accelerator, operating at a center-of-mass energy of 1.8 TeV, delivered to each experiment an integrated luminosity in excess of 160 pb^{-1}([2]). *RunI* was followed by a long shutdown period begun in 1996; during this period both the accelerator complex and CDF detector were improved in order to increase the number of expected events. After detector and accelerator upgrades, since 2001 a second period of data taking has begun (*RunII*).

Detector improvements will be treated later; as far as luminosity is concerned, the number of bunches per beam was increased from 6 to 36 and finally to 108, while maintaining the average number of particles roughly the same as in *RunI*. In fact increasing the number of particles per beam, the main problem one has to

[2] $1barn = 1 \times 10^{-24} cm^2$.

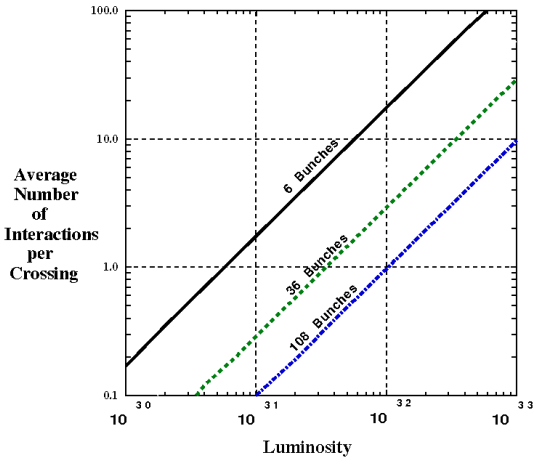


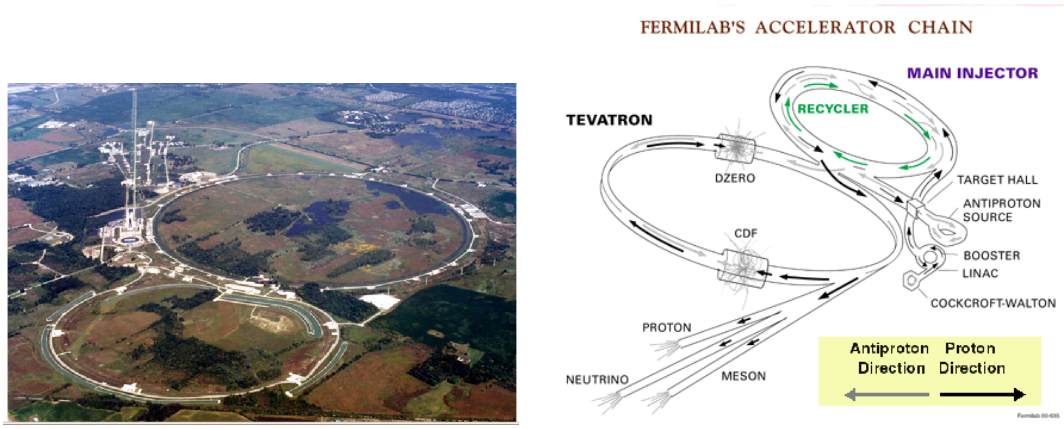
Figure 2.1: Average number of interactions per crossing for different beam conditions: with 108 bunches per beam a greater luminosity can be achieved without increase the number of interactions per crossing.

face with is the superposition of multiple $p\bar{p}$ interactions within the same bunch crossing. If this happens, event reconstruction is much more complicated. So the best thing we can do is leaving the number of particles per bunch unchanged while increasing the number of bunches (see figure 2.1).

2.2 The Tevatron ring

The Tevatron is the last stage of the Fermilab accelerator chain. A schematic view of the acceleration chain is provided in Figure 2.2. The Tevatron is a 1 km radius synchrotron able to accelerate the incoming 150 GeV beams from Main Injector to 980 GeV, providing a center of mass energy of 1.96 TeV. The accelerator employs superconducting magnets throughout, requiring cryogenic cooling and consequently a large scale production and distribution of liquid helium. The Tevatron operates at the 36×36 mode, which refers to the number of bunches in each beam.

The antiprotons are injected after the protons have already been loaded. Just before the antiproton injection a set of electrostatic separators are used to create a pair of non-intersecting helical closed orbits. When the Tevatron loading is complete, the beams are accelerated to the maximum energy and collisions begin. The beam revolution time is 21 μs. The beams are split in 36 bunches organized in 3 trains each containing 12 bunches (see Fig. 2.3). Within a train the time



(a) (b)

Figure 2.2: (a): An airplane view of the Fermilab laboratory. The ring at the bottom of the figure is the Main Injector, the above ring is the Tevatron. On the left are visible the paths of the external beamlines: the central beamline is for neutral beams and the side beamlines are for charged beams (protons on the right, mesons on the left). (b): A sketch of the Fermilab accelerator chain.

spacing between bunches is 396 ns. An empty sector 139 buckets[3] long (2.6 μs) is provided in order to allow the kickers [13] to raise to full power and abort the full beam into a dump in a single turn. This is done at the end of a run or in case of an emergency.

In the 36×36 mode, there are 72 regions along the ring where the bunch crossing occurs. While 70 of these are parasitic, in the vicinity of CDF and DØ detectors additional focusing and beam steering is performed, to maximize the chance the proton strikes an antiproton. The focusing, driven by quadrupole magnets, reduces the beam spot size and thus increases the luminosity, as seen in Eq. 2.7 that shows how smaller values of σ_x, σ_y imply larger luminosity values. During collisions the instantaneous luminosity decreases in time as particles are lost and the beams begin to heat up. Meanwhile, new antiprotons are stored in the Accumulator. When the luminosity becomes too low (approximately after 15-20 hours) it becomes beneficial dumping the current store and start a new cycle.Table 2.1 summarizes the accelerator parameters.

Figure 2.4 shows the Tevatron peak luminosity as a function of the time. The blue squares show the peak luminosity at the beginning of each store. The red triangle displays a point representing the last 20 peak values averaged together. Continous improvements in the accelerator complex led to the rapid increase of

[3]A bucket is a stable region in longitudinal phase space that contanins bunch.

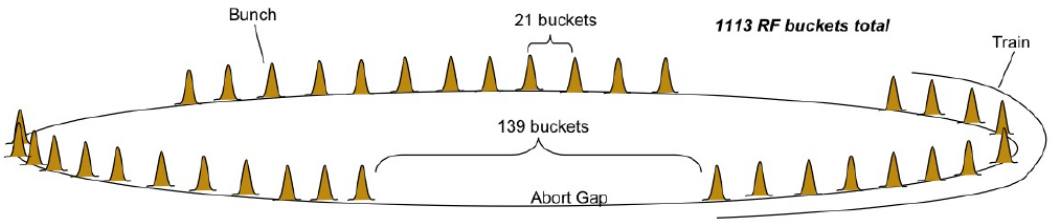


Figure 2.3: Bunch structure of the Tevatron beams in Run II.

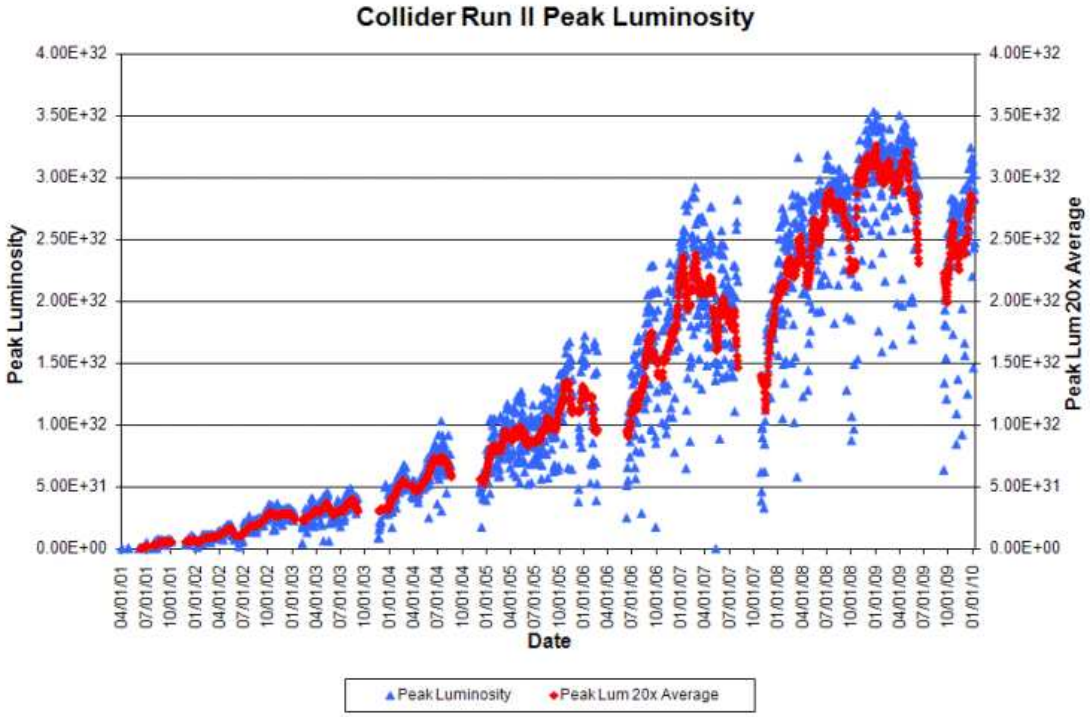


Figure 2.4: CDF initial instantaneous luminosity as function of the time.The performance of the accelerator has steadily increased over time and it's foreseen to be able to beat the 3.5×10^{32} cm^{-2}s^{-1} record before the end of operations in 2011/2012.

the initial instantaneous luminosity and, more important, to the increase of the rate of integrated luminosity delivered the experiments.

Parameter	Value
Particles collided	$p\bar{p}$
Maximum beam energy	0.980 TeV
Time between collisions	0.396 µs
Crossing angle	0 µrad
Energy spread	$0.14{\times}10^{-3}$
Bunch length	57 cm
Beam radius	39 µm for p, 31 µm for $\bar{p}$
Filling time	30 min
Injection energy	0.15 TeV
Particles per bunch	30×10^{10} for p; 9.7×10^{10} for $\bar{p}$
Bunches per ring per species	36
Average beam current	82 µA for p, 27 µA for $\bar{p}$
Circumference	6.12 km
$\bar{p}$ source accumulation rate	25×10^{9}/ hr
Max number of $\bar{p}$ in accumulation ring	2.4×10^{12}

Table 2.1: Accelerator parameters for Run II configuration.

Figure 2.5 shows the total Tevatron luminosity delivered compared to the total luminosity recorded by the experiments as a function of the time. At CDF an average efficiency of $\sim 85\%$ is reached in collecting the delivered luminosity. About 5% of inefficiency arises from trigger dead-time. In fact in order to maximize the physics of the experiment the trigger is run at its limit, where some dead-time is unavoidable. Another 5% comes from beam conditions where losses are too high and do not allow to operate the detector properly. The last 5% is either from small detector problems or operational decision to dedicate part of a store to detector studies.

2.3 The CDF experiment

The Collider Detector at Fermilab (CDF) [14, 15] experiment improved its perforamences during its life, in order to increase the number of collected events. In fact the number of events for a process, with cross section σ, we expect to detect is

$$N_{events} = \sigma\epsilon \int \mathcal{L} \mathrm{d}\gamma$$

where ϵ is the detector efficiency for that process, $\mathcal{L}$ is the instantaneous luminosity and the integral is over the period of data taking. To increase σ we have to

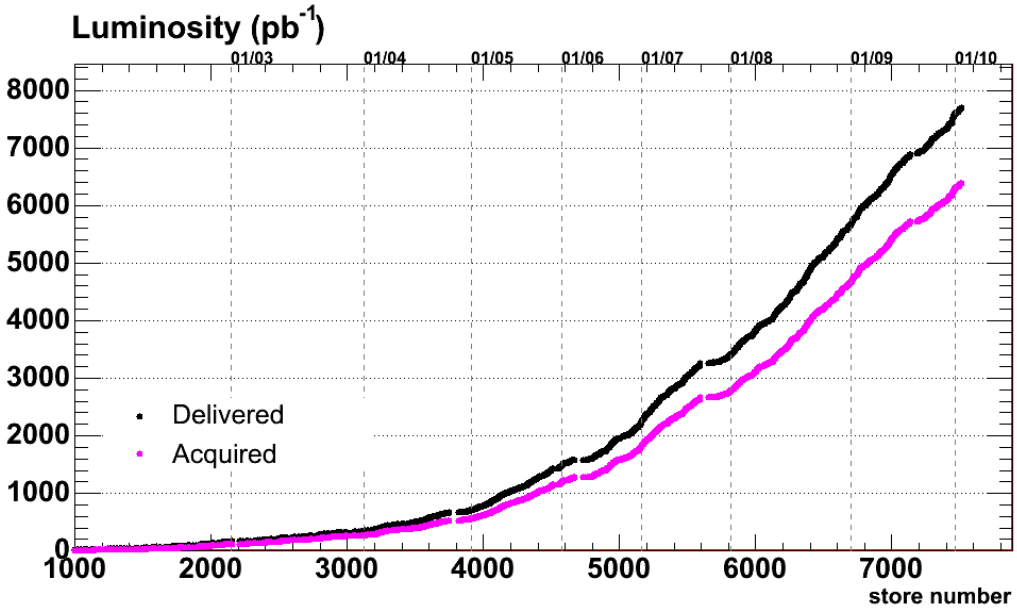


Figure 2.5: Total integrated luminosity delivered by the Tevatron collider as a function of the time. The acquired integrated luminosity by the CDF detector is also shown.

increase the center-of-mass energy, while, as we have seen in the previous section, to increase luminosity we have to act on bunch frequency and antiproton storage and recycling. Increasing ϵ means improving the detector acceptance; from *RunI* to *RunII* there have been several improvements on CDF to achieve this goal as we will point out describing the detector.

2.3.1 The CDF detector

The CDF is a general-purpose detector designed to detect particles produced from the Tevatron $p\bar{p}$ collisions. It is located at one of the two interaction points along the Tevatron collider ring as shown in Figure 2.2.

As illustrated in Figure 2.6, the detector has a cylindrical layout centered on the accelerator beamline and it has the typical structure of a collider experiment: many sensors disposed in an "onion"-like structure from inside to outside. The inner detector is the tracker made by internal barrels equipped with silicon double-face microstrip sensors (it is subdivided in three subdetectors starting from interaction point: L00, SVX II and ISL) followed by a multiwire drift chamber (COT). Tracking detectors are installed in the region directly around the interaction point to reconstruct charged-particle trajectories inside a 1.4 T uni-

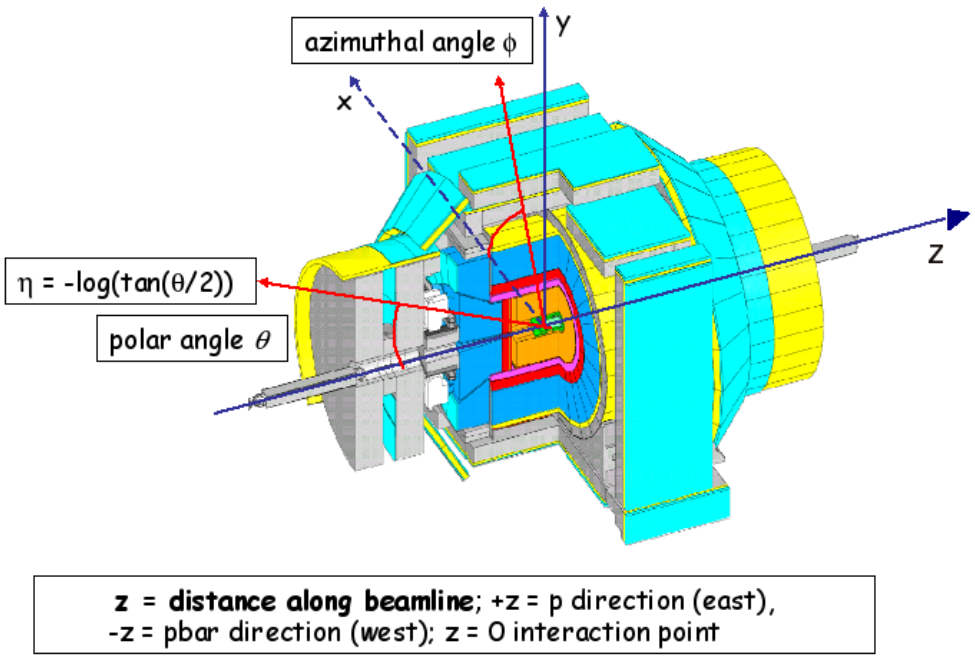


Figure 2.6: Isometric view of the CDF II Detector and its coordinate system.

form magnetic field (along the proton beam direction). The field is produced by a 5 m long superconducting solenoid located at the outer radius of the tracking region (1.5 m). Calorimeter modules (preshower, electromagnetic calorimeter and hadronic calorimeter) are arranged in a projective tower geometry around the outside of the solenoid to provide energy measurements for both charged and neutral particles. The outermost part of the detector consists of a series of drift chambers used to detect muons, which are minimum-ionizing particles that typically pass through the calorimeter with minimum interactions.

2.3.2 CDF Coordinate system

CDF uses a Cartesian coordinate system centered in the nominal point of interaction, with the z axis coincident with the beamline and oriented parallel to the motion of the proton beam. The x axis is in the horizontal plane of the accelerator ring, pointing radially outward, while the y axis points vertically up (see Figure 2.6).

For the simmetry of the detector, it is often convenient to work with cylindrical (z, r, ϕ) coordinates. The azimuthal angle ϕ is measured in the $x-y$ plane starting

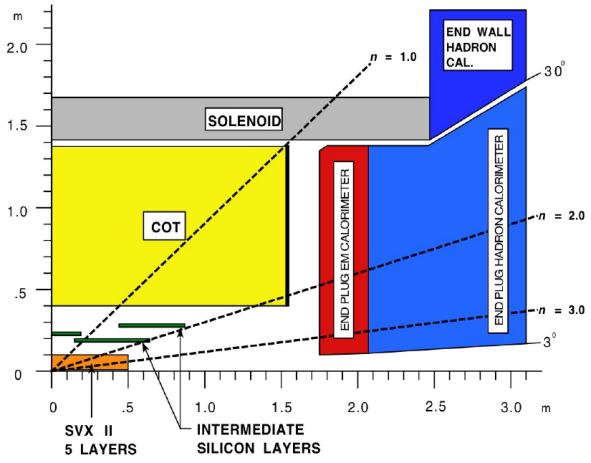


Figure 2.7: The CDF II detector subsystems projected on the z/y plane.

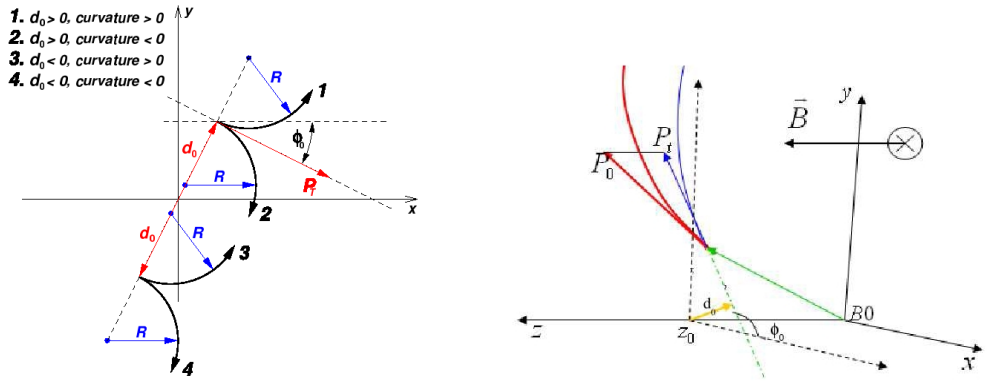
from the x axis, and it is defined positive in the anti-clockwise direction seen from the z orientation. The coordinate r defines the transverse distance from the z axis. The polar angle θ is measured from the positive direction of the z axis. Usually a function of θ is used, called *pseudorapidity* and defined as:

$$\eta = -\log \tan \frac{\theta}{2} \tag{2.9}$$

Particles perpendicular to the beam line have $\eta = 0$. The pseudorapidity is usually preferred to θ at hadron colliders, where events are boosted along the beamline, since it transforms linearly under Lorentz boosts, i.e. η intervals are invariant with respect to boosts. For these reasons, the detector components are chosen to be as uniformly segmented as possible along η and ϕ coordinates.

2.3.3 Tracking System

The inner part of the CDF II is devoted to tracking systems. Figure 2.7 shows one quadrant of the longitudinal section of the CDF tracking system. Charged particles passing through matter cause ionization typically localized near the trajectory followed by the particle through the medium. Detecting ionization products gives geometrical information that can be used to reconstruct the particle's path in the detector by means of the *tracking* procedure. The tracking system's volume is permeated by an uniform magnetic field of magnitude B=1.4 T, oriented along the z-axis. This feature constrains charged particles to an helicoidal trajectory by means of the Lorentz force, whose radius, measured in the transverse plane $(x - y)$ is directly related to the particles transverse momentum, P_T.



(a) (b)

Figure 2.8: (a) Illustration of helix track parametrization. (b) CDF track parameters and coordinate system.

Particle trajectories can be completely described by five parameters [16]:

- z_0 : the z coordinate of the closest point to the z axis;
- d_0 : the impact parameter defined as the distance between the point of closest approach to z axis and the z axis;
- ϕ_0 : the ϕ direction of the transverse momentum of the particle (tangential to the helix) at the point of the closest approach to the z axis;
- $\cot\theta$: the helix pitch, defined as the ratio of the helix step to its parameter;
- C : the helix curvature.

Actually, the impact parameter and the curvature are signed quantities defined by :

$$C = \frac{q}{2R}, \tag{2.10}$$

$$d_0 = q(\sqrt{x_c^2 + y_c^2} - R), \tag{2.11}$$

where q is the charge of the particle, $(x_c^2+y_c^2)$ is the center of the helix as projected onto the x-y plane and R is its radius. A graphical view of these variables together with the ϕ_0 is shown in figure 2.8(a). The figure 2.8(b) show the five parameters used to describe the helix trajectory traveled a charged particle in the magnetic field B.

From helix parameters one can easily derive particle transverse and longitudinal momenta as:

$$P_T = \frac{cB}{2|C|}, \tag{2.12}$$

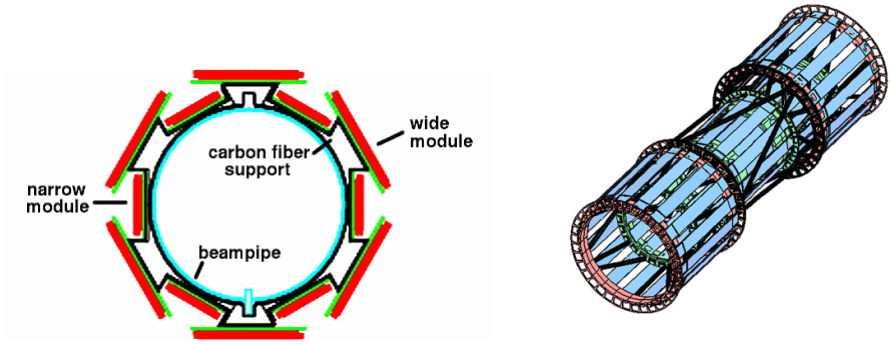
$$P_z = P_T \cot\theta. \tag{2.13}$$

Inside the COT there are the silicon detectors: SVX II, ISL and L00. They are complementary to the COT and they extend the η coverage. The silicon detectors provide 3-dimensional track reconstruction. The achieved longitudinal impact parameter resolution is 70 µm. CDF inner tracking system consists of three silicon detectors (SVX II, ISL and L00), responsible for high precision measurements, and a drift chamber (COT), devoted to add further information for track reconstruction. They provide an excellent transverse impact parameter resolution of 27 µm.The silicon detectors provide 3-dimensional track reconstruction. The achieved longitudinal impact parameter resolution is 70 µm. Going in more detail and starting from the interaction point we find:

- **Layer 00 (L00)**
  A single sided silicon microstrip detector located immediately outside the beam pipe, at a radius of approximately 1.6 cm and covering $|\eta| \leq 4$ (Fig. 2.9(a)) [17]. It provides best impact parameter resolution and better b quark tagging efficiency compared to the other tracking devices .

- **Silicon Vertex Detector (SVX II)**
  A double sided silicon micro-strip detector, located outside L00, extending from $r = 2.1$ cm to $r = 17.3$ cm and covering $|\eta| \leq 2$. Figure 2.10(a) shows a cross section of the SVX II. The SVX II is organized into 12 azimuthal wedges. For each wedge there are 5 detector layers each providing one axial measurement on one face, while in the other side are aligned with a small (1.2°) angle stereo (layers 2 and 4) and with a 90-degree stereo (layers 0, 1 and 3). It provides high precision tracking and secondary vertex detection. Figure 2.10(b) shows an isometric view of the SVX II. The SVX II is made of three mechanical barrels. Each mechanical barrel is made of two electrical barrels. In fact, within a mechanical barrel each detector element is built of two silicon sensors with independent readout paths. The two sensors are aligned longitudinally to achieve a total length of 29 cm, which is the length of each mechanical barrel. Hence, for each wedge and for each layer there are a total of 6 sensors belonging to 3 different mechanical barrels.

- **Intermediate Silicon Layer (ISL)**
  A double sided silicon micro-strip detector (Fig. 2.9(b)), with axial strips on one side and small angle stereo strips on the other side. It consists of



(a) (b)

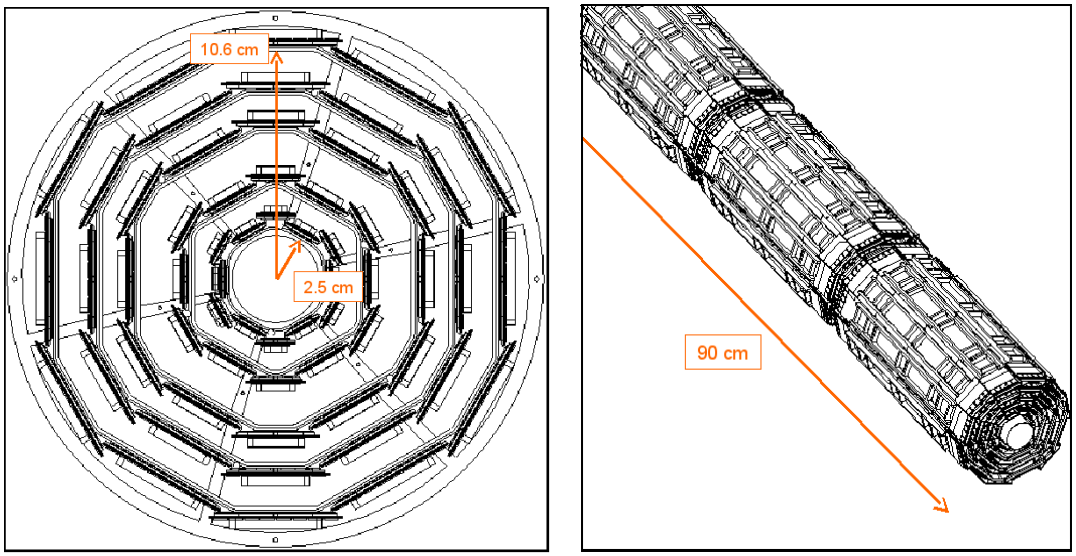
Figure 2.9: Layer 00 and ISL: (a) Transverse view of Layer 00, the innermost silicon layer, (b) Perspective view of ISL.

three layers, positioned at different radii: central layer is at $r = 22$ cm while forward and backward layers are respectively at $r = 20$ cm and $r = 28$ cm. They also have different η coverage: $|\eta| \leq 1$ for central layer and $1 \leq |\eta| \leq 2$ for the others. The L00 and ISL detectors are not used by the SVT.

- **Central Outer Tracker (COT)**
  An open cell drift chamber with argon-ethane gas in a 50/50 mixture. It's located outside SVX from $r = 40$ cm to $r = 137$ cm, covering $|\eta| \leq 1$, thus providing tracking in the central regions of the detector with an excellent curvature resolution of 0.15 P_T(GeV)%. Its 96 detector layers are grouped into 8 super-layers, each containing twelve layers of sense wires. Four of the super-layers have wires parallel to the beam (*axial* super-layers), while the others have wires at a small (2°) stereo angle (*stereo* super-layers).

2.3.4 Multi-level trigger

At hadron collider experiments the collision rate is much higher than the rate at which data can be stored on tape. At CDF the predicted inelastic cross section for $p\bar{p}$ scattering is 60.7 ± 2.4mb, which, considering an instantaneous luminosity of order $10^{32}\text{cm}^{-2}\text{s}^{-1}$, results in a collision rate of about 6MHz, while the tape writing speed is only of $\sim$ 100 events per second. The role of the trigger is to efficiently select the most interesting physics events. Events selected by the trigger system are saved permanently on a mass storage and subsequently fully reconstructed offline.



(a) (b)

Figure 2.10: CDF II Silicon Vertex Detector: (a) $r-\phi$ view of SVX II. Each SVX barrel is made by five layers and on the r-ϕ plane is subdivided in 12 slices wide 30° ϕ called wedges. (b) Perspective view of SVX II. SVX is made by three separate barrels (called mechanical barrels) of five layers detector in the z direction. Each barrel is made by two bonded barrels (called electrical barrels).

Each subdetector of the Tracking Detector has its own data channels, and it's own time response and readout bandwidth. Not all subdetectors can be read at every collision. At CDF, the silicon tracker can be read at a maximum rate of 30 kHz without damaging the sensors and causing dead-time to the experiment.
Furthermore the algorithms to extract useful information from sampled data have a wide range of timing and complexity: finding global calorimetric parameters (sum of all transverse energies, missing transverse energy, for example) is very fast, finding jets (clusters of energy in calorimeter) is slower like finding tracks with offline quality. Also the trigger decision algorithm, that apply cuts on the parameters reconstructed by the various trigger processors, might be of a wide range of complexity and timing.
In this context it is not convenient to use all processors at one time on the same event, because it would be always necessary to wait for the slowest, and apply an efficient but complex and slow decision. This strategy would lead to a certain amount of time where collision would happen but the system would be busy and the data would be lost.
It is much more convenient to group processors based on their bandwidth and latency, then organize the trigger in a pipelined multi level scheme:

- at the first level the fastest algorithms are executed and a first decision is taken reducing the input rate that has to be analyzed by the slower processors at level 2.

- At the second level the second fastest algorithms are executed on data collected by the first level and a second decision is taken and so on.

This scheme allows to employ complex algorithms that otherwise would generate dead-time at later levels characterized by lower input rates. The amount of data that needs to be buffered before the final decision is also minimized.
This strategy suggests to put slower processors at high levels of the trigger, but for the sake of collecting high purity data it's mandatory to be able to do sophisticate selections from the first levels of trigger. The solution is to employ powerful dedicated processors in order to make complex and precise algorithms fast enough to be put in the first levels of trigger. This is the strategy that CDF has followed for triggers based on reconstructed tracks, pushing tracking processors at the first two levels of trigger and allowing collection of high quality data.

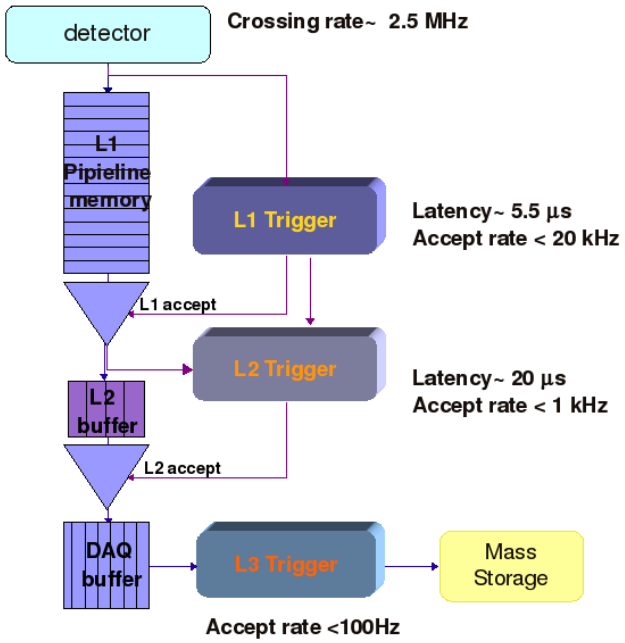


Figure 2.11: CDF trigger system.

Trigger system

The CDF trigger system has a three-level architecture providing a rate reduction sufficient to allow more sophisticated event processing one level after another with minimal dead-time (see Figure 2.11).

- **Level 1**
  The level 1 (L1) is a synchronous pipeline system where up to 42 subsequent events can be stored in 5.5 µs of latency while the hardware is making a decision (in about 4 µs), so that it doesn't cause any dead-time. If no acceptance decision is made data lost[4]. The L1 decision is generated by:

  – XFT (extremely fast tracker), which reconstructs approximate tracks in the transverse plane by exploiting information from superlayers[5]. These tracks can be propagated to the calorimeter and to the muon chambers to contribute to higher trigger levels.

  – the calorimeter trigger, which indicates large energy releases in the

[4]In order to be as fast as needed by the no-dead-time condition L1 employs only hardware.
[5]It searches the 4 axial superlaers for track segments, then the Linker Board tries to link together at least three of them to form a track.

electromagnetic or hadronic cells (these can be seed for electrons or jets identification).

– the muon trigger, which matches XFT tracks to stubs in the muon chambers.

The L1 rejection factor is about 150 and the typical output rate is reduced from 2.5 MHz to about 20 kHz. Events passing the L1 trigger requirements are then moved to one of four on-board Level 2 (L2) buffers.

- **Level 2**
  The second level is asynchronous, made by dedicated processors and a final decision commercial CPU. Events accepted L1 are sent to 4 asynchronous buffers at level 2 (L2). Buffers are used to store events until a decision is made. Because of the limited size of the buffers dead-time may occur. Each separate L2 buffer is connected to a two-step pipeline, with an average latency time of 20 µs: in step one, single detector signals are analyzed, while in step two the combination of the outcome of step one are merged and trigger decisions are made. It's mandatory for all L2 processors to have not only the processing time with compatible average, but also short tails to avoid dead-time. L2 purposes are to:

  – add the energy deposited in the towers in small regions around L1 seeds, as an approximate measure of an electron or jet energy,

  – refine L1 calorimetric decision,

  – reconstruct a full 3D COT track and associate it to an outer muon stub in order to improve muon signature,

  – indicate tracks with a large impact parameter by means of Silicon Vertex Trigger (SVT) on which to trigger on secondary vertexes from decay of long-lived beauty hadrons.

  The data acquisition system allows a L2 trigger accept rate of $\sim$ 1kHz and a $L1 + L2$ rejection factor of about 2500. Events satisfying both L1 and L2 requirements are transferred to the Level 3 (L3) trigger processor farm.

- **Level 3**
  The third level is a CPU farm. L3 addresses event objects delivered by L2 to the Event Builder (EVB), which reconstructs the entire event with the same accuracy as in the offline reconstruction algorithms. The final decision to accept an event is made on the basis of a list of observables indicating candidate events of physical interest (top production events, W/Z events, Drell-Yan events, etc).The third level reduces the rate of events to 100 Hz for permanent storage on tapes.

Chapter 3

Online Track Reconstruction Algorithm

In this chapter I will describe how information from the *Tracker*, made of silicon detectors and COT, are used to reconstruct charged particle trajectories.

3.1 Track reconstruction

Track reconstruction is based on the position of the hits[1] leaved by charged particles on detector components. Combining these hits one can reconstruct particle trajectories. Tracking algorithms reconstruct the trajectory of a charged particle.

From the the trajectory it is possible to extract very useful informations for event selection. If the tracker is within a magnetic field, for example a solenoidal field oriented along the beam axis, it is possible to reconstruct the transverse momentum of the particle from the curvature of its trajectory. Reconstructing with great precision the track allows the primary vertex identification and also to find secondary decay vertices and thus identify events with particles with long life (i.e. b-quarks, taus), extremely effective to select physics of interest.

Charged particle tracking is a very rich source of information, in fact it's a major technique in offline analysis where most sophisticated algorithms were developed.

To be able to develop an online tracking algorithm with timing performance suitable for trigger decision, but quality similar to offline tracking, it's a very

[1] signals coming from fired detector channels are called hits.

challenging task. To analyze the problem we must start from how the information is recorded in the tracker sub-detector.
The charged particles cross different parts of the tracking system. Different information is obtained from each of the sub-detectors:

- L00 measures the ϕ coordinate of a hit with its microstrips parallel to z axis.
- SVX II is composed of 5 layers equipped with double face microstrip detectors. Each layer has strips parallel to z axis on one side in order to measure ϕ coordinate. Layers 1, 3 and 5 have on the other side strips perpendicular to z axis thus providing a measure of z position. Layers 2 and 4 have strips with a small angle with respect to ϕ strips. These strips are used to combine the r/ϕ measurement of a track with its z measurement.
- ISL provides r/ϕ measurements of a hit, because its layers have axial strips on one side and small stereo angle on the other.
- COT provides r/ϕ measurement too, because half of its layers have wires parallel to z axis and the others have wires at a small stereo angle.

3.2 Tracking Algorithm

When a charged particle crosses a tracking detector, for example the five-layer SVX II detector, its passage is observed as five strip hits, one in each layer. The tracking algorithm must reconstruct the particle trajectory from the position of this five strips. This is not the only task of the tracking algorithm: if, for example, two particles cross the detector the signal will be ten strip hits, two in each layer, but there's no additional information that suggests which hit was produced by which particle. The tracking algorithm must also solve the combinatorial problem associating hits into candidate tracks, selecting the ones that are real tracks and rejecting the fakes.

The problem of associating hits into track candidates is as important as reconstructing trajectory parameters from the candidate itself. This is a very time consuming task for the tracking algorithm, especially in modern experiments where hundreds if not thousands of charged particles are present at the same time, together with the remnant signal of previous collisions (pile-up) and a certain amount of random noise.

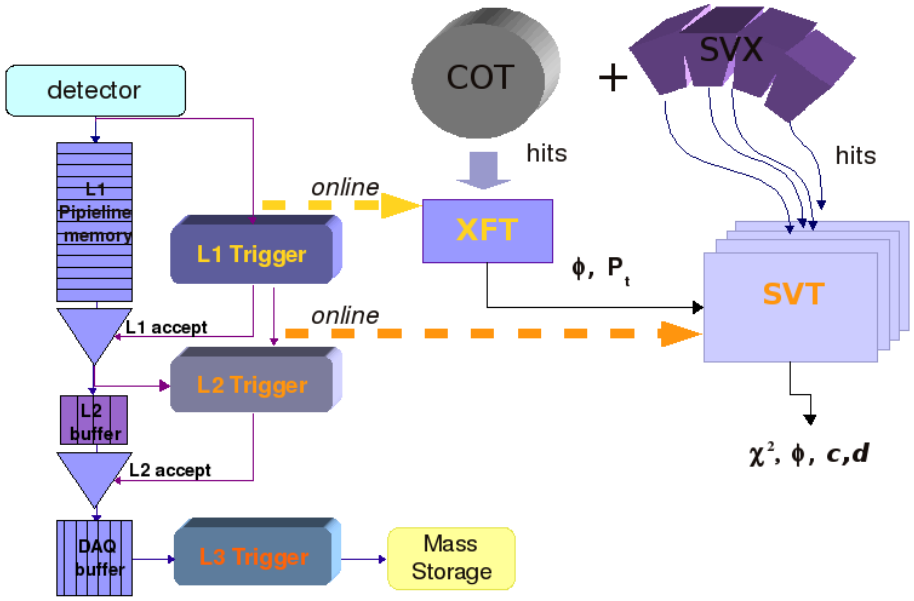


Figure 3.1: XFT and SVT in the Trigger chain.

It's worth to notice that at CDF a tracking processor is present starting from the first level of trigger (figure 3.1). In fact at level one (L1) there is a hardwired algorithm named **X**tremely **F**ast **T**racker (XFT) for the reconstruction of transverse trajectories segments in the COT chamber and at level two (L2) there is **S**ilicon **V**ertex **T**rigger (SVT) for offline quality reconstruction using SVX II and XFT tracks. We will describe SVT in details in the section 3.2.2.

3.2.1 XFT

XFT[1], [18] has been developed to reconstruct tracks in the plane of the drift chamber transverse to the beam axis in time for L1 decision using hit data from the 4 axial superlayers of the chamber. Track identification is performed searching and combining track segments in the 4 axial superlayers of the drift chamber. XFT measures transverse momentum P_T and azimuthal angle ϕ of all the tracks with P_T >1.5 GeV/c with an efficiency greather than 96% and a resolution $\sigma_{P_T}/{P_T}^2 \sim 2\%$ (GeV^{-1}) and $\sigma_\phi \sim 6$ mrad.
Track segments are also found in the outer stereo layers of the chamber. This feature allows to reject at L1 fake axial tracks by requiring the association with stereo segments. Stereo segments can also be sent to L2 and matched to the

axial tracks for 3D track reconstruction which provides a good resolution on $cot\theta$ ($\sigma_{cot\theta}$= 0.11) and z_0 (σ_{z_0} = 11 cm).

3.2.2 SVT

The SVT[19, 20] (Silicon Vertex Trigger) is a L2 trigger processor dedicated to the reconstruction of charged particle trajectories (the parameters curvature c, impact parameter d and azimuthal angle ϕ, as defined in the section 2.3.3) in the plane transverse to the beam line. The algorithm exploits the information coming from five layers of the SVX II detector (using only one face of the double face layers) plus the parameters c and ϕ reconstructed by the XFT processor (see Fig.3.1).
The algorithm is subdivided in two distinct phases:

- Pattern recognition: finds low resolution tracks, called roads[2], by associating the signals coming from the detector (hits);
- Linear Fit: does the combinatorics and finds high resolution track parameters by fitting all the combinations inside each low resolution track candidate.

3.2.3 Pattern recognition

The first step consists on finding the roads associated to the hits in the detector tracker. Low resolution hits are the result of the matching. This step is necessary in order to contain the memory where patterns are stored, called pattern bank, within an acceptable size. For each event, a number of particle tracks traverse the detector. Each track crosses one bin per layer[3], generating hits. Therefore, each event is associated to specific strings of hits and misses: the fired bins are then grouped in super-bins for the spatial resolution reduction. All tracks of physical interest correspond to bit patterns that are explicitly enumerated and stored in an appropriated data bank. The size and the number of stored patterns depend on several factors such as:

- the luminosity region, i.e. the region around the interaction point we want to reconstruct tracks coming from

[2]A road is a coincidence between hits on at least four out of five of the silicon layers and a XFT track.A road is precalculated and stored in large memories.
[3]Every detector layer is segmented into many bins.

- the coverage bank, i.e. the geometrical acceptance of all patterns stored in the bank[4]
- the coverage as function of the P_T and d_0 of the tracks
- the number of hit combinations to be solved to find the track once a road has be found.

The current bank is made from tracks with $p_T > 2$ GeV/c and a beam spot radius of 0.14 cm.
Given a fixed amount of storable patterns, the biggest the coverage of each pattern, the highest the overall coverage, but also the number of combinations of fits belonging to the pattern will be higher. Optimization of the bank coverage is a difficult problem, which solution is strongly dependent on the detector characteristics. It is anyhow true that, regardless the kind of optimization, the largest the bank the highest the coverage and so the efficiency of the tracking algorithm.

Pattern recognition is performed by the Associative Memory (AM)[21] and returns the address of the matching location. The AM is a massive parallel mechanism based on the search of roads among the list of SVX II hits and XFT tracks. Upon receiving a list of hits and tracks, each AM chip[5] [22] checks to see if all of the components of one of its roads are present in the list of hits and XFT tracks.
How an associative memory works can be explained with the following example: each element of the associative memory is a pattern and is like a bingo player with his own scorecard, incoming data flux is distributed to each pattern like the numbers in bingo are read out loud. At each given number each player checks if it is present in his own scorecard and when it makes bingo - when all super-bins in the pattern are hit in a given event - it announces the win. All winning players, i.e. all pattern addresses, are collected and sent to output.

From the data bank the fired roads are extracted and associated to the high resolution hits, which are then used to fit the track parameters. A conceptual sketch of this procedure is shown in fig.3.2.

Once we have a pattern bank that covers at best the volume of all possible trajectories, once the patterns present in a given event are found,the remaining tracking problem is compute the combinations inside each pattern and perform

[4]The coverage of the bank is defined as the ratio of covered trajectories with respect to all possible ones.

[5]An associative memory chip, able to receive the flux of hits coming from the detector and find all patterns for a given event, has been completely developed by the CDF collaboration for SVT.

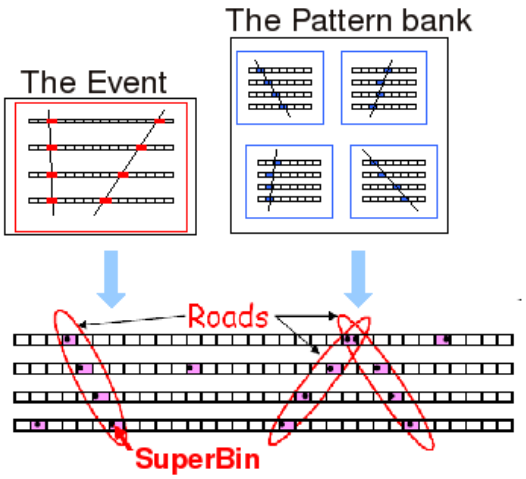


Figure 3.2: A schematic rapresentation of pattern recognition

the fit.

3.2.4 Linear fit

The association of hits into a track candidate is usually a huge combinatorial problem, very time consuming.

The reconstruction of the charged-particle trajectory consists in determining the track parameters trough an helix fit. Fitting an helix can be a tough problem but the algorithm used by SVT makes it easier, permitting to solve it within the time imposed by Level-2 trigger. It is based on the Principal Component Analysis [23] and the linearization of the problem. Linearity is guarantee by the small road size (see fig.3.3) and it allows to greatly reduce the processing time.

We start to look at this problem: how the combinations of hits coming from real charged particles differ with respect to the generic random combination?

If we look at the n-tuple of hits as a point in a n-dimensional space, where n is the number of detecting layers, and we eliminate every possible error of measurement or statistical physical effects, we'll see that the points in n-dimension that are coming from real particles belong to a well defined m-dimensional manifold where m is the number of free parameters in the trajectory equation.

This means we can write $n - m$ equations of the hit coordinates $\vec{x}$, called constraint equations:

$$f_i(\vec{x}) = 0$$

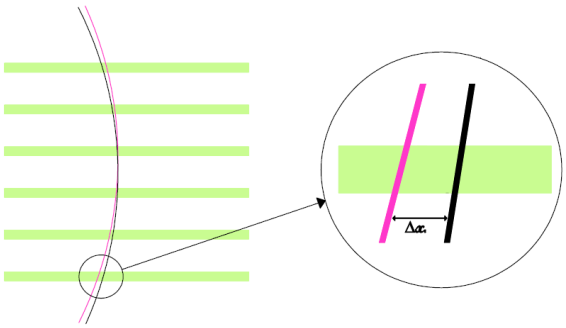


Figure 3.3: Within a road, the hits coordinates in each layer are separated by a small displacements. The track parameters and χ^2 can then be approximated with functions linear in the displacements coordinates.

In SVT we have a six dimensional space (four coordinates from SVX II corresponding on the positions of the hits on four of the five SVX II layers, and two coordinates from XFT, c and ϕ); every track can be thought as a point on $\mathbb{R}^6$ represented as $\vec{x} = (x_1,....,x_6)$.

The trajectory equation being constrained in the transverse plane has three free parameters (c, d, ϕ). Hence it is possible to write three constraint functions $f_i(\vec{x}) = 0$, i= 1, 2, 3, to reduce the degrees of freedom from the initial six to three. If these functions are known, substituting a group of six coordinates inside them will enable to determinate if it is a track.

The three constraint equations, representing a 3-dimentional surface embedded in $\mathbb{R}^6$, can be very complex but they can be locally linearized, i.e. the surface can be locally substituted with its 3-dimensional tangent plane.

$$f_i(\vec{x}) \sim v_i \cdot (\vec{x} - \vec{x_0}) = \sum_{j=1}^{6} v_{ij} x_j + q_i; \;\; i = 1, 2, 3$$

This approximation is very good within a road, where coordinates are small displacements. v_{ij} and q_i depend only on the detector geometry and can be calculated offline with simulations of the detector by using real data. In a real detector, uncertainties will make the coordinates to fluctuate statistically.

The x_i belonging to the same track are correlated trough a covariance matrix

$$\sigma_{ij} = \langle x_i x_j \rangle - \langle x_i \rangle \langle x_j \rangle$$

Computing the covariance matrix F_{ij} of f_i is a way to quantify the characteristics

of this volume. At first order is:

$$F_{ij} \simeq \frac{\partial f_i}{\partial x_k}\frac{\partial f_j}{\partial x_l}M_{kl}$$

where M_{kl} is the covariance matrix of $\vec{x}$.

With F_{ij} it's possible to build a χ^2 function:

$$\chi^2 = \sum_{i,j=1}^{3} f_i \cdot F_{ij}^{-1} \cdot f_j$$

This expression can be simplified writing new constraint equations $\widetilde{f}_i$ such as:

$$\widetilde{f}_i = \frac{S_{ij}f_j}{\sigma_i}$$

S_{ij} is found diagonalizing F_{ij}^{-1}:

$$F_{kl}^{-1} = S_{ik}\frac{\delta_{ij}}{\sigma_i}S_{il}$$

From which we can rewrite the above in a more compact form:

$$\chi^2 = \sum_{i=1}^{3} \widetilde{f}_i^{\,2}$$

that is distributed as a χ^2 with three degree of freedom and can be used to determine if a combination $\vec{x}$ of coordinates is compatible with a real track. The χ^2 equation defines a region of probability, next to the 3-dimensional surface defined by the constraint functions, where the probability for a set of coordinates to be a real track becomes greater as the representative point approaches the surface. Therefore, we can operate a cut on the χ^2 value to accept a $\vec{x}$ as a track (see fig. 3.4 as example of linearization in a 3-dimensional space with one contraint). In general it's not easy to compute the $\widetilde{f}_i$ and the charts for the fit, but we can exploit the fact that a differentiable manifold locally admit a tangent hyperplane and thus linearize the problem. We can find a series of n-dimensional hypercubes where apply the linear approximation and obtain an atlas of linear charts for all the manifold. In CDF we'll see that the geometry of the detector itself suggests how to find those regions.

This way we'll find for each one of those regions a set of constants $\vec{v}_i$, q_i, such that the constraint equations became:

$$\tilde{f}_i(\vec{x}) = \vec{v_i}\cdot\vec{x} + q_i$$

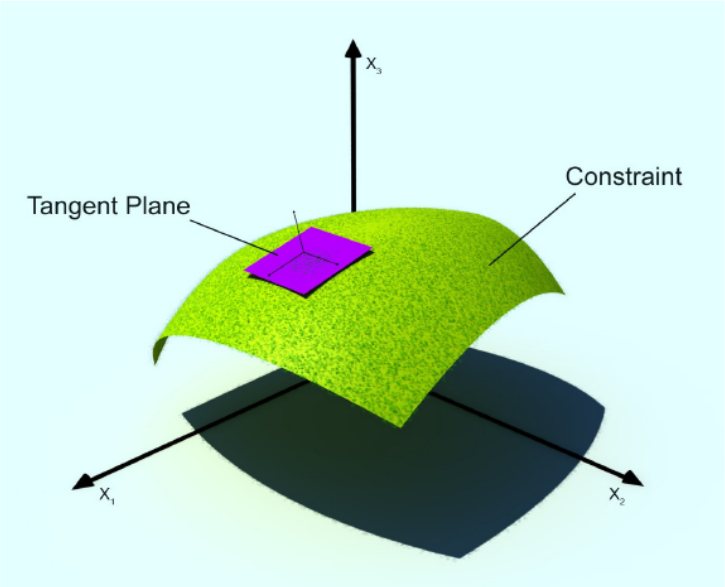


Figure 3.4: The figure shows a representation in a 3D space of the constraint surface. The surface is approximated in a small area with the tangent plane. The rotated coordinate system is also sketched. The points representing $\vec{x}$ coordinates are distributed in the probability region next to the constraint. The χ^2 is the distance of the point from the surface measured in the metric induced by the covariance matrix.

From the knowledge of the equation of motion of the charged particle, of the detector geometry and of the statistical effects on measurements, it is possible to analytically find an expression for such constants, but it's more practical to do a linear transformation of the variables that defines a new set of coordinates. On the new first axis will lie the coordinate with the maximum variance, on the second axis the coordinate with the next variance and so on. This transformation is found computing eigenvectors and eigenvalues of the covariance matrix M of the data sample $\vec{x}$: the eigenvalues λ_i quantify the variance of each axis found by the corresponding eigenvector. Three λ_i eigenvalues are negligible with respect to the others. The larger λ_i have eigenvectors in the directions lying on the constraint surface, while the small λ_i have eigenvector perpendicular to the same surface.

In the new coordinate system, redefining the $\widetilde{f}_i$, it is possible to write:

$$\chi^2 = {\chi_1}^2 + {\chi_2}^2 + {\chi_3}^2;\ \chi_i = \sum_{j=1}^{6} v_{ij}x_j + q_i\ i = 1,2,3;\ j = 1,...,6$$

which is easier to evaluate, and represents the distance of $\vec{x}$ from the constraint surface and where the χ_i are the redefined constraints that can be properly linearized. $\vec{x}$ depends on track parameters that are now to be found ($x_i = x_i(d, \phi, c)$). The dependence of parameters from the coordinate x_i can also be linearly approximated as:

$$c = \vec{v_c} \cdot (\vec{x} - \vec{x_0}); d = \vec{v_d} \cdot (\vec{x} - \vec{x_0}); \phi = \vec{v_\phi} \cdot (\vec{x} - \vec{x_0})$$

The constants $\vec{v_c}, \vec{v_d}, \vec{v_\phi}$ can be evaluated from a Monte Carlo simulation or the analysis of data from real tracks during the training of the system.
To summarize, at the end of the whole procedure[6] of linear track fitting, each generic parameter p is calculated, in a very short time, with a sum and a scalar product

$$p = \vec{v_p} \cdot \vec{x} + p_0.$$

3.2.5 Design and performances

SVT proceeds through three main steps: hit finding, pattern recognition and track fitting as illustrated in figure 3.5. Each of these steps is handled by one or more cards that are sketched in figure 3.6.

- SVX II hits coming from each wedge of each mechanical barrel (figure 2.10(b)) are found by a Hit Finder (the corresponding superstrip is computed for each hit) that converts the information in bit words for the pattern recognition. Superstrip are sent to the associative memory bank, through a Merger[7] board that merges this information with the XFT information. The hits bus from the Hit Finder is also sent to a smart database ordered by superstrip, called Hit Buffers, that sotres them waiting for the roads to come from the Associative Memory.

[6]Further details concerning the application of principal component analysis to track fitting can be found in [24].

[7]The Merger board has four input and two outputs. The purpose of the board is to merge the data coming from the four inputs or any subset of them into one output. The two outputs of the board are identical copies of the merged data, but with separate hold signal handling: it has the possibility to consider or ignore one or both holds on the two outputs. This board is used inside the SVT pipeline at various stages, but it's also extremely useful for planning the upgrade tests and commissioning because the two outputs provide easily a copy of the data made at any stage of the SVT pipeline to a new processor to be tested.

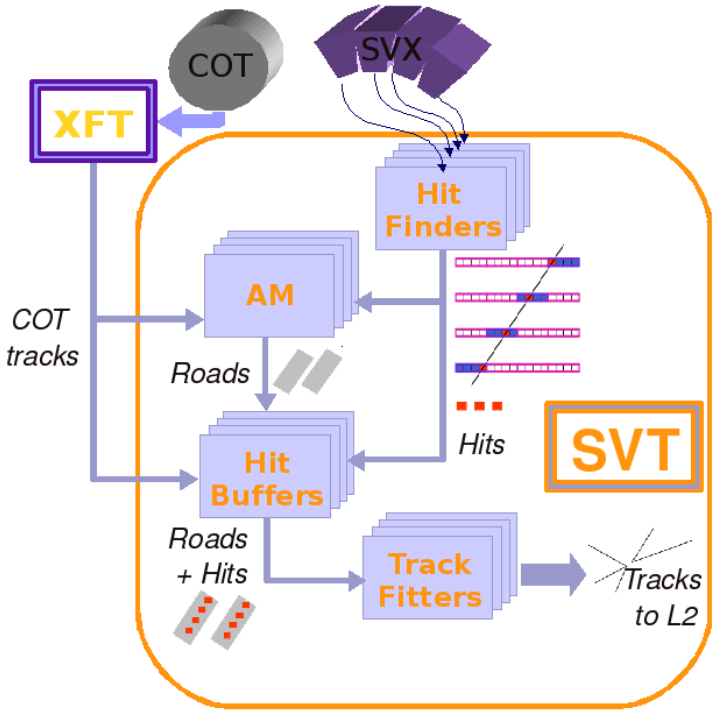


Figure 3.5: SVT scheme: input and major algorithm steps are drawn in this schematic view. Input data come from SVX II and XFT, SVX hits are clusterized, patterns are found in the associative memory and tracks are fitted with the high quality linear fit. The found tracks are sent to L2.

- Association between XFT tracks and SVX II hits is performed by the Associative Memory (AM). Patterns found in the event (roads) are received from the associative memory and patterns containing the same information (hits) are deleted ("ghost roads" removal). When it has determined that a road might contain a track, the roads hits are retrieved from the Hit Buffer and passed to the track fitter (TF).

- TF calculates the track parameters, with the full spatial resolution, of all the possible tracks corresponding to the road. The computation is done by a simple scalar product, using a linear approximation in each SVX II wedge. The TF provides precise measurement of track impact parameter d (measured with a resolution of 35 µm for 2 GeV/c tracks, which is comparable to the resolution obtained for offline reconstruction), curvature c and ϕ for all tracks with $P_T > 2$ GeV/c, as well as the χ^2 of the fit. All fits that under a

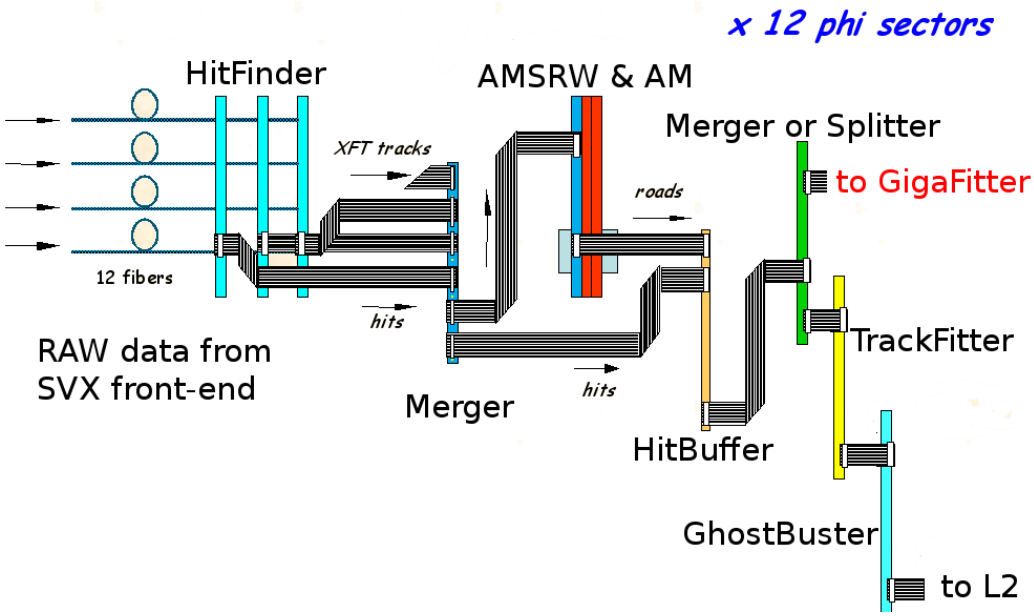


Figure 3.6: The dataflow in one SVT wedge is shown up to the final stage of the Ghost Buster were all wedges are merged in a single data cable. The position of the GigaFitter in parasitic mode is highlighted.

certain χ^2 value are collected, duplicated tracks characterized by different silicon hits associated with the same XFT track are deleted ("ghost tracks" removal) by the Ghost Buster board. Beam position is subtracted from impact parameter of each track and a second order correction is applied on ϕ parameter. Finally the tracks are sent to the output.

The overall SVT efficiency is about 80% in the geometrical acceptance region.
An important feature of SVT is that all of those steps, except the duplicate tracks removal and final corrections, can be executed independently in parallel on subregions of the detector. Since the SVX detector is subdivided in twelve wedges (figure 2.10(a)), 12 dedicated SVT pipelined systems process in parallel data for each wedge up to the last steps.
This makes SVT a highly segmented system. This configuration helps with the maintenance and the upgrade[25] of the system. I worked for the last upgrade of SVT, the Gigafitter upgrade,which I will describe in details in this thesis from the next chapter, a single board that replaces all of twelve TF++ boards enhancing the SVT capabilities. First, in sec. 3.2.6, one of the last upgrade of SVT, the Track Fitter upgrade [3] is described.

3.2.6 TF++

The running Track Fitter (TF++) is one of the last upgraded parts in SVT. It is implemented in Pulsar boards, every board fitting tracks from a SVX II ϕ sector or wedge. The TF was upgraded because the old implementation was not sufficient for the upgraded system; in particular it was too slow to process large numbers of tracks and couldn't handle more than 128k patterns. The TF++ is compatible with 512k patterns and has gained a factor 2-3 in speed. The system receives road packets from the HB, processes track parameters for multiple combinations of hits in the packet, and outputs this information to a Merger and then to the GB. The TF++ uses the three FPGA[8] chips mounted on the Pulsar board (see sec 4.2 for the Pulsar). An FPGA is in charge of all I/O. Upon receiving road-hit packets, it forms all combinations of hits in a road and creates input words for the other FPGAs, which will do all the fitting. Each one of the two chips is connected with two mezzanine cards mounted on the board, where the constants needed for the fitting algorithm are stored in RAMs.

TF++ limits

To introduce the improvements that the Gigafitter project, presented in this thesis, brings in SVT, I will now show which are the limits of the TF++.

- One of the difficulties encountered in implementing the TF++ in Pulsar FPGAs is the width of the multiplications that can be carried out in these chips. The hits information coming from the HB is 15 bits wide. To fit track parameters 15 bits multiplier are then needed, but Altera chips on Pulsar do not have dedicated multipliers, so only 8x8 bits multipliers could be implemented with normal logic. This fact makes the fitting algorithm a little more complex. To reduce the width of the multiplications the hit information is re-definited as the sum of the position x_0 of a superstrip (a low resolution bin) vertex (7 bits) and the distance d of the hit on the superstrip from the vertex (8bits) (x = x_0+ d). In this way the scalar product performed by the TF++ has to be split into two terms:

[8] A Field Progammable Gate Array(FPGA) is an integrated circuit designed to be configured in order to update its funcitionality by the customer or designer after manufacturing. FPGAs contain programmable logic components called "logic blocks", and a hierarchy of reconfigurable interconnects that allow the blocks to be "wired together". FPGAs especially find applications in any area or algorithm that can make use of the massive parallelism offered by their architecture.

1. The products $x_{0i} \cdot c_i$ of the superstrip coordinate and the coefficient of the linearized fitting algorithm are precalculated and stored in a RAM called SSMAP.
2. The products $d_i \cdot c_i$ that fits in the 8x8 multipliers.

This procedure introduces a one by one correlation between the dimension of the AM and the TF++ memory that turns out to be very large. This feature subtracted a large amount of external memory from the coefficients memory and it is the actual limit to the AM pattern bank size and thus SVT efficiency (see sec. 3.2.7).

- Another limit of the current TF++ arises in the case of a track with hits in all 5 SVX II layers (*full of hits track*): the TF++ uses 4 out of 5 hits, discarding one hit on the basis of the layers used and the quality of the hits. In a high luminosity environment, with increasing probability of fake hits due to noise, this choice can reduce track reconstruction efficiency: if a real hit of low quality is discarded in favour of a fake hit, the corresponding track does not pass the χ^2 cut and is rejected. The Gigafitter has instead enough computing power to fit all possible combinations of 4 hits out of 5 and select only the best.

- Another limiting attribute of the current TF++ is the clock frequency of the used FPGA. The problem is not limited to a greater processing time, but also affects the reconstructed tracks quality and efficiency. Fitting the same track many times, deleting one layer in each fitting process, would eliminate inaccurate track reconstruction due to noisy hits, which are recurrent, in particular, in the high luminosity run. With L2 time requirements this strategy is not possible, in the pre-Gigafitter implementation. The introduction of Gigafitter, as I will show in the next chapter, will solve this difficulties with a new design based on the newest FPGA technology. The Gigafitter has been designed to upgrade the track fitting system of SVT in order to overcome its limits during the final period of CDF data taking and increase SVT track reconstruction efficiency.

3.2.7 Track reconstruction efficiency

SVT track reconstruction efficiency is currently around 80% for tracks having at least 4 hits in the silicon detector.

The tracking efficiency needed adjustments causing variations during time, especially at the beginning of the data taking. Data taken by CDF before June 2003 used only four silicon layers connected to the XFT segment for the pattern

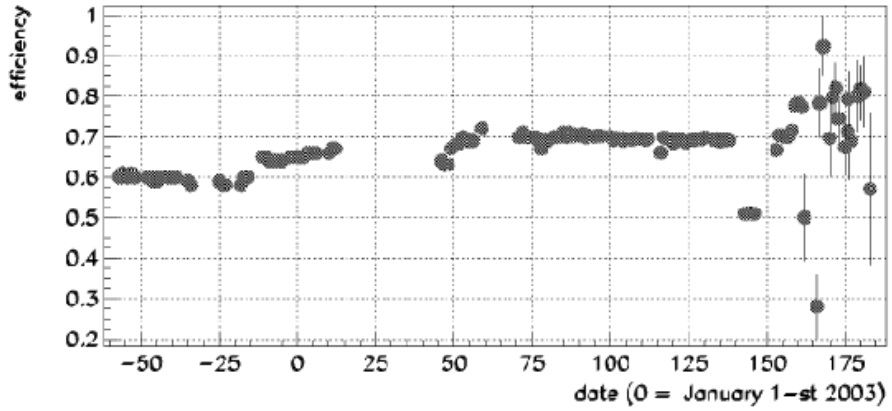


Figure 3.7: At the beginning of 2003 the SVT efficiency was limited from the status of SVX layers. The implementation of majority logic helped to overcome the non uniformity of SVX efficiency on all layers.

recognition, and requiring all of them to be fired (4/4). An important efficiency gain has been obtained implementing the use of the "majority logic" in the track match criteria. In fact SVT can require 4 fired layers among a total of 5 silicon layers (4/5). The gain is a "varying" number since it is a function of the detector status which, especially at the beginning, changed, even on short timescales. The gain increased when the detector signal/noise decreased causing higher thresholds and single channel inefficiencies. However, the gain decreased again when inside a wedge a full layer was broken, because in this case the 4/5 criteria is the same as 4/4 in that wedge.

Figure 3.7 shows the SVT track efficiency as a function of day in 2003, when the majority was implemented. The plot shows a long data taking period, where the 0 corresponds to the first of January 2003. The track efficiency has a slow increment from 60% to 70% due to the SVX detector improvement (larger number of active strips/ladders). This 70% efficiency is the product of different concurrent contributions:

1. the single hit efficiency (95%) contributes as the 81% to the track efficiency,

2. the bank efficiency (95%),

3. the χ^2 cut efficiency after the track fitting (95%),

4. a geometrical acceptance due to the SVX ladder status (95%). This is the relevant part to explain the improvement from 60% to 70%.

The 4/5 has been implemented in June 2003 and is shown in the plot as an additional track efficiency improvement up to 80%. Statistical errors are much larger in the last period since the track efficiency is calculated using a low-statistic sample. In fact the track P_T acceptance threshold has been increased in that period from 1.5 GeV to 2.0 GeV in order to reduce the L2 processing time, that increased a lot when the majority was implemented. Moreover, for the same goal, most of the events not used for the impact parameter selection have been forbidden to transit through SVT. Only few of them are allowed just to calculate the efficiency shown in figure 3.7.

The efficiency depends on the number of roads stored in AM memory and their size as well as on the number of constant sets available for the fit and stored in the TF++ memory. As an example, the current bank doesn't account for all the possible tracks which can be left by charged particles traversing the detector: tracks with $P_T < 2$ GeV are not considered, as well as track crossing the mechanical barrels. In figure 3.8 the track reconstruction efficiency as a function of $cot\theta$ and z is shown: the efficiency loss in correspondance of mechanical barrels crossing is clearly visible.

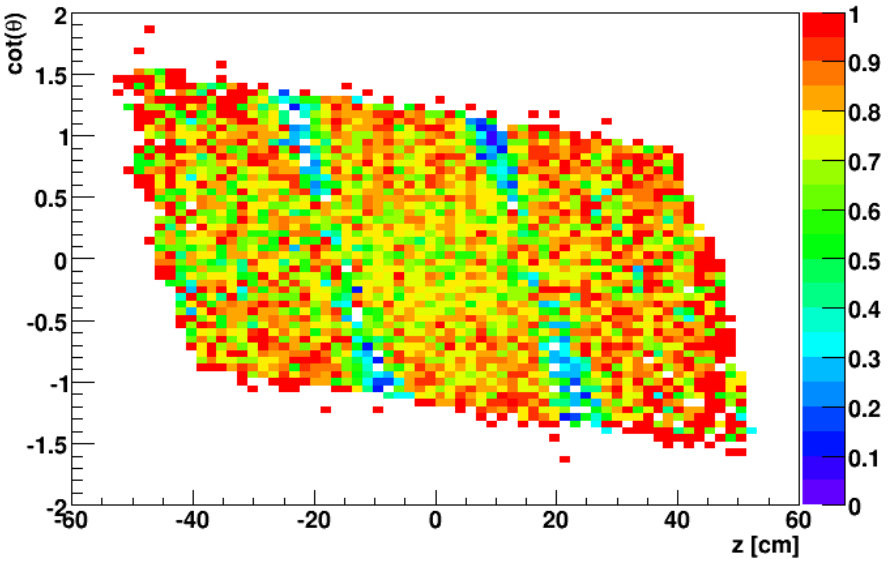


Figure 3.8: Track reconstruction efficiency as a function of $cot\theta$ and z:

The track reconstruction efficiency also depends on the road size. The raod size current value is 200 μm, the best compromise achievable with the current hardware between track reconstruction efficiency and processing time. A larger

value would provide a greater efficiency but also higher processing time, because many tracks could be associated to the same road. On the other hand, a smaller size would reduce the processing time, at the price of a lower efficiency, because the number of possible roads is limited. The maximum number of fits that can be performed by the TF++ board is limited, so increasing the road width is not feasible; as it is not possible to increase the number of the roads for a smaller road size. With the Gigafitter upgrade (see section 4.1.1 for more details) there is not he need for precalculated terms the AM memory can be fully exploited adding more patterns. Moreover, without precalculated terms, the memory on the GF board can be used to increase the number of constant sets. The Gigafitter has been designed to upgrade the track fitting system of SVT in order to overcome its limits during the final period of CDF data taking and increase SVT track reconstruction efficiency.

Chapter 4

Gigafitter

The GigaFitter (GF) is an hardware processor designed as the last upgrade of the SVT processor in order to optimize the track fitting task of the SVT algorithm. It was born from the idea of replacing a complex system made by 16 boards: 12 TF++ and 4 boards for data stream merging. It is aimed at reproducing all functionalities of current system and enhancing its capabilities, by achieving a shorter SVT processing time, a better SVT efficiency, more stable performances as higher instantaneous luminosity and a better SVT acceptance due to the capability to handle larger AM banks.
The new system must be faster than the old TF++, especially at high luminosity when events are complex and many candidate tracks must be fitted and evaluated. The GigaFitter shall allow the use of SVT at higher luminosities with larger efficiencies.
The basic idea of the GF is to implement the fitting algorithm core inside a Xilinx Virtex-5 FPGA chip[26], a single chip with a clock running up to 550 MHz, contains memories for a total of several Mbytes and hundreds of 18x25 multipliers and adders inside fast DSPs[1]. The fit of track coordinates in SVT and processors is, thanks to the linear approximation presented in the previous chapter (sec. 3.2.4), a matter of scalar products. The advantage of using DSP-like processors, packed in large number inside the FPGA, is to perform many fits in parallel. Furthermore, the high density of the packaging inside the Virtex-5 and the amount of memory and logic available permits to make inside a chip what is presently done with several FPGA chips by the TF++. The computation power of the GF,

[1]Digital signal Processing element (DSP): the add/subtract function implemented in this device is extended to function as a logic unit. This logic unit can perform a host of bitwise logical operations when the multiplier is not used. The DSP48E slice includes a pattern detector and a pattern bar detector that can be used for convergent rounding, overflow/underflow detection for saturation arithmetic, and auto-resetting counters/accumulators. The 18x25 bit multiplier inside the slice operates on twos complements operands.

thanks to the use of the most recent FPGA technology, is largely beyond the old TF++ capability.
During my thesis work I have written the basic firmware of the Gigafitter, that is the part executing the fit operation and contribute to test the GF system in order to commissioning at CDF. In this chapter I will show which is the logical structure of the Gigafitter in all its parts, I will describe the hardware and the functions that I have implemented in the firmware, also presenting some of the simulation that I have performed in order to certify the proper functioning of the system.

4.1 Design features and contribute to SVT

The Gigafitter design is based on a synchronous pipeline of simple and optimized logic modules; all modules with functions longer than one clock cycle are replicated and put in parallel to maximize the bandwidth. A set of FIFOs and buffers help to keep high clock frequencies, cross different clock domains and compensate fluctuations in the input and output data streams.
The system is able to fit and evaluate one candidate track per clock cycle on each of the 12 inputs with an internal clock of 120 MHz, about 1.4 fit/ns.
Goals of the Gigafitter are:

- full precision fits,
- larger variety of constants sets,
- better 5/5 track handling,
- a faster timing than the TF++ system.

These new features will produce better track efficiency and background/signal rejection.

4.1.1 Full precision fits

As seen in sec. 3.2.4 the computation of track parameters and χ^2 components is done with a scalar product plus a constant term:

$$p_n = c0_n + \sum c_{ni} * x_i$$

where $c0_n$ and c_{ni} are known constants and x_i are the hit positions. The terms c_{ni} and x_i are 18 bit and 15 bit wide. In the Gigafitter there are DSPs with

18x25 bits dedicated multipliers so it's possible to compute exactly p_n with the equation above, instead of a clever approximation adopted by the TF++ board, where only 8x8 bits multipliers were implemented. Each c_{ni} and x_i is decomposed as

$$c_{ni} = c_{ni}^{high8bit} * 2^{shift_{ni}} + c_{ni}^{low}$$

and

$$x_i = x_i^{ssborder} + x^{low8bit}$$

where the subscripts ^{high8bit} and ^{low8bit} indicate the 8 most significant or less significant bits respectively and $x^{ssborder}$ is the position of the road border on each layer (usually called $s$uper$s$trip).

This way the multiplications is written as:

$$c_{ni} * x_i = c_{ni} * x_i^{ssborder} + c_{ni}^{high8bit} * x_i^{low8bit} * 2^{shift_{ni}} + c_{ni}^{low} * x^{low8bit}$$

The terms $c_{ni} * x_i^{ssborder}$ and $shift_{ni}$ depends only on constants and patterns, so they can be computed offline and preloaded in a memory on the TF++. The information provided by the most significant bits is included in pre-calculated terms, one term for each AM pattern to be stored in dedicated memories of the TF++. This choice introduces a one by one correlation between the dimension of the AM and the TF++ memory that turns out to be very large. This is a disadvantage of the TF++ currently installed inside SVT: the constants used in fit's scalar products require a very large memory. This feature is the actual limit to the bank size we can use inside SVT.

The term $c_{ni}^{high8bit} * x_i^{low8bit}$ is a 8x8 bit multiplication and is calculated online.

The term $c_{ni}^{low} * x^{low8bit}$ is negligible and is not computed.

The effect of not computing the last term account for a little smear of the resolution for the TF++ with respect to the full precision computation as done by the Gigafitter and the offline code. The difference for each parameter and χ^2 is shown in figures 4.1, 4.2, 4.3 and 4.4. The χ^2 difference shown in 4.1 is proportional to the χ^2 itself because the $c_{ni}^{low} * x^{low8bit}$ term is 1-2 units for each component, then squared and summed. A small amount of track χ^2 found by the GF above the threshold were accepted by the TF++ and vice versa. Globally this effect is about 2% of the total number of tracks, but we'll see that the GF is more efficient of about the same percentage without increasing the number of fakes, so the χ^2 computed by the GF is a more accurate quality parameter.

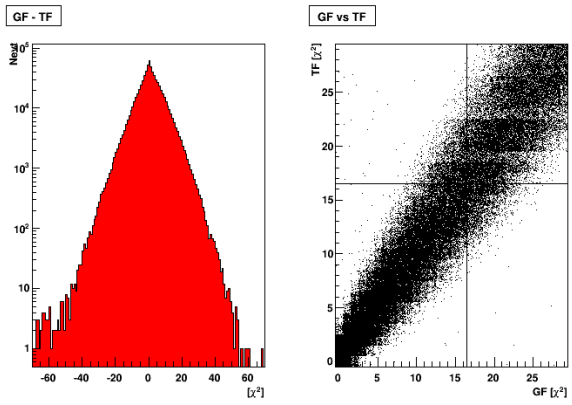


Figure 4.1: Differences in χ^2 computation between GF and TF++ due to $c_{ni}^{low} * x^{low8bit}$ term not computed by TF++. Current cut values are shown with the solid lines.

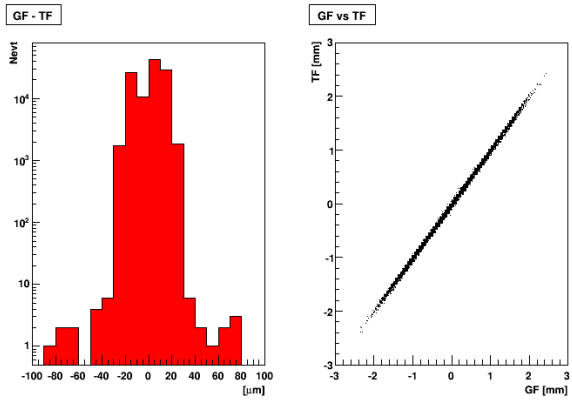


Figure 4.2: Differences in impact parameter ($d0$) computation between GF and TF++ due to $c_{ni}^{low} * x^{low8bit}$ term not computed by TF++.

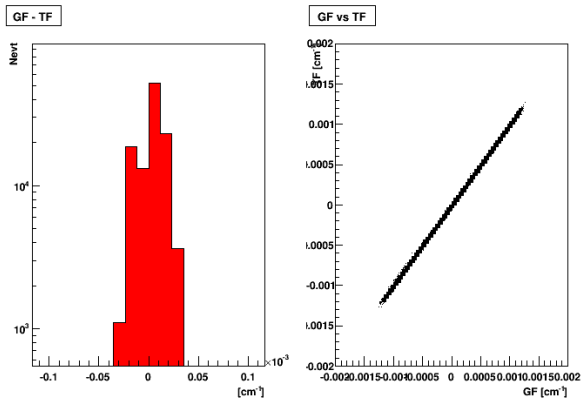


Figure 4.3: Differences in curvature (c) computation between GF and TF++ due to $c_{ni}^{low} * x^{low8bit}$ term not computed by TF++.

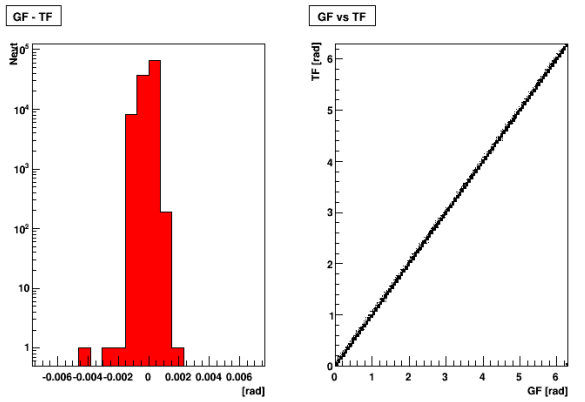


Figure 4.4: Differences in ϕ computation between GF and TF++ due to $c_{ni}^{low} * x^{low8bit}$ term not computed by TF++.

4.1.2 Several sets of constants for improved efficiency

The TF++ has different constants sets for each of his TF++ boards, one per wedge, since the fit constants extend to the whole wedge. However, inside a wedge each particular track configuration needs specific constants to be reconstructed precisely. For this reason each wedge requires various $c0_n$ and c_{ni} constants sets, each one computed for track fitting in particular conditions or regions of the detector characterized by a particular layer configuration. The large size of each TF++ constant set (the $c_{ni} * x_i^{ssborder}$ and $shift_{ni}$ constants needs to be computed for each AM pattern) puts a limit to the number of specific cases that can be handled. The TF++ board is able to store only 30 different set of constants: one for each of the 6 SVX II barrel and one for each of 4 out of 5 SVX II layer combination (6x5 constant sets). For example, this limitation results in poor quality of track crossing barrels, so poor that tracks crossing mechanical barrels are not included in the AM pattern bank and thus not reconstructed. If the patterns would be included without the addition of relative constants for their precise fitting, we would probably increase track efficiency, but also would increase the large impact parameter fake rate, resulting in a worst behavior of SVT for trigger decision.

In the GF board, instead, the 25x18 bit hard multipliers allow the use of full resolution hit position words without the storage of pre-calculated terms. The constants sets necessary to perform the scalar products are just the $c0_n$ and c_{ni} and occupy a small amount of memory. There can be a large number of different constant sets to allow the reconstruction of tracks characterized by hit configurations that up to now were discarded due to hardware limitations. This results in a potentially higher SVT efficiency and lower amount of fakes.

4.1.3 Handling of 5/5 tracks

Another clear advantage of having big DSP arrays is the capability of getting rid of noise or mismeasured hits. In fact, GF can fit many times the same track deleting one particular layer in each different fit. We then choose the layer configuration producing the best track quality.
In the current TF++ a track that has hits in all the five layers is fitted using a fixed combination of four layers and no attempt to find a better combination is performed, even if the resulting χ^2 is higher than the cut value and the track is rejected. As the Tevatron collider luminosity increases, it is very important to have the capability to evaluate the track parameters under the assumption that the probability to have a noise hit in the fitted combination is quite high. This discrimination capability allows to reduce the degradation of the SVT efficiency due to the high detector occupancy.

4.1.4 GF contribute to SVT

The Gigafitter computation power will grant a reduction of the SVT processing time also leading to a better efficiency with stable performances at higher instantaneous luminosity.

The different fits are performed in parallel, without latency increase. As the Tevatron collider instantaneous luminosity increases, it is important to have the capability to evaluate track parameters under the assumption that there could be a noisy hit in the fitted combination. This discrimination capability allows reducing the SVT efficiency and impact parameter resolution degradation due to the high detector occupancy.

Whit the Gigafitter we will be able to use a large number of different sets to allow reconstruction of tracks historically discarded because of hardware limitations. This means a better SVT efficiency.

A larger AM pattern bank will translate in three important improvements:

- Lepton coverage improvement in the forward region; the increase of the muon-electron-tau coverage at CDF is provided be L2 SVT high quality tracking in the forward region by using the SVX II only where the COT is missing. Higgs physics acceptance can be improved coupling L2 high quality tracking with high quality L2 calorimetric measurements provided by the last calorimetric trigger upgrade.

- Extension of the SVT acceptance in track P_T that will significantly improve online b-tagging capability. With this upgrade SVT will be able to reconstruct tracks down to $P_T > 1.5$ GeVGeV/c, while now only tracks with $P_T > 2$ GeV/c are used.

- Extension of the SVT acceptance in track Impact Parameter that will significantly improve the lifetime measurements. It is planned to achieve a sensibility to Impact Parameter up to 2-3 mm, while now only tracks with Impact Parameters smaller than 1.5 mm are available.

4.2 Hardware Structure

The GF has been designed to upgrade with a single board the track fitting system: a motherboard called Pulsar [27], which is an existing motherboard already used in the SVT upgrade, and three mezzanines mounted on it.

1. The Pulsar board, shown in figure 4.5, is a general purpose 9U VME interface board for HEP applications. It has been designed for the CDF trigger upgrade (L2 Global Trigger upgrade [27], L2CAL upgrade [28], and SVT

upgrade [21, 3]) but its design is general enough that it can be potentially used in many other applications, as it has been with Magic [29], for example.

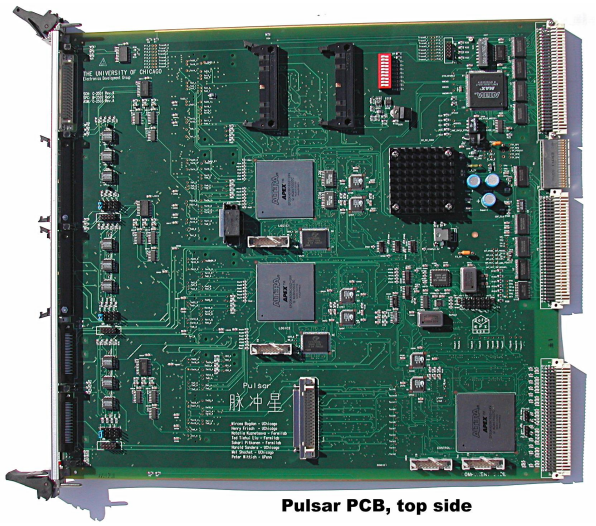


Figure 4.5: Front side of the Pulsar board. Mezzanine connectors used by the GigaFitter system are on the back side.

Pulsar is a motherboard provided with many different interfaces and three interconnected Altera APEX20K FPGA[30]: two of them, called Data I/O, handle two mezzanine connectors each, while the last, called CONTROL, handles the various input and output connectors of the motherboard. VME communications are possible directly with each FPGA.

It can hold four mezzanine cards that are connected to the two central Data I/O chips. The mezzanine card approach allows Pulsar to interface with any data path. Data I/O are connected to the third FPGA (Control) that has configurable I/O connections. All the chips on the board can be accessed via VME interface from PC, using the connectors on the backplain[2]. To program the Altera chips I used the VHDL hardware description language with the support of Leonardo Spectrum [31] for synthesis and Quartus II [32] for place and routing. This choice was driven by the fact that the VHDL language and the designing tools were used in other CDF applications based on Pulsar boards, e.g. in the old TF. The GF Pulsar board uses

[2]A backplane is a circuit board that connects several connectors in parallel to each other, so that each pin of each connector is linked to the same relative pin of all the other connectors, forming a computer bus.

two clocks: 40 MHz to communicate to GF mezzanines (clock to mezzanines is sent by the motherboard) and a 66 MHz clock for all other functions.

2. The Gigafitter Mezzanine, shown in figure 4.6, has been developed exclusively for the GF system by INFN Padova and INFN Pisa.
   On the board back side the interface with the Pulsar is mounted, on the front side there are four connectors from which the Gigafitter will receive data from each corresponding wedge. The full GF system with all 12 inputs connected is shown in figure 4.7. All communications between SVT boards are made with standard LVDS[3]cables (see section 4.3).

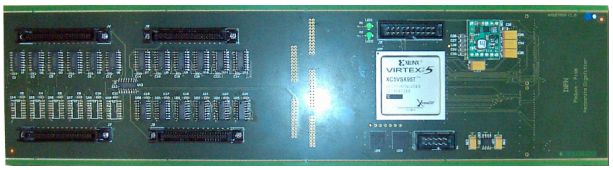


Figure 4.6: The GigaFitter mezzanine. All components on the front side.

Between the connectors there are 24 receivers and one driver that allow the translation from LVDS signals to the TTL[4] standard used inside the mezzanine and in the Pulsar. The number of receivers and drivers follows the number of bits in input (24 bits per wedge) and output (one HOLD signal per wedge). The signals are then directly sent to the Virtex-5 which is mounted on the board with the necessary power supply and two EEPROM. The two EEPROM will contain the data to program the FPGA at every startup.

The core of the mezzanine is a Xilinx Virtex-5 XC5VSX95T FPGA [26].

The Virtex-5 family provides advanced devices in the FPGA from Xilinx. Built on a 65-nm copper process technology, these chips provide a very high logic density. The family comprehends different platforms, each platform containing a different ratio of features to address the needs of a wide range of application. The device mounted on the mezzanine is the XC5VSX95T of the SXT platform (optimized for DSP and memory-intensive applications),

[3]Low Voltage Differential Signaling (LVDS) is a differential signaling system, meaning that it transmits two different voltages that are compared at the receiver. LVDS uses this difference in voltage between the two wires to encode information.

[4]Transistor transistor logic (TTL) is a class of digital circuits originally built from bipolar junction transistors and resistors. Usually the TTL acronym is also used to identify circuits that use the same voltage levels, even though they do not ude bopolar junction transistors.

which, when the mezzanine was designed, was the device containing the largest number of useful DSP slices.
The device basic constituents are:

- 640 DSP slices (DSP48E), DSP-like processors with 18x25 bit multiplier tied to a 48 bit adder, an adder/subtracter/accumulator and many other logic to perform several operations. This is the feature that drove the choice of this device and it is intensively exploited in the GF design. They perform in parallel the scalar products for the track fitting and fully exploit the computing power of the device. Thanks to them it has been possible to synthesize many parallel fitting units.
- I/O blocks that are the package interface.
- Configurable Logic Blocks (CLBs) that provide the basic logic functions, shift registers and distributed RAMs of 1.5Mb.
- Block RAMs of 8.6 Mb that are composed of 36 Kb true dual-port RAM blocks with optional dual 18 Kb mode, provided with multi-rate FIFO support logic.
- Clock Management Tile (CMT) blocks, composed of two DCM (Digital Clock Manager) blocks and one PLL (Phase-Locked-Loop) clock generator.

Figure 4.7: The GigaFitter system in the test crate of SVT. All 12 inputs are connected in parasitic mode to split HB++ outputs.

The mezzanine FPGA receives a 40 MHz clock from the motherboard (all the logic implemented on the Pulsar operate at this rate)and generates

internally three clocks using Digital Clock Manager (DCM-PLL) dedicated cells: a 40 MHz clock to communicate back to the motherboard, a 25 MHz clock to handle VME and a 120 MHz for all the other functions.

4.3 Input and output

The SVT system employs a uniform protocol for data transfer. Each word is 25 bits wide. The communication between the receiving and the transmitting boards is based on two signal:

- a $\overline{\text{Data Strobe}}$ (DS_) signal, that is an asynchronous active-low signal used as timing reference from the transmitting board (data words are sent on the cable at every positive going DS_ edge).
- An $\overline{\text{Hold}}$ (HOLD_)signal from the receiving board. Is is an active-low signal used to prevent loss of data when the destination is busy.

At every positive going DS_ edge input data are sent on the cable by the trasmitting board and pushed into a FIFO buffer. The FIFO provides an $\overline{\text{Almost Full}}$ signal that is sent back to the source on the HOLD_ line. The source responds to the HOLD_ signal by suspending the data flow. Using $\overline{\text{Almost Full}}$ instead of $\overline{\text{Full}}$ gives the source plenty of time to stop. The source is not required to wait for an acknowledge from the destination device before sending the following data word, allowing the maximum data transfer rate compatible with the cable bandwidth even when transit times are long. Signals are sent over flat cable as differential TTL. The maximum DS_ frequency is roughly 40 MHz.
On each cable there are 21 data bits, $\overline{\text{End Packet}}$ (EP_), $\overline{\text{End Event}}$ (EE_), DS_ and HOLD_ bits. Data are sent as packets of words (each word is then 25 bits wide), the EP_ bit marks the last word of each packet: the End Packet word. The EE_ bit is used to mark the end of the data stream for the current event. End Event words are one-word packets so EP_ is also 1, the data field is used for Event Tag (8 bit), Parity (1 bit, computed on all data words of the event) and Error Flags (12 bits).
The Gigafitter board receives hits and roads from the 12 Hit Buffer (HB++) boards and sends all found tracks merged in a single output to the GhostBuster board for non-linear corrections, beam position subtraction and duplicate tracks suppression.

Input data stream

The HB++ transmits for each event some hits+road packets, one for each road found by the AM, followed by an EP-bit, and reset properly the DS_ signal,

after the translation from LVDS to TTL standard performed by the mezzanine receivers.

The hits+road packet contains all of the hits measured by the SVT Hit Finder, associated to a given road found by the Associative Memory plus the road identifier (road ID) from the pattern recognition, and the track parameters measured by XFT, as described in table 4.1.
When the AM finds multiple track candidates the number of words in the packet is not fixed: the minimum is 7 words while maximum is open and depends on the road size (when there are more than one hit in a layer, the GF receives more than seven words per road). The road size commonly used in the past years gives a maximum of 25 words.

	24	23	22	21	20 19 18	17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
1	HOLD_	DS_	EE_	EP_	Layer (0..4)	SVX Hit
2	HOLD_	DS_	EE_	EP_	Layer (0..4)	SVX Hit
..					...	
x	HOLD_	DS_	EE_	EP_	Layer (0..4)	SVX Hit
x+1	HOLD_	DS_	EE_	EP_	Layer XFT	XFT 1st word
x+2	HOLD_	DS_	EE_	EP_	XFT 2nd word	
..					...	
x+2n+1	HOLD_	DS_	EE_	EP_	Layer XFT	XFT 1st word
x+2n+2	HOLD_	DS_	EE_	EP_	XFT 2nd word	
x+2n+3	HOLD_	DS_	EE_	EP_	Road ID	

Table 4.1: HitBuffer++ to Gigafitter packet format

Output data stream

Output data is composed by a packet for each track found in an event followed by the EP-bit. The track packet is always composed by 7 words and contains information about SVX II hits associated to the track on each layer, the linked XFT track, AM road, fitted track parameters and fit quality (χ^2 and GF fit status) as described in table 4.2. TF++ output protocol is used to make the GF perfectly compatible with the SVT system. However, the old protocol did not provide all the AM road ID string. So, the two missing bits (Road ID(20-19)) are placed inside two hit words by making use of two spare bits. While in the input protocol the EP bit sets the end of a roads, in the output protocol the EP bit is set with the last word of each track. The EE word is the copy of the input EE word. The track number is composed by the nine MSBs of the XFT ϕ. Since there cannot be two tracks with the same XFT information, it is used for track identification.

	24	23	22	21	20	19	18	17 16 15 14 13 12 11 10	9	8 7 6 5 4 3 2 1 0
1	HOLD_	DS_	EE_	EP_	1	0	z out-in	phi		
1	HOLD_	DS_	EE_	EP_	road		sign	c	sign	phi
1	HOLD_	DS_	EE_	EP_	phi sector				road	
1	HOLD_	DS_	EE_	EP_			x1			x0
1	HOLD_	DS_	EE_	EP_			x3			x2
1	HOLD_	DS_	EE_	EP_			χ^2			x4
1	HOLD_	DS_	EE_	EP_			GF status			track num

Table 4.2: GigaFitter to GhostBuster packet format

4.4 Pulsar internal structure

Tracks from the 12 wedges, processed by three Mezzanines, are merged in the Pulsar in the single output of the board (figure 4.8). Tracks found in each mezzanine are merged inside the three Pulsar FPGA: Data1, Data2 and Control. The final stream is sent on one SVT cable downstream to GhostBuster board.

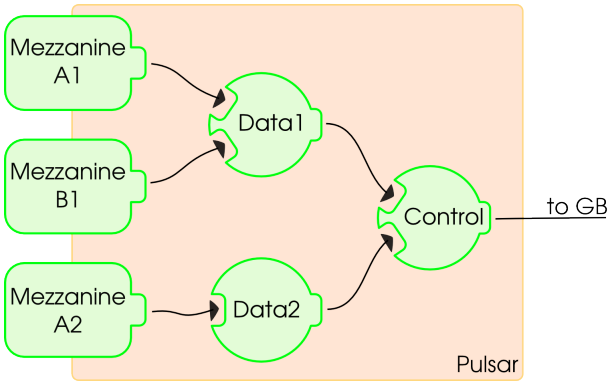


Figure 4.8: GF Pulsar Scheme.

All FPGAs, of both Pulsar and mezzanine, have VME modules that communicate on the Pulsar board; they are used to set all the needed configurations (initializing functions), to monitor the status of the board and for debugging purposes.

4.5 Mezzanine internal structure and algorithm

The GigaFitter is structured with a modular design: in each mezzanine FPGA there is one independent processor for each GF input, for a total of four inde-

pendent engines for four SVT wedges. The system is very flexible: an arbitrary number of inputs (wedges) can be activated, a feature that was extremely useful during the developing and commissioning phase.

Inside each mezzanine FPGA, as seen in the sketch in figure 4.9, there are:

- four track processing modules that work in parallel to compute tracks from one wedge each,
- a merger module, i.e. a final unit logic that merges the four data streams in a single output FIFO that communicates with the Pulsar motherboard.

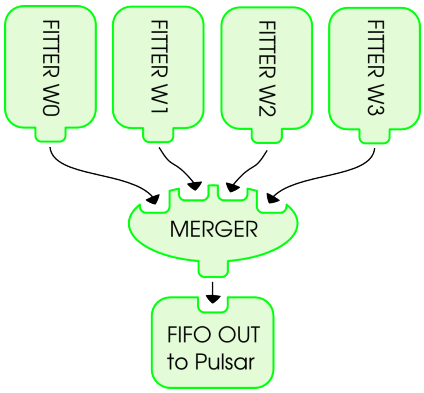


Figure 4.9: The internal structure of the GF mezzanine: four parallel fitter engines from one wedge each, and a final Merger for the four data streams merging in a single output FIFO that communicates with the Pulsar.

4.5.1 The track processing module

The track processing structure, as seen in figure 4.10, is naturally divided in six different modules: the "Combiners", the "Fit Organizer", the "Serializers", the "Fitters", the "Comparator" and the "Formatter". Large RAMs are used to store the fit constants. FiFos are used to interconnect the various stages of the pipeline and shift registers store data for the time needed to ensure the synchronization of the pipeline.
The Combiner provides combinations of input hits to be tested by the fit. The Fit Organizer coordinates the fetching of hit combinations and starting of Serializers. The DSP Fitter performs the fit. The Comparator judges the fit results and selects the best choice in the case of multiple fits. The Formatter provides the right data format to the output, exactly the same of the TF++ output.
It is a more complete design and it has improved performances with respect to the first ideas presented ([33]).

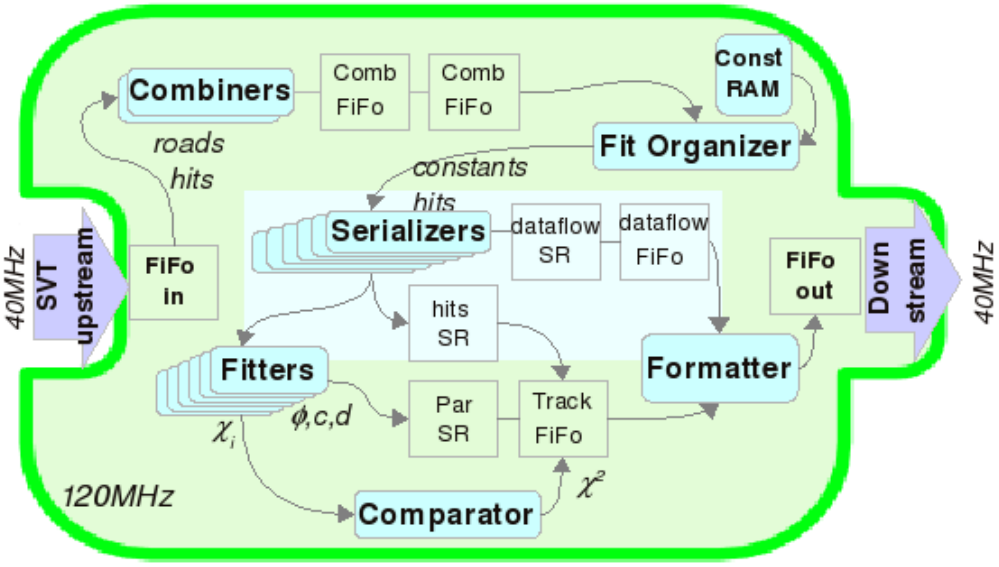


Figure 4.10: Fitter w_i: sketch of the Gigafitter fitter module for one wedge.

Combiners

The information coming from the AM boards includes the roads found in the pattern recognition and the corresponding hits. The Combiner is the part in charge of the generation of hit combination. It interfaces with the Input FiFo thanks to a control module that also recognizes the protocol words. The Combiner works in two subsequent steps:

1. it interfaces with the Input FiFo thanks to a control module to pops road packets (figure 4.11) and stores them inside small RAMs (32x19 bit each, implemented in the distributed memory of the FPGA), one for every layer. Every road can include none, one or more than one hit per layer but it must always have XFT hits. This requirement comes from the fact that, in order to have a good resolution in P_T , the track reconstruction needs the information from COT. The module reads the layer information and stores all the hits occurred on the same layer inside a shift register. Counters keep track of how many hits are recorded in each layer.

2. The road is now processed. Combiner forms the candidate tracks by generating all the combinations that can be done with the road hit list.

   It checks the presence of the XFT words, and if it receive the EP without having found them, it discards the data. When it receives the EP signal, it loops over all the layers in order to form the combinations. Using the counters information to generate RAM addresses, it fetches hits from the RAMs (one per layer) in parallel to create one hit combination at each clock cycle until all combinations are fetched.

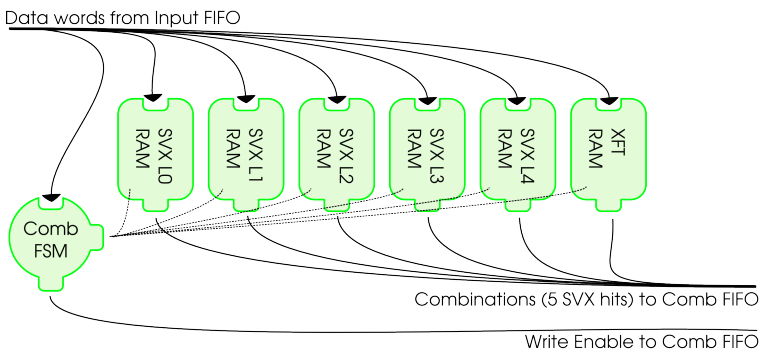


Figure 4.11: GF Combiner module: the Combiner is made by five RAMs and a finite state machine that controls both writing and combination pop.

There are two independent "Combiners", each one provided of its set of RAMs, working in parallel. While one is processing one road, the other can pop and load hits of a following road from the Input FiFo. Both Combiners work full time to provide a continuous flow of combinations that are stored in a large FiFo called Combination FiFo.

The Combiner works generating combinations of hits with always 5 SVX layers: when an SVX layer is missing in the road hits (4/5 road) it has zeros in place and the missing information is stored in an hitmap field of the combination word. All the other stages in the GF track processing module works with 4 SVX layers and the hitmap information, like the TF++. For this reason a simple finite state machine and a multiplexer connect the two Combination FiFos and converts all 4/5 combinations in a 4 SVX layer + hitmap format removing the missing layer from the combination word and all 5/5 combinations in five 4 SVX layer combinations computing the appropriate combination word.

Fitter Organizer

When combinations are ready, the Fitter Organizer pops them out of the Combiner FiFo and must retrieve the right constants and parameters for each of them. The right constants are fetched taking into account these informations:

- the involved barrels,
- the input and output barrel z,
- the hit layout (missing layers and hit map),
- the quality of the hits (Long Cluster).

Long Cluster (LC) information is flagged by the Hit Finders as low precision point, encoded in the LC map. I will now explain what the LC is.
The SVX II detector is segmented into small bins to achieve a better spatial resolution. After the digitalization of the event information, more than one bin in a layer could have been fired and these bins could be contiguous. If that is the case, only one hit is recorded and the LC flag is raised to take account of the poor spatial resolution on the measurement. Therefore, there could be a long cluster for different reasons: two particles cross the layer in adjacent bins and thus we cannot resolve them, a particle crosses the layer with a high angle of incidence traversing two bins and induced noise. Therefore the LC information is very important because it informs us about the uncertainty in the coordinate measurement, so it takes part to the constants selection and goodness of fit evaluation.
Each set of constants is a 756 bit word (7 18-bits terms in each scalar product to be multiplied by 6 scalar products). The RAM implemented using the memory

blocks embedded in the chip (BlockRAM), provides space for 256 sets (756x256 RAM).
The used layers are identified by the hit map, which is a 5 bits word. Each bit in the map corresponds to a single logical layer where c and ϕ from XFT are considered together. A look-up table is addressed with 3 bits from the z of the innermost hit in the combination, 5 bits for the LC map and 5 bits for the hit map. Because there are always 4hits/5 from SVX II for the hit map in output 3 bits are necessary. Moreover, because of the missing layer, LC map is described by 4 bits. In conclusion, the address the constants RAM would require 13 bit addresses, but the physically relevant configurations are only 240 thus a two-RAM system is used: from the combination a first 8x8 kbit RAM is accessed which is used to access the 756x256 constant RAM.
The whole set of hits and the associated constants are extracted in parallel in a single clock and the Fitter Organizer sends a start signal to a Serializer.

Serializers

One Serializer can accept one combination every 6 clock cycles so there are 6 parallel Serializers and the Fitter Organizer keeps track of which one has to handle the fetched hit combination (see the sketch in figure 4.12).

Each Serializer registers the hits and constants, then serializes them associating each hit to the corresponding term in the constants set and sending one hit-constant pair for clock cycle to its own associated Fitter.

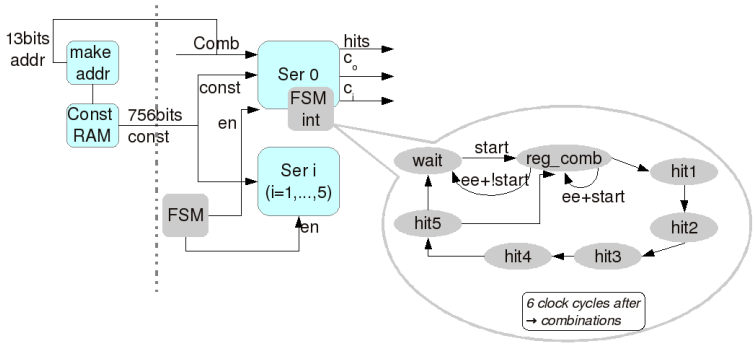


Figure 4.12: GF Serializer sketch

Fitters

Each Fitter receives the hits and constants data and calculates the track parameters and the fit quality parameters (χ^2 components). This function requires the

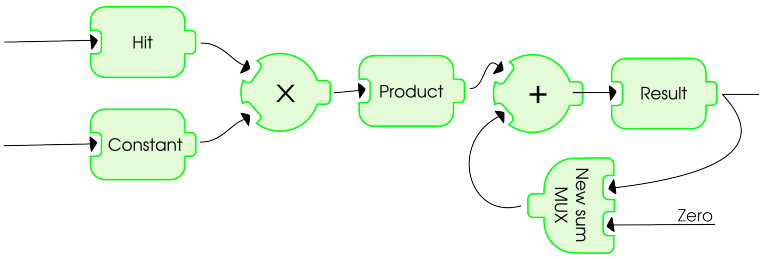


Figure 4.13: GF DSP Fitter unit: the scalar product unit is made inside the specialized DSP48 unit configured in MACC (multiply-and-accumulate) mode. Each unit can compute a scalar product in 6 clock cycles. The unit is ready to compute the next scalar product, although two additional clock cycles of latency are needed for the result to appear in output. Each Fitter consists of 6 such units, one for each fit parameter or χ^2 component.

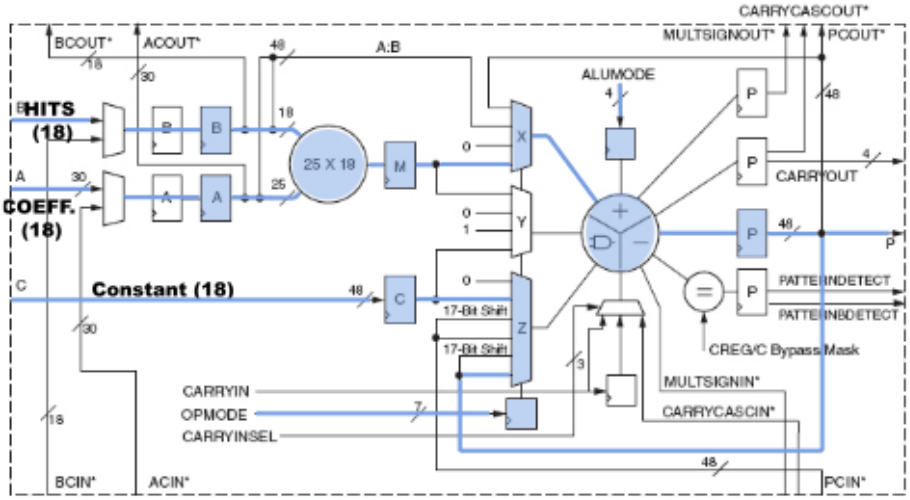


Figure 4.14: Sketch of a DSP48E slice: used resources are highlighted. Until the scalar product is not complete, the partial product P is sent back to the adder via the Z multiplexer. Z is controlled by the Control state machine via OPMODE. ALUMODE is fixed to use the DSP in MACC configuration.

computation of 6 scalar products, which are executed in parallel by exploiting the large number of DSPs of the Virtex 5 device. The multiplications of each 6-term scalar product (i.e. each fit parameter c, d, ϕ or one of the three χ^2 components) are performed by a DSP configured as MACC (multiply and accumulate) and serially processing the hits. The multiplier inside the DSP slice operates on two complements operands while data sent by the Serializers are coded with module and a sign bit. A conversion is then needed before the multiplication and it is done with simple logic ports. The scalar product will be reconverted into signed integers. As shown in fig. 4.14, hits and coefficients are in input in the A e B buses. The C input bus is used for the constant. The input buses width is fixed to 18 bits, that is the maximum width for the B input and is sufficient for both hits and constants. Controlling the OPMODE, the Z multiplexer is, at first, set to add the constant to the first product and then set to accumulate the subsequent products.
Six scalar products are obtainded at once after a 6 clock-cycles latency (figure 4.13). Thus, with the 6 Fitters, for a total of 36 DSPs, each associated to a Serializer, the GF is able to process one combination every clock cycle.
Once the results are ready, the χ^2 components are sent to the Comparator, while the track parameters obtained by the fit and the used hits are stored in the shift registers waiting for the Comparator decision.

Comparator

After the fitting algorithm has been carried out, the next step is to choose which hit combinations represent real tracks. The Comparator is the GF part in charge of this selection. Its structure is shown in figure 4.15.
The Comparator, at the right time, recieves additional information provided by the Combiners and maintained in shift registers because not used in the fit. This kind of information is particularly important when different fits of the same track have to be judged to choose the best one. As already mentioned, in fact, the GF has the capability to fit many times one track that has hits on all layers ("full of hits track" or 5/5 track) deleting one particular layer in each different fit and finally chooses the best. The Comparator has to fine-tune the final decision using not only the χ^2, but also the hit combination layout (used layers and quality of the hits).

The Comparator has the ability to choose the best track of an arbitrary sequence of tracks and the control bits going to the Comparator are thus set to consider the five 5/5 tracks as a sequence, while the 4/5 tracks are considered as one-track sequences. This system is flexible and we could use it to consider all combinations of the same XFT hit as a single sequence implementing a sort of Ghost Buster suppression at road level. This feature may be implemented in

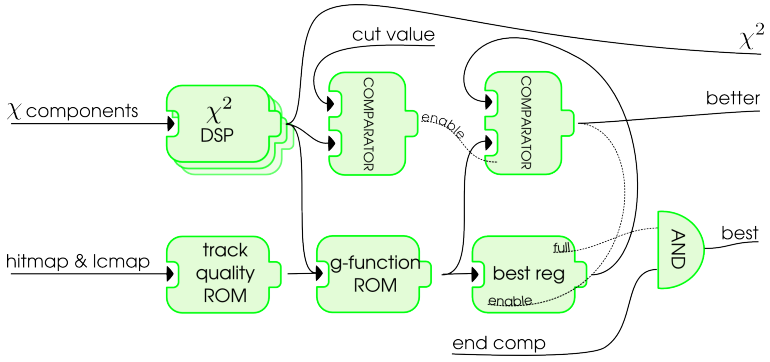


Figure 4.15: GF Comparator unit: the Comparator computes the χ^2 of the tracks from χ components, applies the χ^2-cut selection and compares the track with the previous one finding the best.

a future revision of the firmware. The Comparator calculates the final χ^2 using a DSP in MACC configuration like the one used in Fitter units (figure 4.13). Three clock cycles are necessary for each track and there are three of such units to sustain the output rate of one track candidate every clock cycle.
The Comparator compares the result with the threshold configurable via VME. If the track passes the χ^2-cut its χ^2 and the track quality (a function of used layers and hit quality) are used to compute a goodness function, called g, which is compared with the g of the best track in the sequence. If a better track is found, a signal is sent to update the registers that store the best track (parameters, χ^2 and additional informations). Once the sequence is finished, if there was at least one track passing the χ^2 cut a track-found signal is used to store the best track in the Track FiFos.

Formatter

Finally the Formatter reads the parameters and the χ^2 of the accepted tracks from the Track FiFos and merges all this information with the hits, the road identifier, and some status data, pushing them to the output in accordance to the SVT protocol.

4.5.2 The merger module

The same merger logic is used inside the mezzanine FPGA and Pulsar FPGAs to merge the various output data streams in a single stream. The merge is done in a simple and predictable way ("deterministic merge"): the inputs are ordered and the one with higher priority is read until an End Event packet is found. Then the

next is read and so on; after all inputs reach an End Event packet a final eEnd Event packet is sent to the output. The event tags in the End Event packets are used to check that all data streams are correctly synchronized. If the sequence of End Events is not correct in a stream, than a severe error (Lost Sync) is set. The error bit fields of the various input End Events are ORed in the final End Event packet sent to the output.
The "deterministic merge" is not optimal if the data stream occupancies are very unbalanced. If data do not arrive roughly at the same time on different streams, reading them in a predetermined order can be inefficient. A first-in-first-out fashion would be more efficient saving time, but the track order in the output will be unpredictable for the simulation. Their order would depend on timing details that are not available in the simulation. However the extra latency has been measured to be a small effect since the GF is working at a much higher clock frequency than the final output. In conclusion, the output is exactly predictable by the simulation.

4.5.3 Debug features

Diagnostics and debugging is a very important aspect for developing, commissioning and monitoring the status of the board during the normal operation. This aspect has been a key factor of the success of SVT and also in the GF board we implemented the standard SVT debug feature: spy buffers. The GF board is unique in SVT: it has 12 inputs, one output and performs the task that was previously of 15 boards. For this reason, the standard spy buffers at the end of input and output cables were not enough for fully monitoring and diagnosing the GF. It was necessary to attach spy buffers at each end of each "internal SVT cable" (figure 4.16): at the end of each track processing module inside a mezzanine, at the end of each mezzanine and at the end of each merge unit in the Pulsar board. This resulted in 30 spy buffers (12 input and 12 output, 3 mezzanines, 3 Pulsar FPGA), an unprecedented record for an SVT board, but the monitoring software was flexible enough to add all these spy buffers to the code without much effort.

There are also error registers that keep track of various kind of errors (fit overflow, FIFO overflow, invalid data, etc.) for each track processing module and for each merger module. Those registers are readable via VME to investigate online the status of every components of the GF board. There are several registers to configure the severity of each error and select to rise error bits on the EE word or to rise the standard SVT_ERROR and CDF_ERROR lines on VME backplane. When these kind of errors occur, it is possible either freeze spy buffers of all SVT or perform a reset of the DAQ system.

Another tool that has been extremely useful for in depth debugging of the GF Mezzanine firmware is the ChipScope tool from Xilinx. ChipScope is a suite

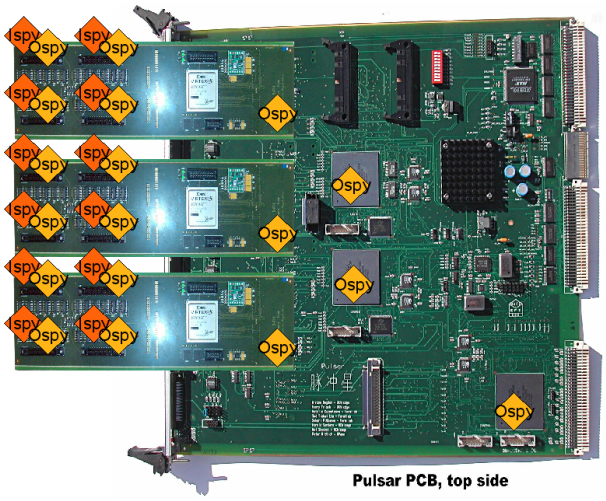


Figure 4.16: GF spy buffers: the GF board is the most complex board of SVT in terms of diagnostic features: it has 30 spy buffers, one for each input and output of track processing module and one for each output of merger units.

of firmware modules (cores in the Xilinx jargon) and a standalone PC software. With the firmware cores is it possible to insert a custom logic analyzer and pattern generator in-chip and fully controllable via the JTAG programmer cable. Using ChipScope it is possible to analyze lots of logic lines, for example all 756 bit constants, 15 bit hits and 48-bit partial fit results at once. The number of debug pins that are available on the PCB (20-pin connector for the Mezzanine), the capability of external logic analyzer and all the problems of routing signal copies to the debug pins are not limited using ChipScope.
The ChipScope features are disabled in the stable version of the GF Mezzanine firmware.

4.6 Configuration for parasitic tests

A very careful test procedure was devised to minimize to a negligible level the impact of the commissioning on detector operation and functionality. The new board has to be deeply tested before being allowed to enter the experiment.
After a first validation, performed on a stand-alone test stand using a Merger board (introduced in sec. 3.2.5) to send and receive data from the GF board, the output was compared to board level simulation. A second level of validation has been performed by using real data spied from the experiment (parasitic mode).
Any GF test phase has been carried out during the 2009 shutdown period (15

June - 15 September) and the first months of data taking with beam (September 2009 - January 2010) [34]. The GF has been installed in `tstsvt2`[5] crate in trigger room. The outputs of the 12 HB++ have been splitted using three Splitter[6] boards and 6 Merger boards, in order to provide the GF with an exact copy of the data in input to the TF++ boards. Three Splitters and three Merger boards have been installed in crate `b0svt09` and three additional Merger boards in crate `b0svt07`. The latency added to the SVT timing by the splitting boards has been measured to be $\sim$ 100 ns, negligible with respect to the overall SVT timing ($\sim$ 25 μs). An additional Merger board has been added in crate `b0svt09` to split the Bypass [35] signal. This signal is necessary at high luminosity to reduce the overall SVT timing: SVT track reconstruction is bypassed for events not required to pass a SVT based selection. The configuration used for the test is shown in fig. 4.17. The GF output is connected to the Bypass board through a Merger. The Bypass output goes to the GB board. The GB board is in the crate to receive GF output and perform all the final corrections exactly as the real SVT. In this parasitic configuration, highlighted in figure 3.6, it is possible to process the same data as the TF++ boards without interfering with normal data taking and thus to make a direct comparison between the current system and the upgraded.

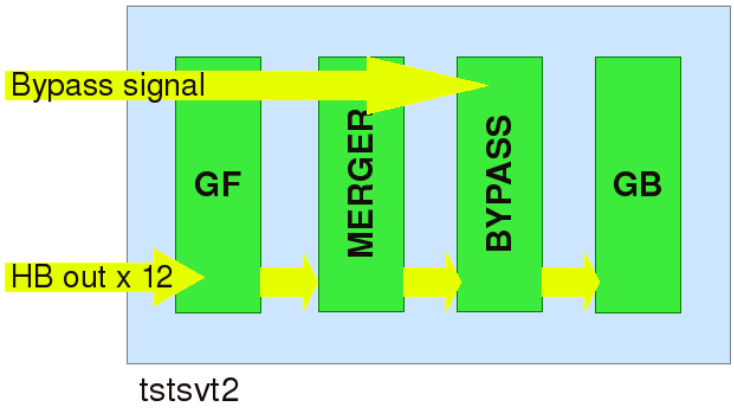


Figure 4.17: Scheme of GF configuration in crate `tstsvt2`.

The output of the crate can be compared directly with that of the SVT fi-

[5] `tstsvt2` is a test crate, placed near the real SVT, configured to duplicate the track fitting function.

[6] The Splitter board has two inputs and four outputs. Each input is copied into two outputs with separate handling of hold signals like the Merger. This board is not used in the normal SVT pipeline but it's used for the parasitic configuration

nal crate receiving identical input data. At this step all discrepancies between the two systems are completely understood. The same inputs are also fed to a board-level simulator. Comparison between the hardware and the simulation are used to validate both the board and the upgraded simulation itself.

The GF studies in parasitic configuration are exposed in the next chapter.

4.7 GF commissioning

After successfully passing step 4.6, the new GF board can replace the old TF++s for a short, low-luminosity test, and, after being successful, for data taking at any luminosity in a whole store. This final test is important because it checks that the control signals used by the DAQ system do not interfere with the board functionality and vice versa. Furthermore, it is an extra validation with higher statistics than the previous tests, allowing for detection and debugging of lower rate errors.

The GF had to work correctly in data taking both at low and high instantaneous luminosity before being considered ready to be installed and remove all the TF++: a commissioning period of one week of GF data taking in the final position (not in the test crate) is required before decommissioning and remove TF++.

The HB++ splitted outputs are not removed after the commissioning. The test crate will be used to validate the spare GF boards and also to quickly develop and test new improvements to the system.

The GF system has currently passed the final review from the Collaboration and started the final commissioning period. It is planned that the GF will be operational in SVT by the end of March 2010.

4.8 Monitoring tools

4.8.1 Online monitoring tools

To online monitor the performance of the GF we have modified the main SVT online monitoring tools, `TRIGMON` and `SPYMON`, to include the GF data.

- `SPYMON` [36] is a monitoring program running on the CPUs of the SVT crates. It collects spy buffer data, reads board error registers and publishes the following three kinds of messages:

  - SVT histograms filled with spy buffer data quantities;
  - SVT status message containing error registers and status words for each of the boards in the crate and statistics about individual error register flags;
  - spy buffer dumps, i.e. a lengthy string containining the formatted dump of (part) of the spy buffer data.

  A custom version of `SPYMON` has been developed to run on `tstsvt2` crate to monitor errors, track multiplicity, SVT occupancy and track parameters in output from the GF and the GB board.

- `TRIGMON` [37] is a low level online trigger diagnostic and monitor. It is comprised of many modules, each monitoring a specific trigger bank. Official SVT data are monitored by `SVTDmonitor` [38] module, which exploits the information contained in the `SVTD` bank [39] written by the GB board. If the `tstsvt2` crate is included in the data taking, GF data can be saved in the `SVDD` bank [39], a SVT internal diagnostic bank which can be written by the GB board. This bank is organized in three cards, containing:

  - the beam fit result for each SVX II barrel,
  - the result of the 3D beam fit,
  - a copy of the tracks stored in the SVTD bank.

  A custom version of `SVTDMonitor` has been developed to read the information stored in the `tstsvt2 SVDD` bank and fill histograms with track parameters and beam fit results. `SVTDMonitor` output is a root file containing all the relevant SVT plots. A directory called *VSLICE*[7] has been added and is filled with GF data if *_useVSLICE* parameter is set true in the `TRIGMON` tcl file.

4.8.2 Offline monitoring tools and simulation

If the `tstsvt2` crate is included in the data taking, the `SVDD` bank is saved on disk and it can be analyzed offline using the SVT offline monitoring tool, `SVTMon`, enabled to read the GF data[8].

`SVTMon` can also run the SVT simulation `svtsim`, to be compared with the hardware response. The GF simulation has been developed following `svtsim` structure: GF or TF++ can be enabled by the correspondent flag to be set in the file `svtsim_lib.h`.

[7] In the past `tstsvt2` crate was used to reproduce a single wedge SVT sector (a so-called Vertical SLICE): we have mantained the same name.

[8] *_useVSLICE* parameter set true in the `SVTMon` tcl file.

Chapter 5

Gigafitter performances

At first the GigaFitter is going to be installed to exactly replace the TF++ system: the same constant and pattern sets will be used. In this chapter I measure the GF performances against the TF++ in terms of:

- the impact of the GF new features (full precision fits, treatment of 5/5 tracks) on parameter calculations and track multiplicity;
- the resolution on the track parameters;
- the impact on the beam position;
- the impact on the SVT timing.
- the track reconstruction efficiency and purity.

The minimum requirement for this first step of the GF installation is a performance equal or better than the current system.
Future Gigafitter improvements may increase the SVT performances. The current SVT efficiency will enhance by recovering unexploited kind of tracks, larger data banks, and consequent constants sets. In fact the current pattern bank is made from tracks with $p_T > 2$ GeV/c and a beam spot radius of 0.14 cm; tracks crossing mechanical barrels are not generated, thus no patterns for those tracks are in the pattern bank. The larger is the parameter acceptance (for example lowering p_T threshold to 1.5 GeV/c or allowing mechanical barrels crossing of SVX II detector) the lower will be the coverage at a fixed number of patterns and road size. Moreover the actual constants sets are generated from tracks with the same parameter distributions as the patterns but with all hits in the same barrel.

The current SVT system performes non-linear corrections, beam position subtraction and duplicate tracks suppression in the dedicated Ghost Buster board placed after the TF++ boards. In a next GF we could perform it inside the

Pulsar board, with two advantages: first, we can work in parallel on different wedges; second, we can reduce the number of tracks before merging the output of the 12 wedges, thus saving data transfer time, specially with crowded events at high luminosity.

5.1 GF new features

5.1.1 Full precision fits

The effects of full precision computation done by the GigaFitter were studied with the simulation in sec. 4.1.1. The full precision fit done by the GF leads to a different χ^2 with respect to the same computation done by the TF++ (see figure 5.1).

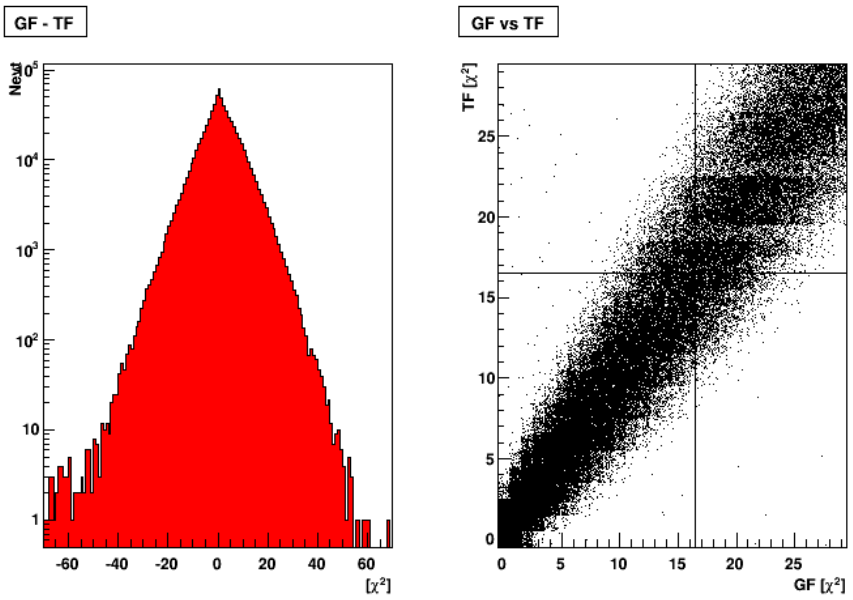


Figure 5.1: Differences in χ^2 computation between GF and TF++ due to the term not computed by TF++. Current cut values are shown with the solid lines. The same figure is also shown in sec. 4.1.1 and reported here for convenience.

The effect of using the GF instead of the TF++ is a small gain in the amount of tracks above the χ^2 threshold that were accepted by the TF++ (fig. 5.1(right)). About 2% the total number of tracks is selected only by the GF or the TF++, but not by both. We will see in 5.5 that the GF is more efficient on real tracks than the TF++ of about the same percentage without increasing the number of

fake tracks recontructed by the system, so that overall the χ^2 computed by the GF is more accurate on the good tracks selection.
The differences on the calculation of the other parameters are instead within the resolution (see Sec. 5.2).

5.1.2 Track parameter studies

The GF performs a different treatment of 5/5 tracks and a better resolution calculation with respect to the TF++. For these reasons the GF overall reconstructs more tracks than TF++ ($\sim 2\%$).
For tracks reconstructed by both systems, we can make a detailed comparison of the track parameters track by track, while for those tracks selected by GF we can only check that the parameter distributions are consistent with the correspondent L3 track distributions.

Tracks reconstructed by both GF and TF

We consider 4/5 tracks reconstructed and selected by both GF and TF++. The tracks are matched if they have the same XFT parent and belong to the same road.
In fig. 5.2 we show the curvature distribution for TF++ (upper left), GF (upper right) and both superimposed (lower left). The RMS of the difference distribution (lower right) is $RMS_{\Delta c} = 7{\cdot}10^{-6}$ cm^{-1}, well below the resolution on this parameter (see Sec. 5.2). As a further check, fig. 5.3 shows for each track the curvature calculated by the TF++ as a function of the correspondent value obtained by the GF.
The same plots are shown in figg. 5.4-5.7 for the azimuthal angle and the impact parameter. In these cases $RMS_{\Delta\phi} = 4{\cdot}10^{-4}$ rad and $RMS_{\Delta d_0} = 7\ \mu$m, well below the resolution.

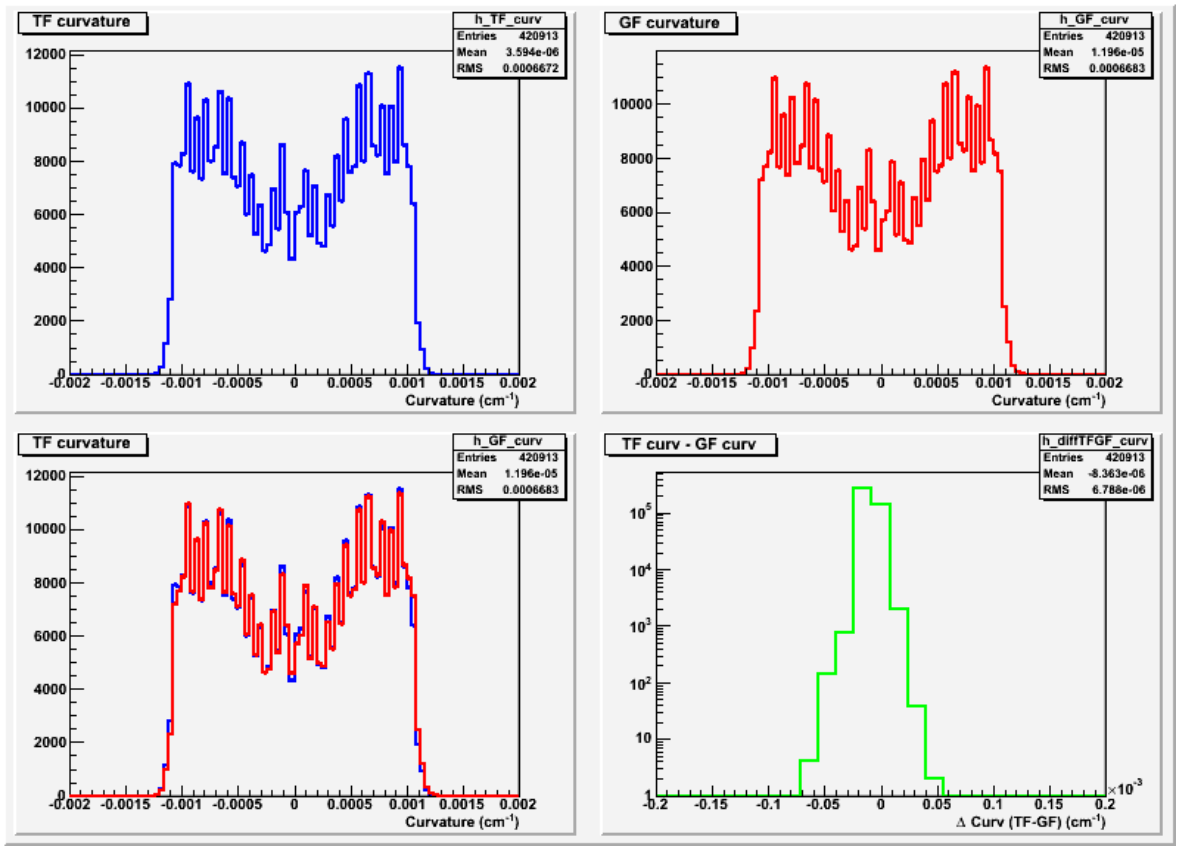


Figure 5.2: Curvature distribution for tracks reconstructed by both GF and TF++: the distribution of the track by track differences is shown in the lower right plot.

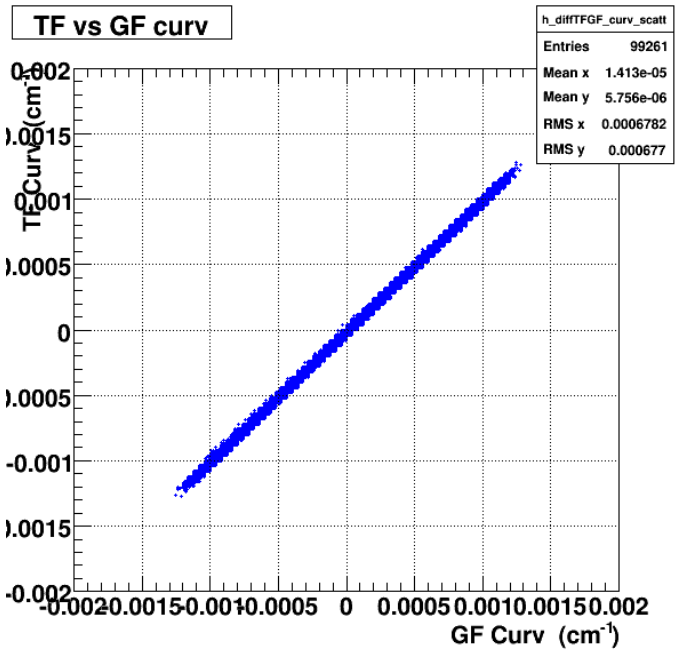


Figure 5.3: Track curvature calculated by TF++ as a function of the value obtained by GF.

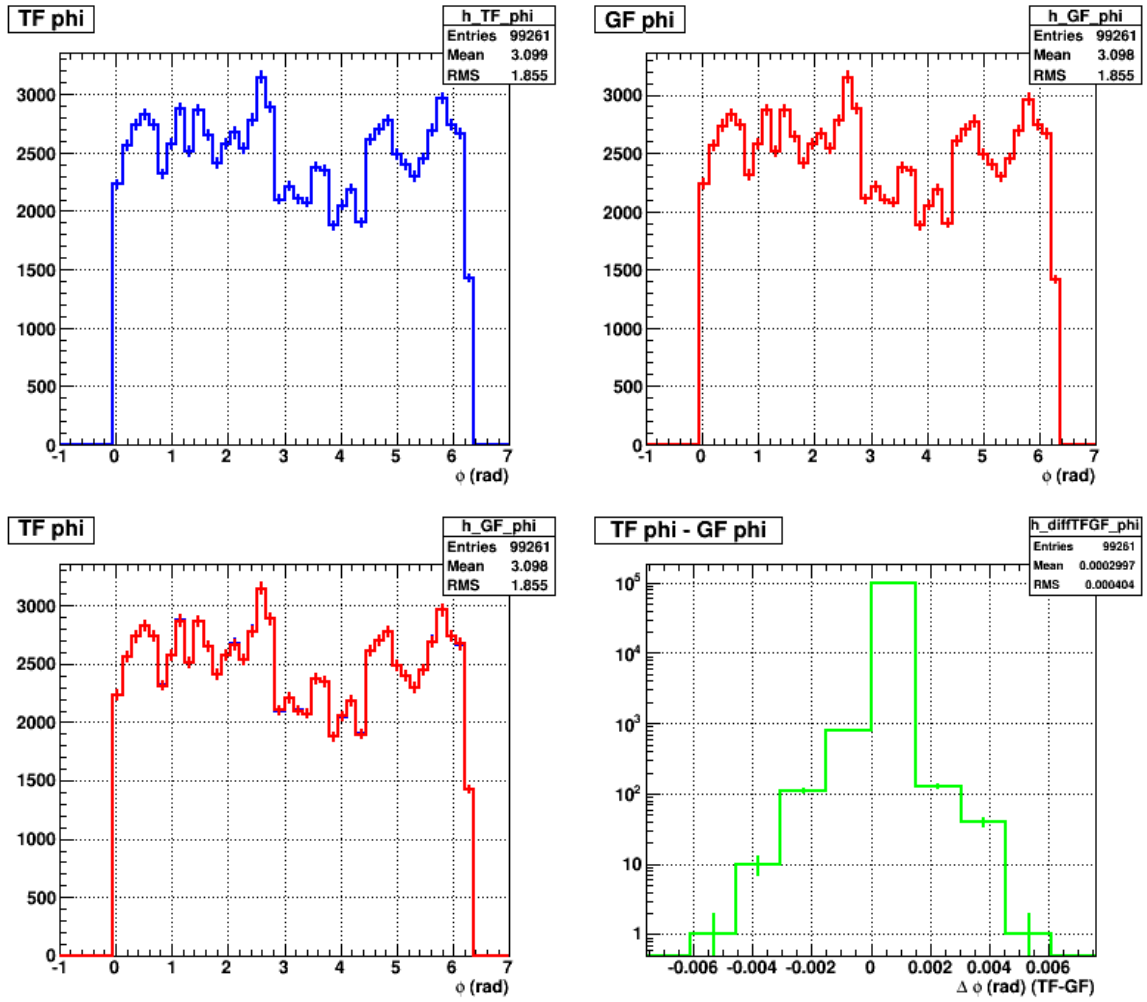


Figure 5.4: Azimuthal angle distribution for tracks reconstructed by both GF and TF++: the distribution of the track by track differences is shown in the lower right plot.

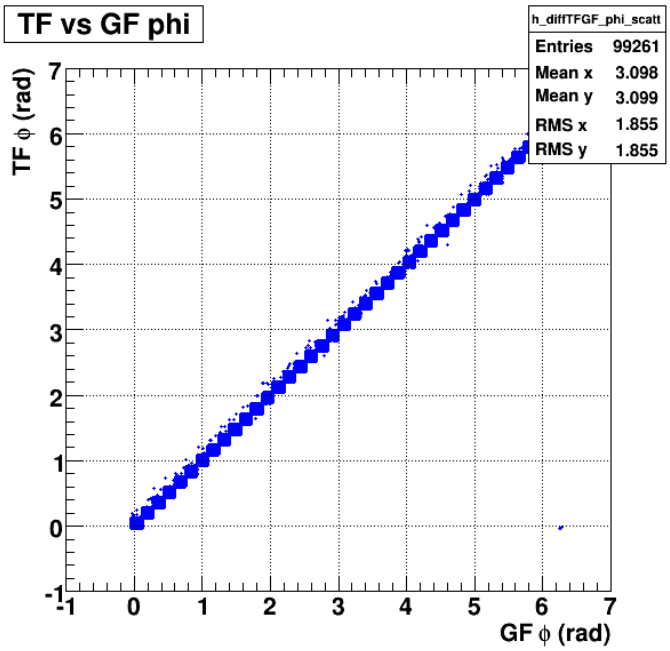


Figure 5.5: Track azimuthal angle calculated by TF++ as a function of the value obtained by GF.

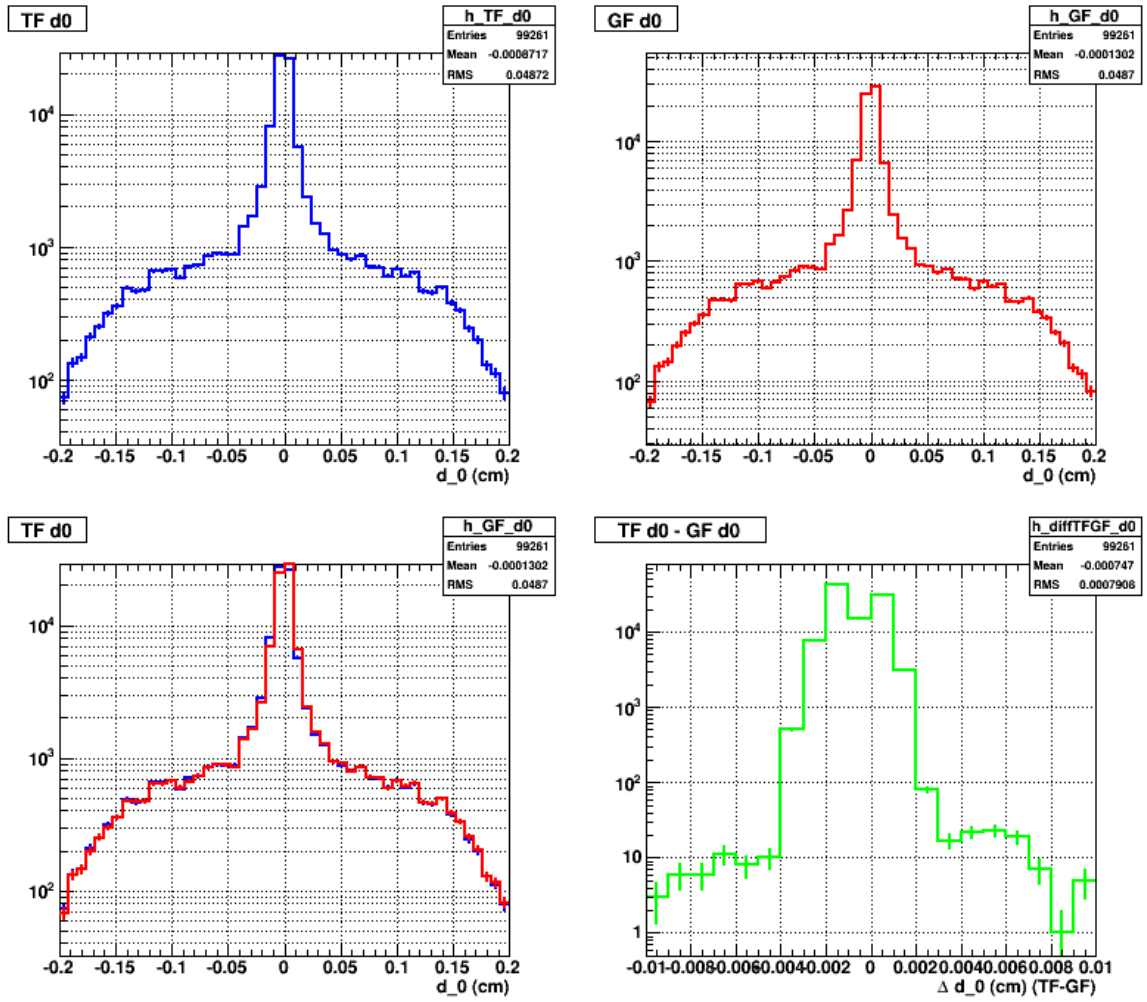


Figure 5.6: Impact parameter distribution for tracks reconstructed by both GF and TF++: the distribution of the track by track differences is shown in the lower right plot.

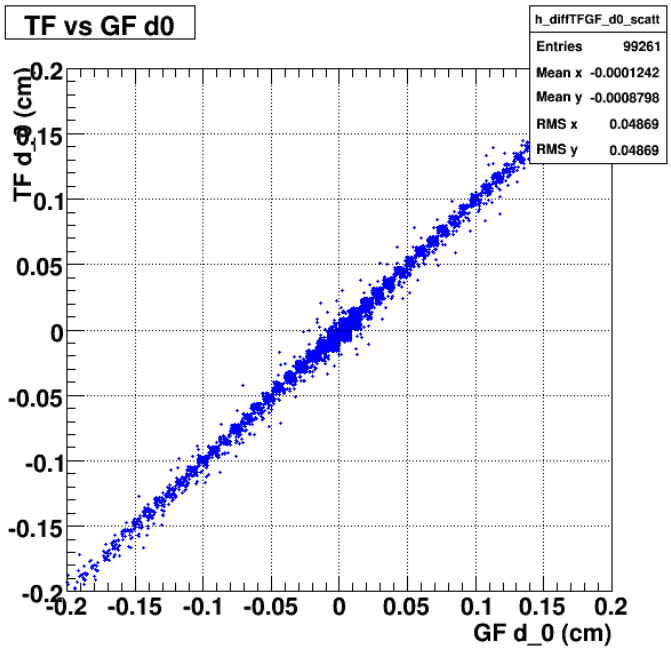


Figure 5.7: Track impact parameter calculated by TF++ as a function of the value obtained by GF.

GF only tracks

For tracks reconstructed by the GF only, we cannot make a direct comparison with TF++ tracks. Anyway, to check that the GF tracks not matched to TF++ ones are well reconstructed, we match them to L3 tracks ($|\Delta\phi| < 0.02$ rad and $|\Delta c| < 0.0002$ cm^{-1}) and compare the parameter distributions. The fraction $N_{GF-only}^{matched-L3}/N_{GF-only}$ of tracks reconstructed only by the GF that match L3 tracks (i.e. the purity for GF only trakcs) is $66.2 \pm 0.8\%$, to be compared to $N_{TF-only}^{matched-L3}/N_{TF-only} = 46.4 \pm 1.6\%$ for tracks reconstructed only by the TF++ systems (see sec. 5.5). The distribution of d_0, ϕ and c for the GF-only tracks is compared to the L3 calculation in figg. 5.8- 5.10.

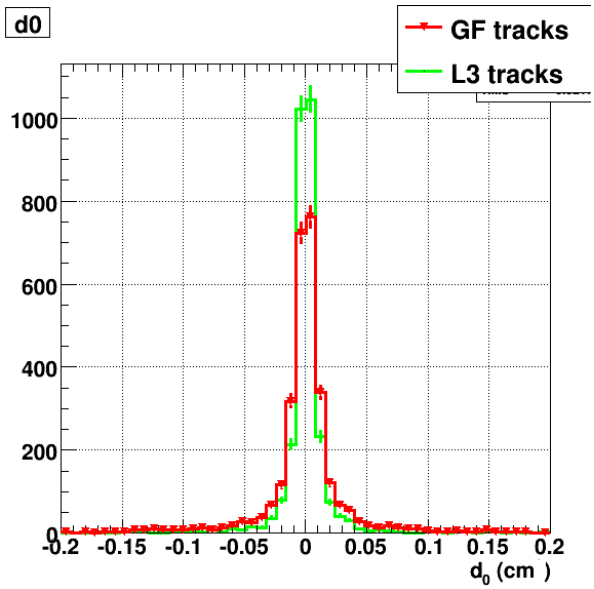


Figure 5.8: d_0 distribution for tracks reconstructed by GF only and the corresponding L3 tracks.

5.2 Resolution on parameters

The resolution on fitted parameters is measured maching the SVT tracks to the L3 reconstructed tracks with at least 4 silicon hits. Figures 5.11, 5.12 and 5.13 show the distributions of the d_0, ϕ and curvature differences (distributions are normalized at 1) between the SVT tracks and the L3 ones. The resolution on the fitted parameters is measured fitting the distributions with a linear plus gaussian

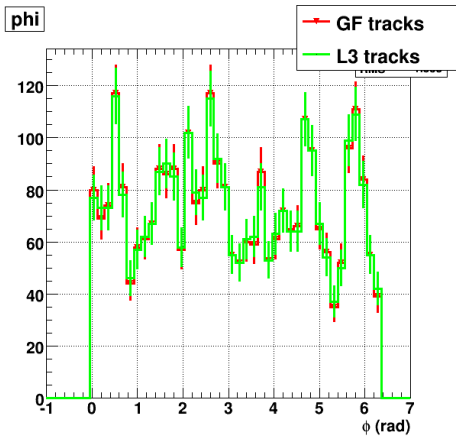


Figure 5.9: ϕ distribution for tracks reconstructed by GF only and the corresponding L3 tracks.

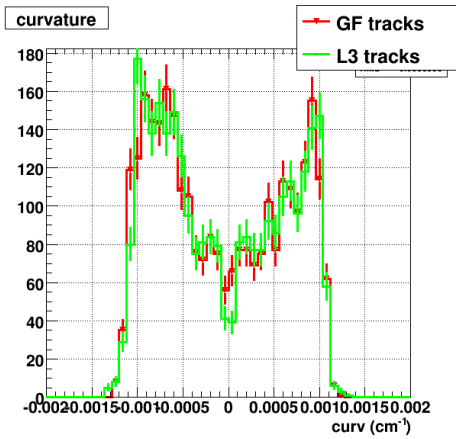


Figure 5.10: Curvature distribution for tracks reconstructed by GF only and the corresponding L3 tracks.

Resolutions	d_0 (μm)	ϕ (mrad)	c ($cm^{-1} \cdot 10^{-5}$)
GF	86.4	0.518	2.65
TF	84.6	0.515	2.69
$\frac{2(GF-TF)}{GF+TF}$ (%)	2.1	0.6	1.4

Table 5.1: Resolution measurements on tracks parameters. The statistical errors on the fit is smaller than the number of digits shown.

function: the standard deviation of the gaussian is the measurement of the resolution on the parameter. The linear component of the fit accounts for wrong matching of the SVT track with the L3 one. The table 5.1 shows the resolution measurement sepately for the GF and TF++. The GF and TF++ have nearly the same resolution, within 1.5%, on all the fitted parameters: $\sim 80\mu$m on d_0, ~ 0.5mrad on ϕ and $\sim 3 \cdot 10^{-5}$cm^{-1} on the curvature c.

5.3 Beam Fit using GF tracks

In SVT the beam position is measured in real time by a task running on the crate hosting the final Merger, the board merging all the data streams of the 12 TF++. The impact parameter d_0 and azimuthal angle ϕ of each track are measured with respect to the origin (0,0). If the beam spot is displaced with respect to the origin and has coordinates (x_0,y_0) in the transverse plane, there is the following relationship between d_0 and ϕ for tracks originating from the primary vertex:

$$d_0 = x_0 * sin(\phi) - y_0 * cos(\phi)$$

and the d_0 vs ϕ distribution has a sinusoidal shape.
The beam fit algorithm reads the track list from the spy buffer of the final Merger and fits the d_0 vs ϕ distribution to obtain the beam position in each of the six barrels. A linear fit to the six values obtained in the different barrels returns the beam line direction, expressed as $\frac{dx}{dz}$ and $\frac{dy}{dz}$, and the mean beam position (x_0,y_0). Small differences in the resolution, as seen in Sec. 5.2, of the track parameters measurement of the GF with respect to the TF++ can have an impact on the beam fit calculation. Moreover, the GF reconstructs more tracks than the current system, due to the different treatment of the 5/5 tracks. As example, in fig. 5.14 the distributions of beam X and Y coordinates is shown as measured by the two systems. The small difference, about 20 μm, in the mean of the position is smaller than the 30μm resolution.

A summary of the beam spot positions in x and y coordinates for all barrels is shown for the GF (red crosses) and the TF++ (blue circles) in fig. 5.15. Barrel

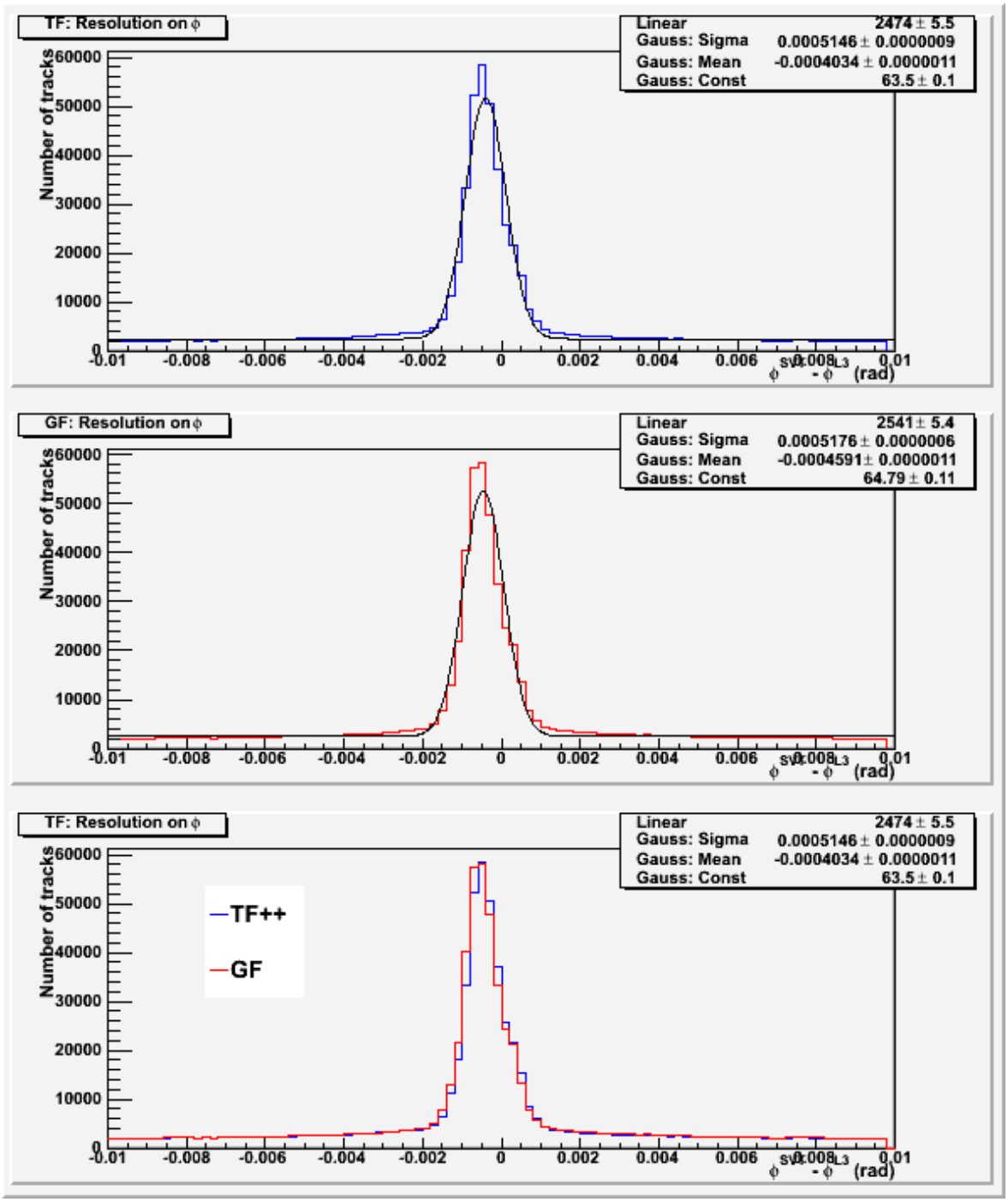


Figure 5.11: Distribution of the ϕ (rad) differences between SVT and L3 tracks for TF++ in blue (upper plot), GF in red (middle plot), and both (lower plot).

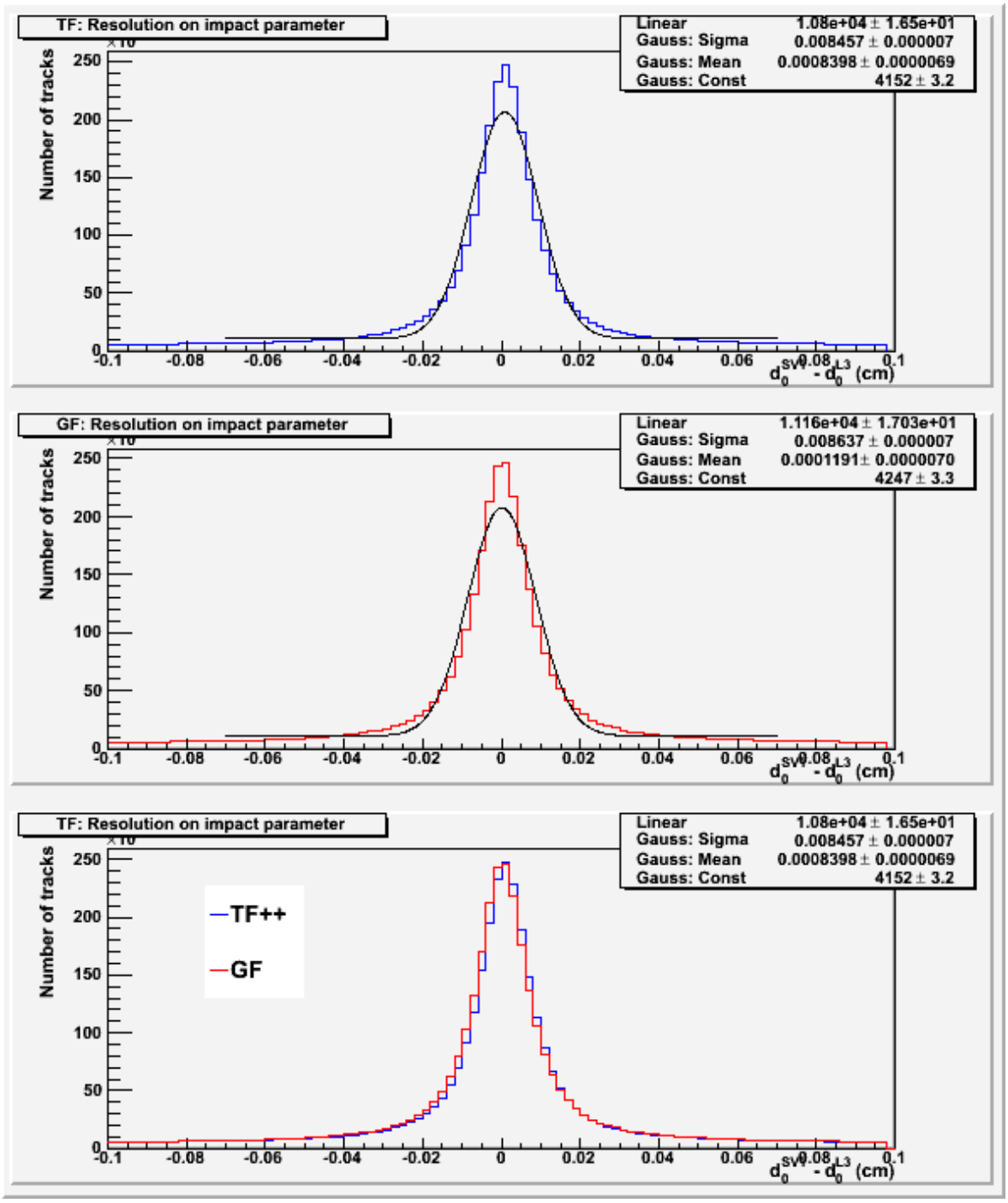


Figure 5.12: Distribution of the d_0 (cm)difference between SVT and L3 tracks for TF++ in blue (upper plot), GF in red (middle plot), and both (lower plot).

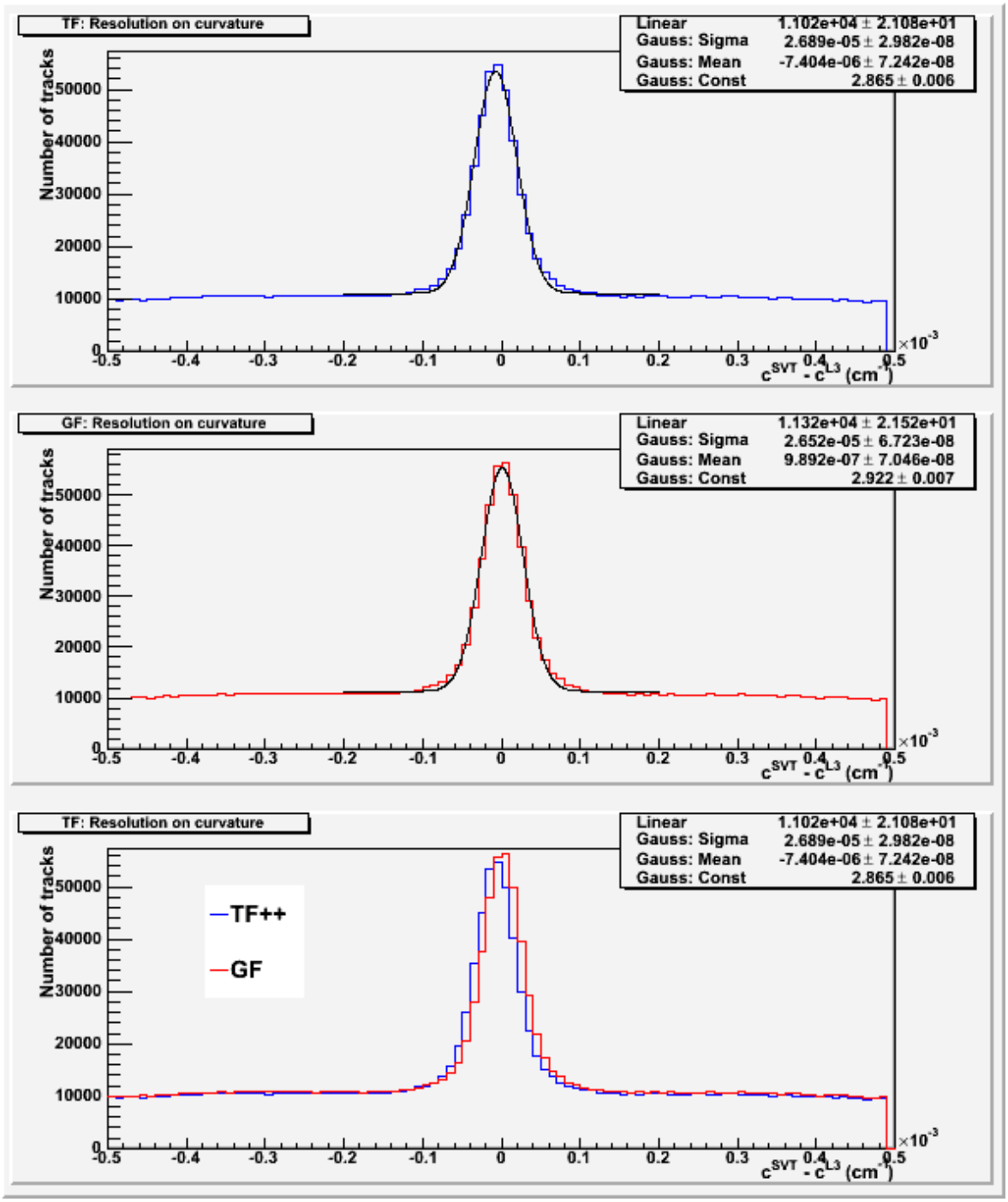


Figure 5.13: Distribution of the c (10^{-3}cm^{-1})differences between SVT and L3 tracks for TF++ in blue (upper plot), GF in red (middle plot), and both (lower plot).

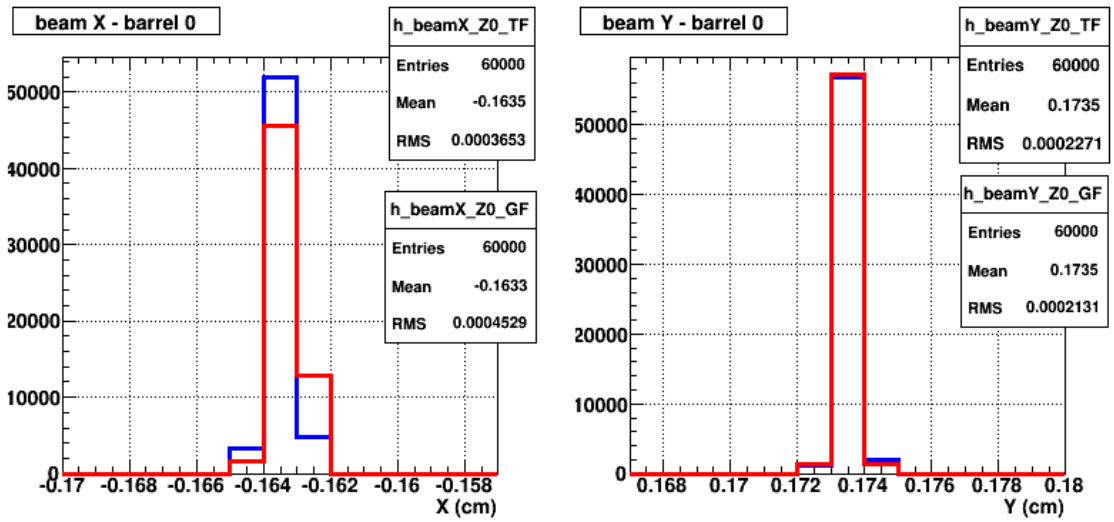


Figure 5.14: GF vs TF++: X and Y coordinates of beam position for barrel 0 (Blue TF++, red GF).

1 shows a small displacement of the x coordinate with respect to the TF++, but within resolution.

The distribution of the result of the linear fit to the 6 values obtained in the different barrels result in the beam coordinates and slopes shown in fig. 5.16: we notice a difference of about 2 and 4 μrad in the mean of the $\frac{dx}{dz}$ and $\frac{dy}{dz}$ measurements; these displacements are within the resolution.

5.4 Processing time

The SVT processing time is determined by multiple factors:

- the latency due to data processing by each board (the Hit Finder, the AMSRW/AM, the HB++ and the TF++), that is proportional to the complexity of the event;
- the data transfer[1] latency through SVT boards, which depends on the number of present hits, the number of found roads and the number of tracks that pass the quality and χ^2 cuts;
- the merge time of the twelve wedges before the GB[2] processing.

[1]All data go from one board to another using the standard SVT cable at 32 MHz (certain boards can run up to 40 MHz, such as HB++ output).

[2]GB output frequency is 20 MHz, slower than any other transfer rate.

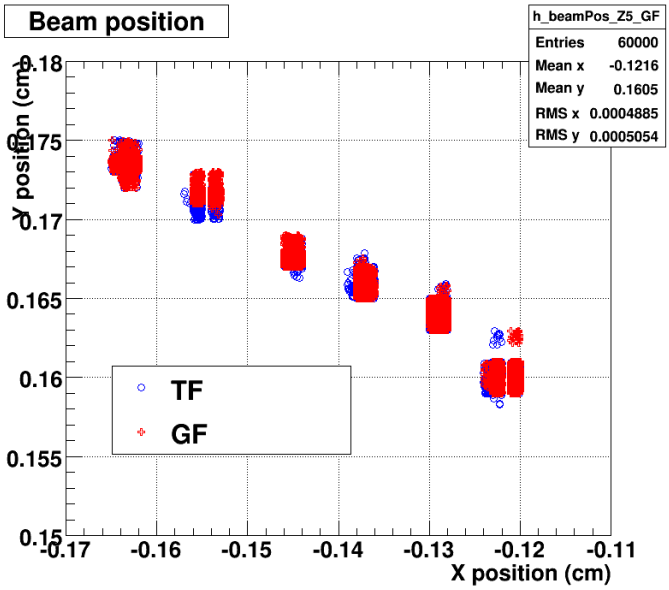


Figure 5.15: GF vs TF++: X and Y coordinates of beam position for all barrels.

The GF receives and sends data at the same speed as the TF++. Unfortunately it's not possible to highlight directly the timing difference between the single TF++ and the logically equivalent module inside the GF Mezzanine FPGA. We can measure only global event timing between the first road arrival at both tracking devices and the last track output at the very end of the GF Pulsar, when the data streams are all merged.
For each event the GB board, the last board of the SVT chain, measures the overall SVT processing time as the difference between the L1 Accept time (the start for level 2 processing and SVX readout) and the arrival time of the End Event word in the GB itself.

Fig. 5.17 shows the SVT timing for the system with the TF++ (blue) and for the system with the GF (red): the global event processing time across the GF path is almost the same as the timing across the TF++ path; the GF does not have any impact on the overall SVT timing. .

Any possible gain in timing due to the much higher track fitting rate provided by the GF is hidden under the bottlenecks caused by the downstream and upstream boards. The GF does quickly its job but spends some time inactive waiting for the inputs. For this reason the global event processing time across

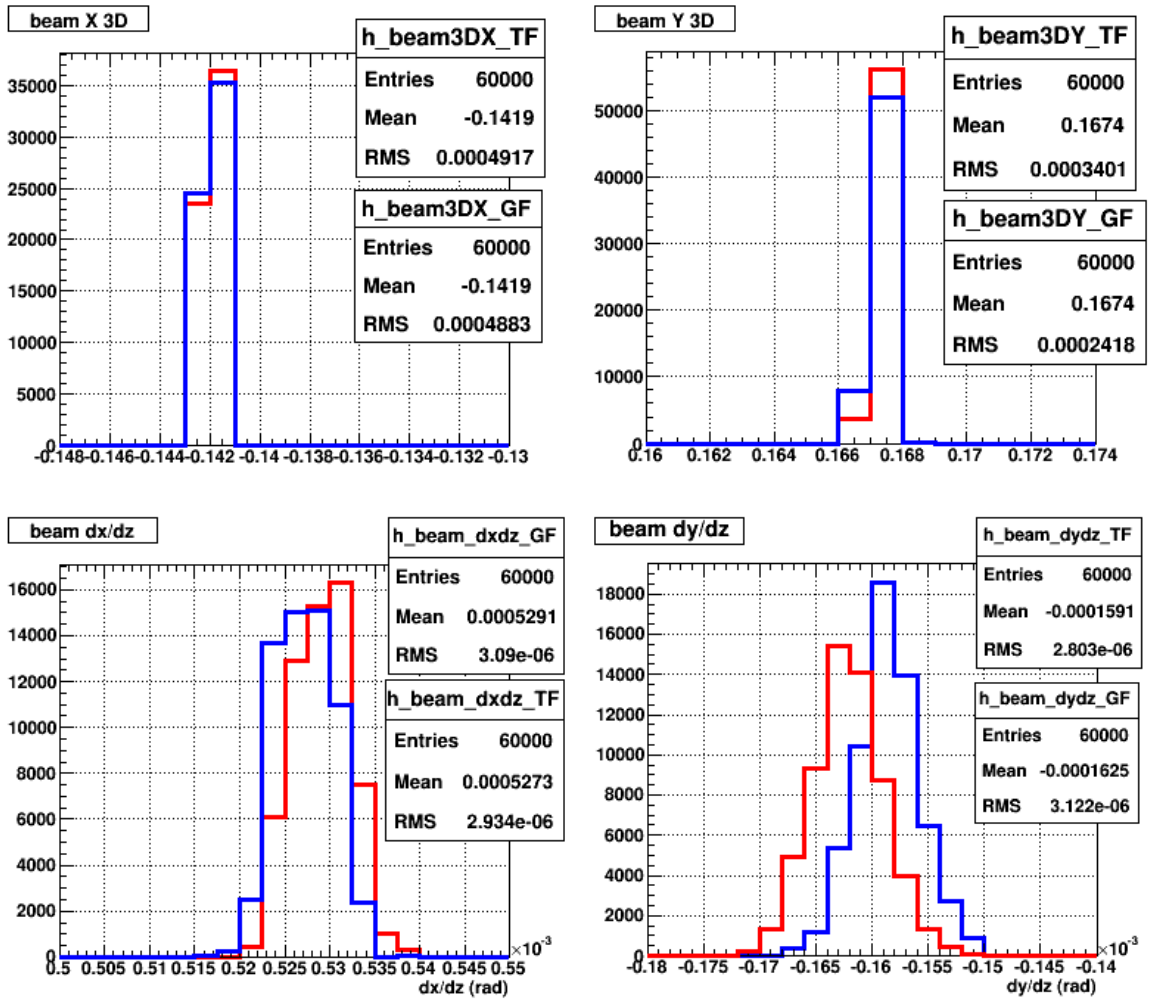


Figure 5.16: Beam coordinates slope.

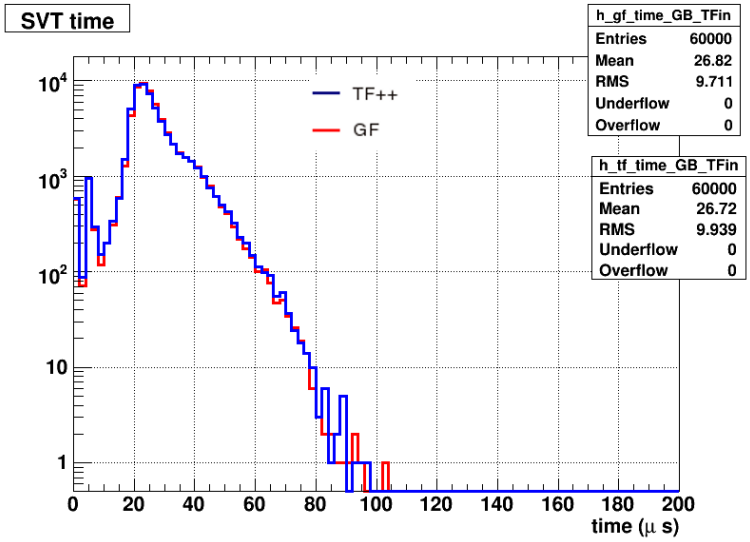


Figure 5.17: SVT overall timing measured by the Ghostbuster board: red line is GF, blue line TF++. Data is taken for both systems at the same time on the same events.

the GF path is almost the same as the timing across the TF++ path.
On some classes of events, anyhow, it is possible to note a certain difference in the timing. A check on the events with many hit combinations is shown in figure 5.18: on the left the mean processing time versus the total number of hit combinations summed over all wedges, on the right the fraction of events with timing falling in the tail section of the global timing distribution (> 50 μs). In these plots the GF appears to be faster, but the effect is small. In fact, when the number of combinations or fits increases, also the number of roads to be transferred to the track fitters and the number of found tracks to be merged increase as well. This implies the same growth for both the TF++ and GF timing.

In the range of low combinations both systems (TF++ and GF) are limited by the input rate from the HB++ and the timing is absolutely similar, instead, when the events have roads with a large number of combinations (approximately over 64, which is the limit of roads per wedge, meaning that on those events roads are carrying more than one combination), a difference appears even if both systems are still limited by the merge rate. The GF is significantly faster when the event complexity goes up. The difference of the fraction of events with long processing time reaches about 20% in the 160-170 combinations range. This is an important feature, much more than the lower mean processing time. The tails in fact have a direct effect on the dead-time of the DAQ system.

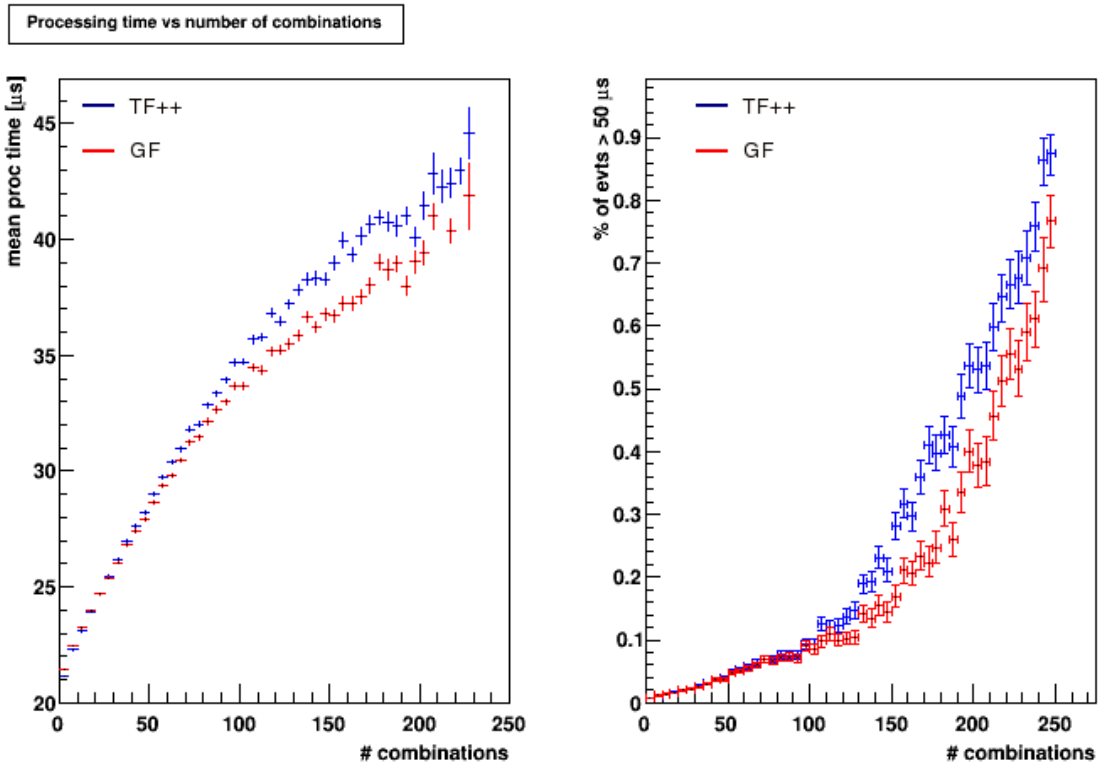


Figure 5.18: The standard SVT system is in blue, the GF parasitic test crate is in red. On the left is shown the mean processing time with respect to the number of processed combinations. On the right the fraction of events with processing time > 50 μs. The processing time in these plots is computed subtracting I/O time to GB board and from GB board to L2, which is linearly dependent with the number of found tracks.

A figure of the GF system power is shown in figure 5.19 were is plotted the mean time with respect to the number of fits performed. The GF is performing much more fits than the TF++ system even if the number of combinations to be processed is the same for the two processors, because the GF fits five times every full hits (5/5) track and then chooses the best fit, without adding latency to the system.

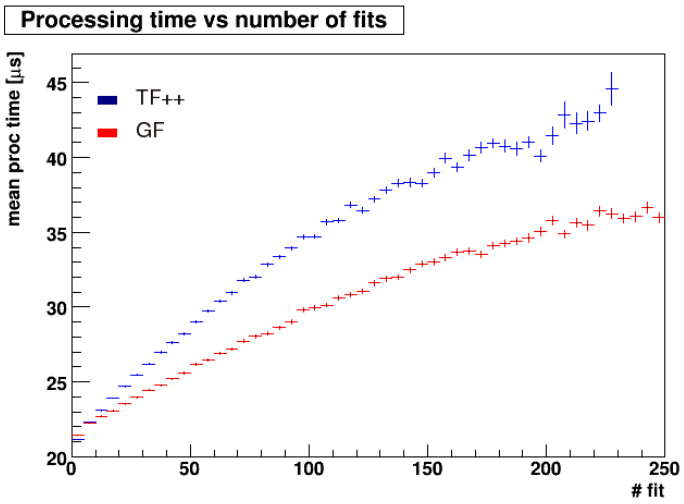


Figure 5.19: The standard SVT system is in blue, the GF parasitic test crate is in red. The mean processing time versus the number of fits done is show, the GF does much more fits than the TF++ system because it fits 5 times every 5/5 track.

The GF is not expected to lower the SVT processing time because of the said bottlenecks (HB++ in input, GB in output) and also because of the current pattern banks and road size: the current system is tuned to be balanced with respect to the computing capabilities of each component of the pipeline. The processing time spent in the TF++ is limited by the small road size that limits the maximum number of hit combinations per road. The GF however allows now to change with more freedom the road size, because it's able to sustain a larger number of fits without impacting on the global timing. The GF is in fact able to do one fit every clock cycle at 120 MHz on each wedge once the roads are loaded.

5.5 Efficiency and Purity

The track reconstruction efficiency is measured with respect to

- L3 tracks fiducial to SVX II wedges;
- L3 tracks fiducial to II wedges and with at least 4 hits in the silicon detector.

The efficiency ϵ is calculated as

$$\epsilon = \frac{N_{SVT}^{matched}}{N_{L3}}$$

where $N_{SVT}^{matched}$ are all the tracks reconstructed by SVT *and* matched to a L3 track, while N_{L3} are all L3 tracks having at least 4 SVX hits, $p_T > 2GeV/c$ and a matching XFT track. Matching between SVT and L3 is defined as $|c_{SVT} - c_{L3}| < 0.0002$ cm^{-1} and $|\phi_{SVT} - \phi_{L3}| < 0.02$ rad. The purity is calculated as

$$P = \frac{N_{SVT}^{matched}}{N_{SVT}}$$

where N_{SVT} are *all* tracks reconstructed by SVT. A low purity means that many tracks are incorrectly reconstructed, i.e. the *fake rate* is high. The fake rate is calculated as

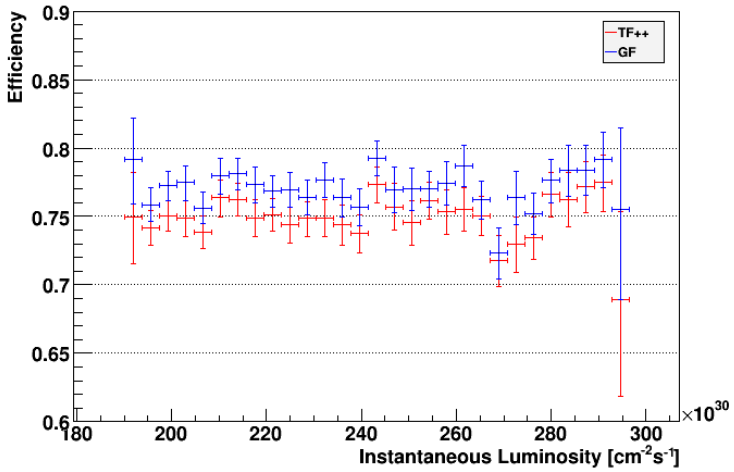
$$FR = \frac{N_{SVT}^{not-matched-L3}}{N_{SVT}}$$

where $N_{SVT}^{not-matched-L3}$ are tracks reconstructed by SVT with no L3 matching. Figure 5.20(a) shows the efficiency for the TF++ and GF paths as a function of the instantaneous luminosity ($290 \cdot 10^{32}$ to $190 \cdot 10^{32}$ cm^{-2}s^{-1} high luminosity range of a recent CDF store): the use of the GF leads to a significant increase in the efficiency of about 2%, even if we are using exactly the same constant and pattern sets. The fake rates are instead very similar for both systems as shown in figure 5.20(b) as a function of the instantaneous luminosity.

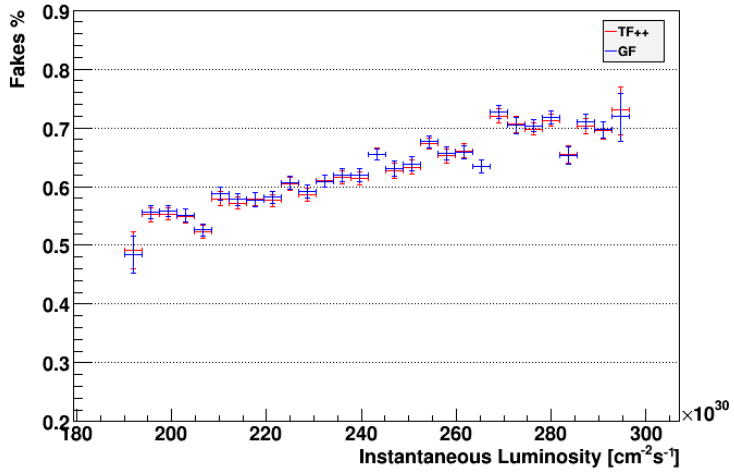
The GF finds about 45% of 5/5 tracks more than the TF++ system for a total of 2% more tracks at the track fitting output. This is the main factor that contributes to the higher efficiency in this high luminosity range. The increase is shown in figures 5.21(a), 5.21(b) as function ot the impact parameter and p_T.

In fig. 5.22 the purity is shown as a function of track parameters while in fig. 5.23 we monitor it *vs* the barrel number, the wedge, the difference between the input and output barrels of the tracks and finally the instantaneous luminosity. For all these distributions we see an anologous behaviour of the two systems.

Overall the Gigafitter has the same purity levels of TF++, with a $\sim 1.5\%$ gain in efficiency.

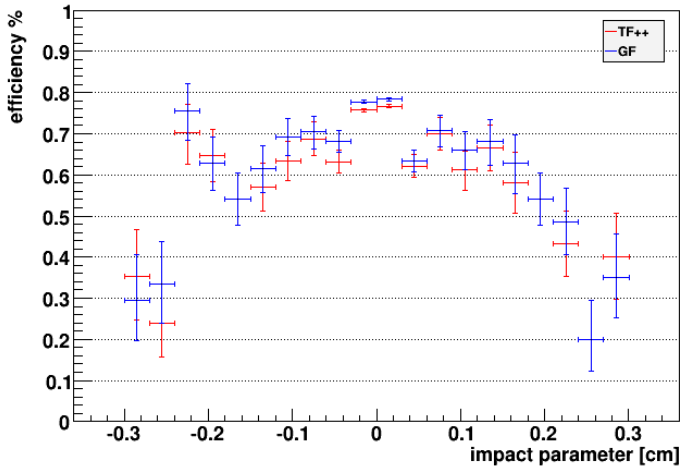


(a)

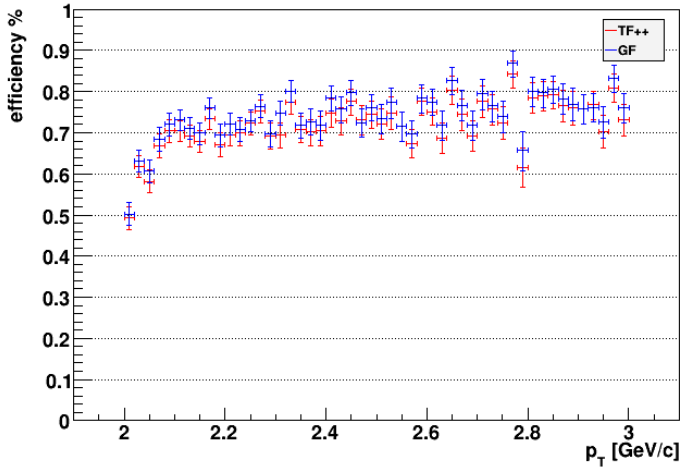


(b)

Figure 5.20: SVT (GF, TF++) Efficiency and fake rate at high luminosity - (a) Efficiency vs instantaneous luminosity: in red SVT with TF++ in blue SVT with GF. (b) Fake rate vs instantaneous luminosity with the same color code.



(a)



(b)

Figure 5.21: SVT (GF, TF++) Efficiency vs impact parameter and p_T -(a) Efficiency vs impact parameter: in red SVT with TF++ in blue SVT with GF. (b) Efficiency vs transverse momentum with the same color code, zoom in the region $2 < p_T < 3$ to highlight the turn on at low p_T.

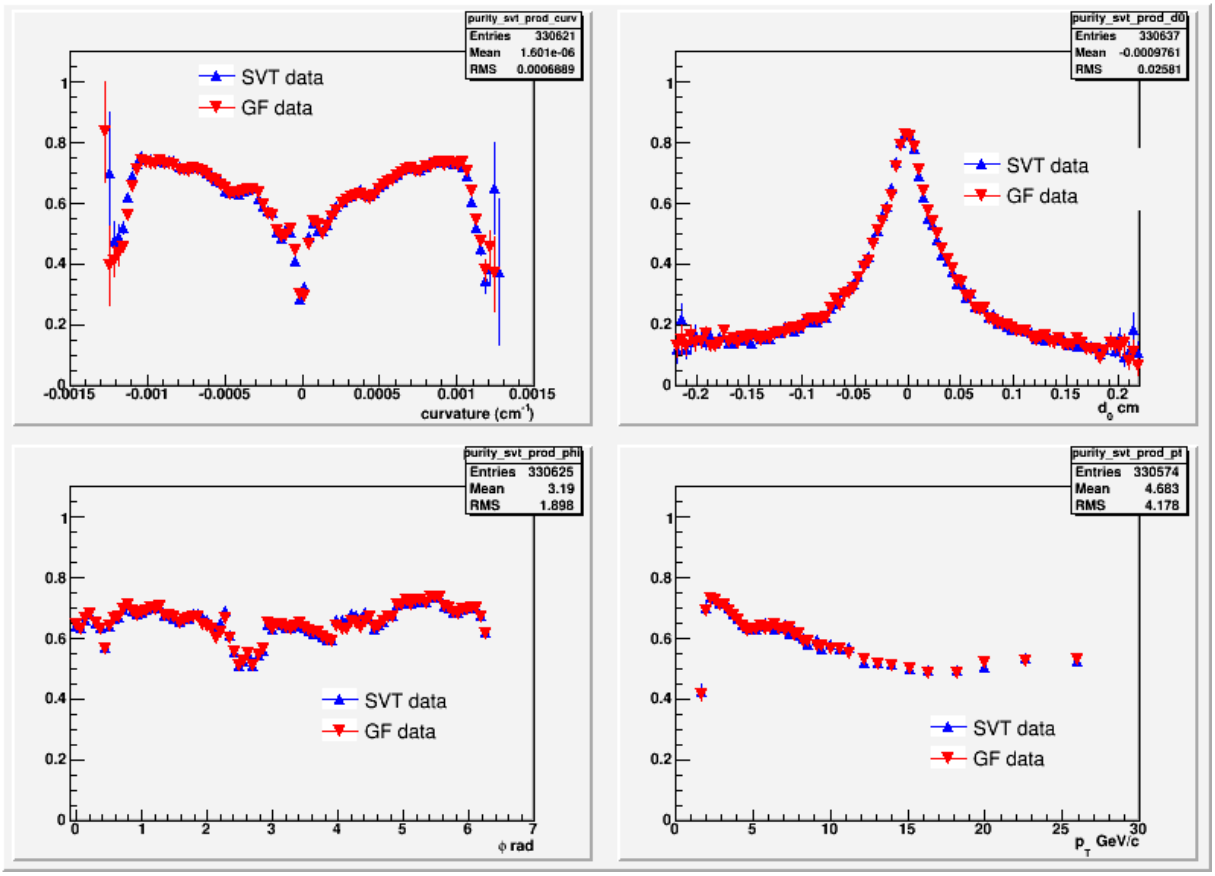


Figure 5.22: GF and TF++ purity as a function of curvature, d_0, ϕ and p_T

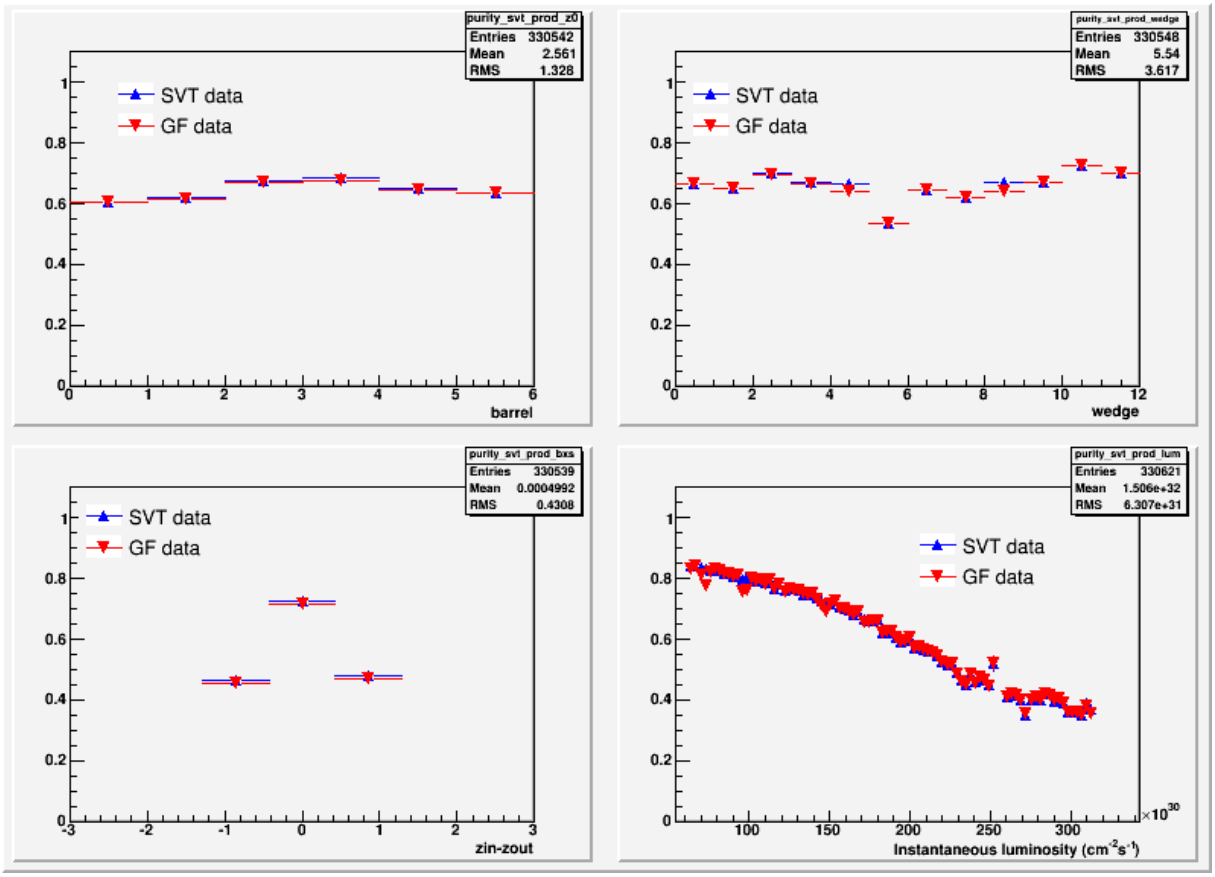


Figure 5.23: GF and TF++ purity as a function of barrel, wedge, difference between the input and output barrels of the tracks ($Z_{IN} - Z_{OUT}$) and instantaneous luminosity.

Chapter 6

Conclusions

The problem of the online selection of events (triggering) in hadron collider physics (chapter 1), has been described in particular at the CDF experiment (sec. 2.3) at the Tevatron collider in Fermilab (chapter 2).
It has been shown how the reconstruction of the trajectory of charged particles (tracking) is a critical task for the trigger and various examples of its usage on actual physics problems were provided (sec. 1.2).

Tracking is considered one of the hardest tasks for online selection: the amount of data sampled by the tracking detector is huge, the number of tracks to find large, but hidden over a much bigger combinatorial background. It has been described a sophisticated technique (SVT algorithm: sec. 3.2.2) to perform the track reconstruction task with performances comparable to the best offline algorithms, but executable by a dedicated processor fast enough for usage in the trigger system.

It has been shown in detail the hardware implementation of such algorithm for the CDF experiment: the SVT processor (chapter 3). Design, current performance and upgrade history has been described. A particular attention has been put in describing the flexibility of the SVT processor and how it was possible and necessary to upgrade the hardware in order to adapt it to the ever increasing Tevatron luminosity.

The first SVT upgrade was also a pioneer in the field of unplanned trigger hardware upgrades: it has shown how even a complex hardware trigger can be upgraded and commissioned during data taking using a phased plan if it was designed to be flexible enough. Thanks to the SVT upgrade experience it was possible to upgrade other parts of the CDF trigger and fully exploit the increased luminosity for physics measures.

6. CONCLUSIONS

The last SVT upgrade was In 2006 SVT was further upgraded to reduce its processing time. Without that upgrade the SVT would have been turned off.

The GigaFitter upgrade is a second generation upgrade and its main goal is to improve SVT efficiency and acceptance without loosing the SVT timing performance, during the final period of CDF data taking. It has been described the architecture (chapter 4) of the GigaFitter processor: a new generation single board processor for the track fitting stage of the SVT algorithm. It has been designed to replace the current 16-board TF++ processor in SVT and to provide the SVT with new and enhanced capabilities.

The GigaFitter board has been fully developed by a small group of physicists and engineers from Pisa and Padova. During my work thesis I have written a very large part of the firmware that implements the actual GigaFitter, contributed in developing the board and partecipated at the steps of the upgrade.
I have also analyzed effects and performance of SVT with the GigaFitter board in order to show how the SVT system can benefit from the new GF board.

At first, the Gigafitter is going to be installed with exactly the same constant and pattern sets of the current system, so we have checked its performance against the TF++ in terms of track parameter quality, resolution on parameters, purity and track reconstruction efficiency, and timing.
In the timing performances (sec. 5.4) it has been highlighted how the GF is able to deal with the most complex events much better than the TF++. The overall timing is not improved much with the current SVT tuning, but is foreseen how the use of the GF opens new possibility of SVT tuning that were forbidden by the lesser TF++ computing power (sec. 5.1).
The effect on the beam position calculation was also estimated (sec. 5.3). We found the GF assures the same resolution on track parameters as the current system and the same purity of the track sample, but with a gain of about 1.5% in track reconstruction efficiency (sec. 5.5), due to a different treatment of tracks having hits in all 5 SVXII layers. The differences in track parameter values are within the resolution as well as the differences in the beam position and slopes (sec. 5.2).

We can conclude that the GF upgrade effectively can enhance the aim of the SVT processor and enable its profitable usage at the new high luminosity of the Tevatron.
The GF board is currently commissioned at CDF. We are phasing out the old TF++ system.

The GF board is a processor with high speed, modularity and flexibility, and also shows how to design a new generation track fitter for this kind of algorithm, exploiting compact and powerful FPGAs with DSP processors. This experience will be essential for the future application of a SVT-like processor called FTK at the Atlas experiment at LHC.

List of Figures

List of Tables

Bibliography

[1] A. Abulencia *et al.*, "The CDF II extremely fast tracker upgrade," *Nucl. Instrum. Meth.*, pp. 358–360, 2007. 3, 33

[2] J. A. Adelman *et al.*, "The 'Road Warrior' for the CDF online silicon vertex tracker," *IEEE Trans. Nucl. Sci.*, vol. 53, pp. 648–652, 2006. 7

[3] J. Adelman *et al.*, "SVT Track Fitter Upgrade at CDF," 2005. CDF/DOC/TRIGGER/CDFR/7872. 7, 42, 56

[4] Fermilab, "Brief description and definitions of SVT Hit Buffers boards." URL `http://www-cdf.fnal.gov/internal/upgrades/daq_trig/trigger/svt/svtupgrade/hb_spec.ps`. 7

[5] B. Simoni, A. Annovi, R. Carosi, and S. Torre, "Patterns generation for the SVT upgrade," tech. rep., CDF Note CDF/DOC/TRIGGER/PUBLIC/8256, 2006. 7

[6] A. Abulencia *et al.*, "Observation of B^0 $(s) \to K^+K^-$ and Measurements of Branching Fractions of Charmless Two-body Decays of B^0 and B_s^0 Mesons in $\bar{p}p$ Collisions at $\sqrt{s} = 1.96$-TeV," *Phys. Rev. Lett.*, vol. 97, p. 211802, 2006, arXiv:hep-ex/0607021. 8

[7] A. Abulencia *et al.*, "Observation of B/s0 anti-B/s0 oscillations," *Phys. Rev. Lett.*, vol. 97, p. 242003, 2006, arXiv:hep-ex/0609040. 8

[8] B. Aubert *et al.*, "Improved Measurements of the Branching Fractions for $B^0 \to \pi^+\pi^-$ and $B^0 \to K^+\pi^-$, and a Search for $B^0 \to K^+K^-$," *Phys. Rev.*, vol. D75, p. 012008, 2007, arXiv:hep-ex/0608003. 8

[9] S. Donati, "CP Violation in the Bs system," vol. FPCP, 2007. 9

[10] S. Donati, "Una strategia per la misura della asimmetria CP nel decadimento B^0 $(s) \to hh$ a CDF," 1997. PhD thesis-Pisa University. 10

[11] D. Coll., "Measurement of the flavour oscillation frequency of B_s^0 mesons at $D\emptyset$." D0note 5474-conf. 11

[12] Fermilab, "Run II Handbook," 2003,
URL `http://www-ad.fnal.gov/runII/index.html`. 13

[13] C. Jensen *et al.*, "The CDF detector: an overview," *Proceedings of the 2001 Particle Accelerator Conference, Chicago*, 2001. 17

[14] R. Blair *et al.*, "The CDF-II detector: Technical design report," 1996. FERMILAB-PUB-96-390-E. 19

[15] F. Abe *et al.*, "The CDF detector: an overview," *Nucl. Instr. Meth.*, vol. A271, pp. 387–403, 1988. 19

[16] H. Wenzel, "Tracking in the SVX," CDF Internal Note 1790. 23

[17] N. Timothy, "The CDF L00 detector," 2001. CDF/PUB/SEC_VTX/PUBLIC/5780. 24

[18] A. Abulencia *et al.*, "The CDF II 3D Track Level 2 Trigger Upgrade," 2007. Presented at 15th IEEE Real Time Conference 2007 (RT 07),Batavia, Illinois, 29 Apr - 4 May 2007. 33

[19] A. Bardi *et al.*, "The CDF online silicon vertex tracker," *Nucl. Instrum. Meth.*, vol. A485, pp. 178–182, 2002. 34

[20] B. Ashmanskas *et al.*, "The CDF silicon vertex trigger," *Nucl. Instrum. Meth.*, vol. A518, pp. 532–536, 2004, arXiv:physics/0306169. 34

[21] A. Annovi *et al.*, "The AM++ board for the silicon vertex tracker upgrade at CDF," *IEEE Trans. Nucl. Sci.*, vol. 53, pp. 1726–1731, 2006. 35, 56

[22] M. Dell'Orso and L. Ristori, "VLSI structures for track finding," *Nucl. Instrum. Meth.*, vol. A278, pp. 436–440, 1989. 35

[23] I. Jolliffe, "Principal component analysis," *Springer Series in Statistics, 2nd ed.*, 2002. 36

[24] H. Wind, "Principal component analysis and its application to track finding," *Formulae and methods in experimental data evaluation. European Physical Society*, vol. III, 1984. 40

[25] K. Anikeev *et al.*, "CDF level 2 trigger upgrade," *IEEE Trans. Nucl. Sci.*, vol. 53, pp. 653–658, 2006. 42

[26] "Virtex-5 data sheets webpage," 2010,
URL http://www.xilinx.com/support/documentation/virtex-5_data_sheets.htm. 49, 57

[27] M. Bogdan *et al.*, "CDF level 2 trigger upgrade - the Pulsar project," *IEEE Nuclear Science Symposium Conference Record*, 2004. 55

[28] A. Bhatti *et al.*, "The CDF level 2 calorimetric trigger upgrade," *Nucl. Instrum. Meth.*, vol. A598, pp. 331–333, 2009. 55

[29] R. Pegna *et al.*, "A GHz sampling DAQ system for the MAGIC-II telescope," *Nucl. Instrum. Meth.*, vol. A572, pp. 382–384, 2007. 56

[30] "Altera apex20k overview."
URL http://www.altera.com/products/devices/apex/overview/apxoverview.html. 56

[31] "Leonardo spectrum."
URL http://www.mentor.com/synthesis. 56

[32] "Quartus ii."
URL http://www.altera.com/products/software/sfw-index.html. 56

[33] S. Amerio *et al.*, "The GigaFitter for fast track fitting based on FPGA DSP arrays," *IEEE Nuclear Science Symposium Conference Record*, vol. 3, p. 21152117, 2007. 62

[34] S. Amerio *et al.*, "The GigaFitter Upgrade," 2010. CDF/DOC/TRIGGER/CDFR/10030. 72

[35] "The BYPASS board."
URL http://www-cdf.fnal.gov/internal/upgrades/daq_trig/trigger/svt/BoardDocs/Bypass/index.html. 72

[36] "Spymon web page."
URL https://www-cdfonline.fnal.gov/internal/ops/svt/spymon/spymon.html. 73

[37] "Trigmon web page."
URL https://www-cdfonline.fnal.gov/internal/mon/consumer/trigmon/trigmon.html. 74

[38] R. Carosi *et al.*, "The SVT Monitoring," 2001. CDF/MEMO/TRIGGER/CDFR/5647. 74

[39] W. Badget *et al.*, "Description of the CDF Data Structure for Run II," pp. 181–186, 2008. CDF/DOC/CDF/PUBLIC/4152. 74