Dissertation
submitted to the
Combined Faculties for the Natural Sciences and for Mathematics
of the Ruperto-Carola University of Heidelberg, Germany
for the degree of
Doctor of Natural Sciences

Put forward by
Diplom-Physiker **Florin MACIUC**
Born in: **Craiova, Romania**
Oral examination: **20.04.2009**

Software Alignment
of the LHCb Inner Tracker Sensors

Referees: **Prof. Dr. Werner Hofmann**
Prof. Dr. Stephanie Hansmann-Menzemer

This thesis is dedicated to my wonderful parents and my entire family.

Abstract

This work uses the Millepede linear alignment method, which is essentially a χ^2 minimization algorithm, to determine simultaneously between 76 and 476 alignment parameters and several million track parameters. For the case of non-linear alignment models, Millepede is embedded in a Newton-Raphson iterative procedure. If needed, a more robust approach is provided by adding quasi-Newton steps that minimize the approximated χ^2 model function. The alignment apparatus is applied to locally align the LHCb's Inner Tracker sensors in an a priori fixed system of coordinate. An analytic measurement model was derived - as function of the track parameters and alignment parameters - for the two cases: null and non-null magnetic field. The alignment problem is equivalent to solving a linear system of equations, and usually a matrix inversion is required. In general, as consequence of global degrees of freedom or poorly constrained modes, the alignment matrix is singular or near-singular. The global degrees of freedom are obtained: directly from the χ^2 function invariant transformations, and, in parallel, by a diagonalization of the alignment matrix followed by an extraction of the least constrained modes. The procedure allows to properly define the local alignment of the Inner Tracker (IT). Using Monte Carlo data, the outlined procedure reconstructs the position of the IT sensors within micrometer precision or better. For rotations, an equivalent precision was obtained.

Die Zusammenfassung

Diese Arbeit verwendet die lineare Millepede-Alignierungsmethode, die im Wesentlichen ein χ^2 Minimierung Algorithmus ist, um gleichzeitig zwischen 76 und 476 Parameter und mehrere Millionen Spurparameter zu bestimmen. Für den Fall der nicht-linearen Alignierungsmodelle ist Millepede eingefügt in eine iterative Newton-Raphson-Methode. Falls erforderlich wird alternativ eine robustere Methode verwendet, die zusätzliche quasi-Newton-schnitte hinzufügt. Der Alignierungsprozedur wird für die lokale Alignierung von LHCb Inner Tracker Sensoren in einem "a priori" festgelegten Koordinatensystem verwendet. Ein analytischea Mess-Modell wurde abgeleitet als Funktion von Spur- und Alignierungsparameter für beide Fälle: mit und ohne Magnetfeld. Die Alignierungsprobleme sind äquivalent zur Lösung eines Systems von linearen Gleichungen, was normalerweise eine Matrixinversion erfordert. Im allgemeinen, als Folge der globalen Freiheitsgrade oder schlecht festgelegter Freiheitsgrade, ist die Alignierungsmatrix singulär oder fast singulär. Die globalen Freiheitsgrade werden direkt von invarianten Transformationen der χ^2-Funktion und simultan durch eine Diagonalisierung der Alignierungsmatrix, gefolgt von einer Extraktion der am wenigsten eingeschränkt Modi, bestimmt. Diese Methode erlaubt eine richtige Definition der lokalen Alignierung für den Inner Tracker. Die beschriebene Methode erlaubt für Monte-Carlo Daten die Bestimmung der Position der IT-Sensoren mit Mikrometer-Genauigkeit oder besser. Für Drehungen werden ähnliche Genauigkeiten erreicht.

Aut inveniam viam aut faciam

Hannibal Barca, crossing the Alps

Contents

List of Figures

List of Tables

Acknowledgments

I wish to convey my profound regards for the help provided by my parents, uncle, aunt and family.
I am deeply grateful for the support and help of my supervisor Michael Schmelling. I wish to especially acknowledge the help of “Max Plank Intitut für Kernphysik” particle physics group members: Markward Britsch, Bernhard Schwingenheuer, Karl-Tasso Knöpfle, Christian Bauer, Thomas Kihm and guest Professor Nikolai Gagunashvili. From Heidelberg University’s “Physikalisches Institut” LHCb group memebers: Johan Blouw, Marc Deissenroth and Manuel Schiller, I am in their debt for their help with LHCb software and more. I also want to mention the help of my friend Nigel Smale former member of the “Max Plank Intitut für Kernphysik” particle physics group. The list is by no means exhaustive, and I apologize to any person not included.

Chapter 1

B-physics, CP Violation and LHCb

1.1 Introduction to the Standard Model of Elementary Particles, Symmetries

1.1.1 Standard Model of Particle Physics

Fundamental Particles

A model of fundamental particles, i.e. non-composite particles, the Standard Model considers the matter composed of a small number of half-integer spin particles or fermions, 6 quarks and 6 leptons. These particles are given in the table 1.1 with the corresponding electrical charge (reference [1]).

Particle	F		lavor	$Q/\|e\|$
leptons	e	μ	τ	-1
	ν_e	ν_μ	ν_τ	0
quarks	u	c	t	$+2/3$
	d	s	b	$-1/3$

Table 1.1: Building blocks of matter, leptons and quarks with the associated charge

All composite particles known today, Mesons and Baryons are made of combination between 2 and 3 quarks, out from 6 quarks and 6 anti-quarks available. The leptons enter the composition of atoms. The protons are baryons and are made of three quarks uud, the neutrons are made of udd. The important and LHCb-relevant mesons B^0 and B_s^0 are made of $d\bar{b}$ and $s\bar{b}$ respectively.

Interactions

The fundamental interactions are given in quantum language by exchange of boson carriers. These integer-spin particles are the mediators of the fundamental forces between the fermions of table 1.1. There are 4 known types of fields or interactions.

The strong interaction are mediated by gluons, particles with null mass. They are responsible for the confinement of quarks into protons, neutrons, or any other hadron, i.e. baryons and mesons.

The electromagnetic interaction is responsible for binding the atoms in molecules and, for atom structure. This interaction is mediated by the massless photon.

The weak force is mediated by charged $W^\pm$ and neutral Z^0 bosons. This time the carriers of the weak force have a large mass, about 100 times the mass of a proton.

Gravity or the gravitational interaction, has graviton as the carrier boson.

Additionally, it is assumed it exists a boson field - the field responsible for the masses of elementary particles - the Higgs field.

Summarizing there are 12 fundamental fermion fields, 4 boson fields and the Higgs boson field. The Standard Model has in total 27 or more free parameters, i.e. parameters found from experiment and not as a fundamental consequence. There are:
1. 4 coupling constant, one for each boson field,
2. 15 masses of which 12 are lepton and quark masses, and the W, Z, Higgs masses respectively,
3. quarks are mixed and this adds 4 new parameters to the grand total,
4. 4 or more mixing parameters for the neutrino mixing.

The fact that there are so many parameters in the Standard Model seems to indicate that a more fundamental level of understanding is required.

1.1.2 Symmetries

The symmetries are crucial in physics. the Noether theorem states that for any differentiable symmetry exhibited by the action, i.e. integral of the Lagrangian function for a physical system, there will be a conserved quantity. This relates the fundamental continuous transformation of space, time, and rotation, to their corresponding conserved quantities momentum, energy and angular-momentum. E.g. with the energy conservation, the Hamiltonian of a physical system is the quantum infinitesimal time translation generator, or in other words $i\partial_t\psi(x,t) = \hat{H}\psi(x,t)$ the Schrodinger equation.

One of the most fundamental discrete symmetries are the charge C, the parity P, and the time inversion T. In the first half of the old century it was believed that all these symmetries are preserved by all known forces. Initially the physics community believed the mirror symmetry to be exact, it was however discovered by Yang and Lee in 1956, that there is no proof of Parity Conservation in the weak interaction [2]. Nowadays we know that this reaction maximally violates the Parity, as it was proven by Wu experiments in 1957, [3].

The CP conservation was disproved too in 1964, by James Cronin, Val Fitch and co-workers [4] while studying the neutral Kaon system. Now we consider only the product of all three discrete transformations, as exact: “CPT Theorem”. The origin of the CP violation is not known yet, but the present model of elementary particles, the Standard Model, includes the CP violation in Kaon domain by assigning a CP-violating phase in the transitions between fundamental quark flavors. The transition is mediated by the weak charge carriers. The mechanism shall be explained later in greater detail in section 1.2.

1.1.3 Cosmological Criteria

There is a known old requirement for the CP violation. We know that our universe is dominated by baryons. There is no known location in universe with a large presence of anti-matter. All known sources of anti-matter are matter-matter collisions or mater field interactions with particle anti-particle creation. The collisions of galaxies show no evidence of annihilation regions, figure 1.1 is a nice display of such events. This discrepancy between this huge asymmetry and the apparently almost exact symmetry be-

Figure 1.1: Collision between galaxies with no gamma ray emission form annihilation processes. Credit: HST/NASA/ESA.

tween particles and anti-particles as seen today demands an explanation. In 1966 Sakharov proposed, under assumption of an initial null baryon number for our universe, a model in which the baryon asymmetry could develop as the result of baryon number non-conservation interactions. He imposed the following set of conditions:
1. baryon number violating interactions;
2. thermal non-equilibrium;
3. CP-violation with C violation.
All this condition need to be fulfilled to explain the evolution from a non-baryonic universe to a baryon-antibaryon asymmetric universe.

1.2 Standard Model of particles and CP violation

Quark flavor is not preserved by the weak interaction, as the associated propagators of this force, the W bosons, change a quark of a certain flavor into a different flavored quark. The early parton model had mass-less quarks, however the unification of the weak and the electromagnetic force introduced the Higgs boson as responsible for the big mass difference between the heavy weak bosons W, Z, and the massless photon. The Spontaneous Symmetry Breaking mechanism is the reason behind the large mass differences.

In the Standard Model the quark mixing and the quark masses have a common origin, as the Yukawa interaction of the leptons and the quarks to the Higgs condensate is responsible for their mass, and in turn has as consequence the existence of a W interaction that converts a left-hand u-type quark into a left-hand d-type quark, see [5]. Without going into details how the quarks acquire the mass, I shall concentrate on the other aspect of quark mixing formalism.

1.2.1 Quark Mixing in Weak Interactions

In the case of a two-generation family quark system, the charged current Lagrangian takes the form of the early Cabibbo rotation (reference [6] and [7]):

$$\mathcal{L}_{CC} = \frac{g}{\sqrt{2}} \left(\bar{u}, \bar{c}\right)_L \gamma^\mu \begin{pmatrix} \cos\theta_c & \sin\theta_c \\ -\sin\theta_c & \cos\theta_c \end{pmatrix} \begin{pmatrix} d \\ s \end{pmatrix}_L W_\mu^+ + h.c. \tag{1.1}$$

here γ's are the Dirac matrices, the quark states are Left-handed, and g is the weak coupling constant.

In other words the quark mass eigenstates d and s were rotated to mixed states d' and s' by Cabibbo rotation and this states coupled to the W field to transform in the up-type quark states. These weak eigenstates, are within the Cabibbo-GIM – Glashow, Illiopoulos and Maiani came up with the charm 4th quark hypothesis – mechanism:

$$\begin{pmatrix} d' \\ s' \end{pmatrix} = \begin{pmatrix} \cos\theta_c & \sin\theta_c \\ -\sin\theta_c & \cos\theta_c \end{pmatrix} \begin{pmatrix} d \\ s \end{pmatrix} \tag{1.2}$$

Thus the W couples to the Cabibbo rotated state $(ud')^T$, in the same way as it does for a lepton pairs of a generation $(\nu_\mu \mu)$.

Now we know there are 3 generations of quark families. Hence the need of a new mixing matrix to allow the carrier W to couple to all 3 families, the Cabibbo-Kobayashi-Maskawa (CKM) matrix:

$$\begin{pmatrix} d' \\ s' \\ b' \end{pmatrix} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \begin{pmatrix} d \\ s \\ b \end{pmatrix} \tag{1.3}$$

with the corresponding charged current Lagrangian:

$$\mathcal{L}_{cc} = \frac{g}{\sqrt{2}} \left(\bar{u}, \bar{c}, \bar{d}\right)_L \gamma^\mu V_{CKM} \begin{pmatrix} d \\ s \\ b \end{pmatrix}_L W_\mu^+ + h.c. \tag{1.4}$$

For each of the nine parameters in the CKM matrix, there are in total 9 vertices that change a quark of one flavor to the on other, via emission or absorption of a W boson. The CKM matrix numbers are give in the Particle Data Group Review [8], without any potential phase factors:

$$|V_{CKM}| = \begin{pmatrix} 0.97419 \pm 0.00022 & 0.2257 \pm 0.0010 & 0.00359 \pm 0.00016 \\ 0.2256 \pm 0.0010 & 0.97334 \pm 0.00023 & 0.0415 \, {}^{+0.0010}_{-0.0011} \\ 0.00874 \, {}^{+0.00026}_{-0.00037} & 0.0407 \pm 0.0010 & 0.999133 \, {}^{+0.000044}_{-0.000043} \end{pmatrix} \tag{1.5}$$

This upper equation 1.5, shows the nowadays qualitative knowledge of the quark mixing. It is hoped by the physics community that the CKM will provide the next "smoking gun" of particle physics, in the sense it will offer evidence of "New Physics" beyond the actual Standard Model. It is not understood way the numbers in the matrix have these values, or whether there is some deeper layer uncovered till now.

One thing that these numbers would offer is a "apparently" straightforward check of the unitarity condition. As the CKM matrix is just a rotation in the base of the QCD eigenstates, the matrix must be unitary. This condition leads to 12 independent orthonormality relations, or conditions. To the 9 orthogonality relationships, 3 normalization conditions were added, and of these 12 relations, the six non-diagonal orthogonality relations are commonly used to study the implications of unitarity. If any one of these is observed to be broken, than there is evidence of New Physics and/or there is a 4th quark family.

1.2.2 B Meson Mixing

A direct consequence of the quark mixing by the weak charged current, is the particle anti-particle oscillation exhibited by some mesons. In the b-quark systems, to the $B^0\bar{B}^0$ oscillations, it was just added the $B_s^0\bar{B_s}^0$ oscillation observed by the CDF detector at TEVATRON [9].

I follow the usual oscillation phenomenology for a system of two oscillating states with non-null decaying amplitudes, see for example the reference [10].

The four mesons pairs $\mathrm{K}^0/\bar{\mathrm{K}}^0$, $\mathrm{D}^0/\bar{\mathrm{D}}^0$, $\mathrm{B}^0/\bar{\mathrm{B}}^0$, and $\mathrm{B}_\mathrm{s}^0/\bar{\mathrm{B}}_\mathrm{s}^0$ respectively, can be viewed as isolated systems, with a "vanishing" probability of decay. In this manner we exclude in this abridged formalism, the end-states of the decaying mesons, and we concentrate only on their oscillation phenomenology.

To begin we consider a non-relativistic Schrodinger equation, with:

$$i\partial_t \phi = H_{eff}\phi \tag{1.6}$$

Where the operator H_{eff} is a matrix responsible for the rotations in the space defined by the meson states, e.g. $\left|B^0\right\rangle$ and $\left|\bar{B}^0\right\rangle$. The flavor eigenstates $\left|B^0\right\rangle = \left|d\bar{b}\right\rangle$ and $\left|\bar{B}^0\right\rangle = \left|b\bar{d}\right\rangle$ are not eigenstates of the CP operator:

$$CP\left|B^0\right\rangle = e^{i\xi}\left|\bar{B}^0\right\rangle \quad \text{and} \quad CP\left|\bar{B}^0\right\rangle = e^{-i\xi}\left|B^0\right\rangle \tag{1.7}$$

where ξ is an arbitrary phase factor.

The mixed combination of flavor eigenstates $\left|B_+^0\right\rangle$ and $\left|B_-^0\right\rangle$ are constructed to be eigenstates of the charge-parity operator, by projection with $1 \pm CP$ operators:

$$\left|B_+^0\right\rangle = \frac{1}{\sqrt{2}}\left(\left|B^0\right\rangle + CP\left|B^0\right\rangle\right) \quad \text{and} \quad \left|B_-^0\right\rangle = \frac{1}{\sqrt{2}}\left(\left|B^0\right\rangle - CP\left|B^0\right\rangle\right) \tag{1.8}$$

$$CP\left|B_{+}^{0}\right\rangle = +\left|B_{+}^{0}\right\rangle \quad \text{and} \quad CP\left|B_{-}^{0}\right\rangle = \left|B_{-}^{0}\right\rangle \tag{1.9}$$

Unless the CP operator and the effective operator H_{eff} commute, the eigenstates $\left|B_{+}^{0}\right\rangle$ and $\left|\bar{B}_{-}^{0}\right\rangle$ can not be simultaneous eigenstates of CP and H_{eff}. As it will be shown later the CP is violated, and hence the mass eigenstates of the effective operator are not the eigenstates of CP operator.

A combination of the flavor eigenstates $a\left|B^{0}\right\rangle + b\left|\bar{B}^{0}\right\rangle$ is diagonalizing the effective operator in equation 1.6. With the time evolution given by the explicit form of 1.6:

$$i\frac{\partial}{\partial t}\begin{pmatrix} a(t) \\ b(t) \end{pmatrix} = H_{eff}\begin{pmatrix} a(t) \\ b(t) \end{pmatrix} \tag{1.10}$$

It is possible to choose, the effective operator as a sum between a mass operator and a decay operator, and this time the latter two operators are hermitian: $H_{eff} = \mathbf{M} - i\mathbf{\Gamma}/2$. Let us choose the "mass eigenstates" of the effective Hamiltonian as:

$$|B_L\rangle = p\left|B^{0}\right\rangle + q\left|\bar{B}^{0}\right\rangle \tag{1.11}$$

$$|B_H\rangle = p\left|B^{0}\right\rangle - q\left|\bar{B}^{0}\right\rangle \tag{1.12}$$

with H and L the heavy and the light component. The time dependence of the new states is:

$$|B_{L,H}\left(t\right)\rangle = \exp\left[-\left(im_{L,H} + \mathbf{\Gamma}_{L,H}/2\right)t\right]|B_{L,H}\left(0\right)\rangle \tag{1.13}$$

Defining the evolution factor $f_{\pm}(t) = \left(e^{-im_H t}e^{-\Gamma_H t/2} \pm e^{-im_L t}e^{-\Gamma_L t/2}\right)/2$, the oscillation time dependence could be written in a very compact form, (see reference [11]):
1. For the B^0 meson,

$$P\left(B^{0} \to B^{0}, t\right) = \left|f_{+}(t)\right|^{2} \tag{1.14}$$

$$P\left(B^{0} \to \bar{B}^{0}, t\right) = \left|\frac{q}{p}\right|^{2}\left|f_{-}(t)\right|^{2} \tag{1.15}$$

2. For the $\bar{B}^0$ meson,

$$P\left(\bar{B}^{0} \to \bar{B}^{0}, t\right) = \left|f_{+}(t)\right|^{2} \tag{1.16}$$

$$P\left(\bar{B}^{0} \to B^{0}, t\right) = \left|\frac{p}{q}\right|^{2}\left|f_{-}(t)\right|^{2} \tag{1.17}$$

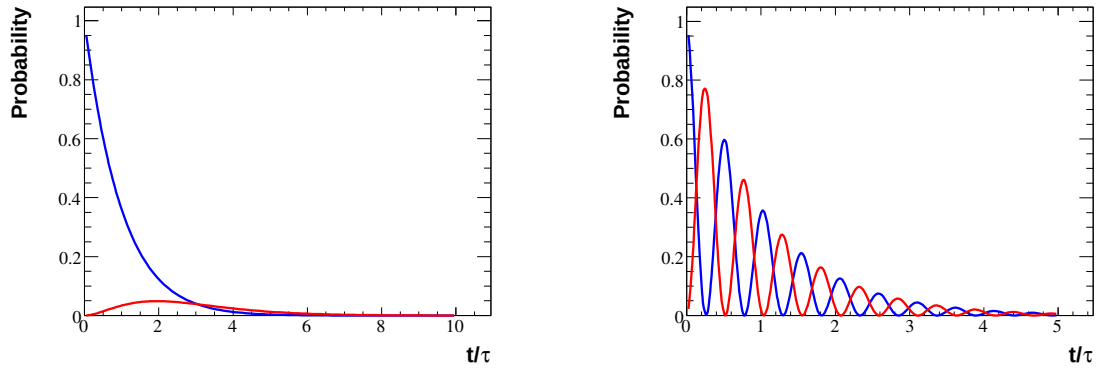
The evolution of probability for a $B^0 \to \bar{B}^0$ is plotted in 1.2 for a B^0 and a B_s^0 meson.

The oscillation frequency is given by the mass difference Δm which for a B^0 is 0.5 ps^{-1} or 0.33 milli-eV, and for B_s^0 is of about 17.8 ps^{-1} or 12 milli-eV. This difference in mass value for B_s^0 with respect to B_0 explains why in the B_s^0 the experiment usually uses the Fourier Transform methods and not a curve fit to extract the mass difference value.

At the level of mixing a CP violation might appear from the difference in the oscillation amplitude:

$$P\left(B^{0} \to \bar{B}^{0}, t\right) \neq P\left(\bar{B}^{0} \to B^{0}, t\right) \tag{1.18}$$

The oscillation of a B to anti-B does not have the same amplitude with the inverse process. Which means in turn that there is a difference of weight between flavor contents in the mass eigenstate of the effective Hamiltonian $|q| \neq |p|$.



(a) $B^0 \to B^0$ blue and $B^0 \to \bar{B}^0$ red

(b) $B_s^0 \to B_s^0$ blue and $B_s^0 \to \bar{B_s}^0$ red

Figure 1.2: Oscillation probabilities in B^0 and B_s^0

1.2.3 CP Violation in the B Sector

In the last subsection it was mentioned that when $\left|\frac{p}{q}\right| \neq 1$ the CP is not conserved, and such asymmetry between flavor states would lead to a decay final state which has not necessarily the same CP value as the very initial B^0 or $\bar{B}^0$ state.

The first observation of the CP violation was observed in neutral Kaon systems, where at very long times compared with the short lived particle life time, it was observed a decay-final state with the CP equal to the CP of the short-lived kaon, see [4]. This violation of the CP state is an indirect CP violation, as the example just discussed for the B-systems. The indirect CP violation is a non-conservation of CP value in mixing between the meson and its own anti-particle.

There are additional two types of CP violations, violations at level of decay directly and at level of interference between decay amplitudes. If we define the amplitudes of decay for each combination of final states and mesons with their CP conjugates:

$$A_f = \langle f | H_{eff} | B^0 \rangle \tag{1.19}$$
$$A_{\bar{f}} = \langle \bar{f} | H_{eff} | B^0 \rangle \tag{1.20}$$
$$\bar{A}_f = \langle f | H_{eff} | \bar{B}^0 \rangle \tag{1.21}$$
$$\bar{A}_{\bar{f}} = \langle \bar{f} | H_{eff} | \bar{B}^0 \rangle \tag{1.22}$$
$$\tag{1.23}$$

Then the 3 possible types of CP violation are:
1. direct, when decay probability do not match:

$$\left|\frac{A_f}{\bar{A}_{\bar{f}}}\right| \neq 1 \tag{1.24}$$

2. in mixing, when:

$$\left|\frac{q}{p}\right| \neq 1 \tag{1.25}$$

3. in interference between amplitudes of decay and oscillation [12]:

$$\Im\left(\frac{q}{p}\frac{\bar{A}_f}{A_f}\right) \equiv \Im\lambda_f \neq 0 \tag{1.26}$$

1.3 CP and LHCb

CP-violation through Interference between Decay and Mixing

Taking the third example of CP violation, a direct physical process that embodies the interference between mixing and decay is the $B^0/\bar{B}^0 \to J/\psi K_S$ transitions. In this process the weight fraction $\left|\frac{q}{p}\right| = 1$ and the only contribution to CP violation comes from the interference term.

The Standard Model prediction for the interference term is $\arg(\lambda_{\mathrm{f}}) = -2\beta$. The latter β angle being the argument of the V_{td} element in CKM matrix. Hence the inclusion of a phase factor in the CKM td element has made possible a CP violation.

$$V_{td} \approx |V_{td}|\, e^{-i\beta} \tag{1.27}$$

The $B^0/\bar{B}^0 \to J/\psi K_S$ channel study is one of the main goals at LHCb. If the upper relation would be proven incorrect with some small non-null contribution, this would indicate the action of New Physics, violating one of the unitarity relations of the CKM matrix.

In the interference type of CP violation, the CP violation can be expressed quantitatively as the CP asymmetry fraction in terms of transitions rates:

$$A_{CP} = \frac{\Gamma\left(B^0 \to f\right) - \Gamma\left(\bar{B}^0 \to f\right)}{\Gamma\left(B^0 \to f\right) + \Gamma\left(\bar{B}^0 \to f\right)} \tag{1.28}$$

where f final state is a CP-eigenstate.

This new expression can be converted to a form in terms of phase-factors ω, Δm, ϕ (see [13]). Here the phase factors are in order:
1. 2ω the relative phase between the transition amplitude for $B^0 \to f_{CP}$ and $\bar{B}^0 \to f_{CP}$;
2. Δm is the mass difference between the heavy mass B_H and B_L, or the oscillation frequency for $B^0 \to \bar{B}^0$ and $\bar{B}^0 \to B^0$;
3. The 2ϕ is an oscillation relative phase factor between $B^0 \to \bar{B}^0$ and $\bar{B}^0 \to B^0$, or in terms of the precedent notation:

$$\phi - 2\omega = -\arg(\lambda_{\mathrm{f}}) = -\arg\left(\frac{\mathrm{q}}{\mathrm{p}}\frac{\bar{\mathrm{A}}_{\mathrm{f}}}{\mathrm{A}_{\mathrm{f}}}\right)$$

with these definition the oscillation and decay interference-induced CP is:

$$A_{CP} \approx -sin\left(\Delta m t\right)\sin\left(\phi - 2\omega\right) \tag{1.29}$$

The equivalent notation can be preserved for B_s^0 with $\phi \to \phi_s$, $\Delta m \to \Delta m_s$ and the same for ω. This was considered a "pure" case, without CP-violations of other types.

Proof:
Assuming there is no CP violation in mixing or decay,

$$\left|\frac{A_f}{\bar{A}_{\bar{f}}}\right| = 1 \quad \text{and} \quad \left|\frac{q}{p}\right| = 1$$

it is possible to write the transition rates $\Gamma\left(B^0 \to f\right)$ and $\Gamma\left(\bar{B}^0 \to f\right)$ in terms of A_f, $\bar{A}_f$, q, p and $\Delta\Gamma = \Gamma_L - \Gamma_D$ ([14]):

$$\Gamma\left(B^0 \to f\right) = \frac{|A_f|^2}{2} e^{-\Gamma t} \left|I_+(t) + I_-(t)\right|$$
$$\Gamma\left(\bar{B}^0 \to f\right) = \frac{|A_f|^2}{2} e^{-\Gamma t} \left|I_+(t) - I_-(t)\right|$$

where:

$$I_+(t) = \left(1 + |\lambda_f|^2\right) \cosh\left(\Delta\Gamma t/2\right) - 2\Re\left(\lambda_f\right) \sinh\left(\Delta\Gamma t/2\right) \tag{1.30}$$
$$I_-(t) = \left(1 - |\lambda_f|^2\right) \cos\left(\Delta m\, t\right) - 2\Im\left(\lambda_f\right) \sin\left(\Delta m\, t\right) \tag{1.31}$$

With λ_f just a phase term, $A_f = A_0 e^{i\omega}$, $\bar{A}_f = A_0 e^{-i\omega}$, and with $\Gamma_L \approx \Gamma_H$, the upper relations when included in the CP asymmetry factor definition:

$$A_{CP} = \Im\left(\lambda_f\right) \sin\left(\Delta m\, t\right) \tag{1.32}$$

In Standard Model the upper quantity is: $A_{CP} = \sin\left(2\beta\right) \sin\left(\Delta m\, t\right)$, with β an angle defined by one of the unitarity relations of CKM matrix.

CKM Parameterization, Standard Model Expectations for Interference-induced CP-violation and LHCb Possible Measurements

The CKM matrix can be parametrized in the so called Wolfenstein parametrization, based on 4 parameters λ, A, ρ, and η – carefully as the λ has nothing in common with the λ_f. With λ parameter small, the parametrization up to a power of 4 in λ is:

$$V_{CKM} \approx \begin{pmatrix} 1 - \lambda^2/2 & \lambda & A\lambda^3(\rho - i\eta) \\ -\lambda & 1 - \lambda^2/2 & A\lambda^2 \\ A\lambda^3(1 - \rho - i\eta) & A\lambda^2 & 1 \end{pmatrix} \tag{1.33}$$

In general LHCb shall look for small deviation from the following CKM unitarity relations (reference in [14]):

$$V_{ud}V_{ub}^* + V_{cd}V_{cb}^* + V_{td}V_{tb}^* = 0 \tag{1.34}$$
$$V_{tb}V_{ub}^* + V_{ts}V_{us}^* + V_{td}V_{ud}^* = 0 \tag{1.35}$$

The first equation 1.34 could be represented as the triangle shown in figure 1.3, and the unitarity condition can be replaced by the condition that the triangle angles should add to 180^0 - any deviation from this relation would mean there is something wrong with the Standard Model.

To the first approximation in λ the triangle 1.3 has angles given in terms of the CKM matrix elements:

$$\arg\left(\mathrm{V_{ub}}\right) = -\gamma \tag{1.36}$$
$$\arg\left(\mathrm{V_{td}}\right) = -\beta \tag{1.37}$$

The inclusion of the two phases in CKM matrix generate distinct mixing between the quarks and the anti-quarks, and this is a reason for the CP violation.

The Standard Model box diagrams are shown in 1.4 for both B and B_s mixing. These two processes give a large mixing phase for B_d oscillation and a small phase for B_s. The mixing phases ϕ and ϕ_s are ([13]):

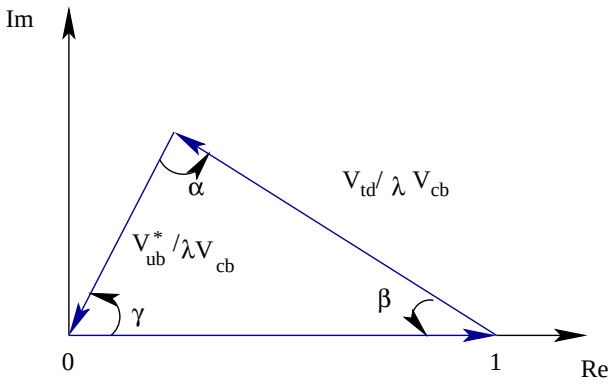
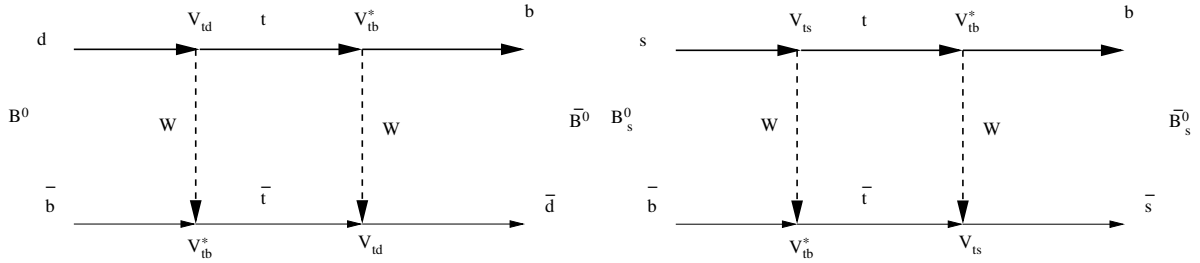


Figure 1.3: One unitarity triangle corresponding to identity 1.34



(a) $B^0\bar{B}^0$ oscillation (b) $B_s^0\bar{B}_s^0$ oscillation

Figure 1.4: Box diagrams of Standard Model for a B^0 to $\bar{B}^0$ oscillation

$$\phi = 2\mathrm{arg}\,(\mathrm{V_{td}}) = -2\beta \tag{1.38}$$
$$\phi_s = 2\mathrm{arg}\,(\mathrm{V_{ts}}) \approx 0 \tag{1.39}$$

For the mixing frequency the opposite is true:

$$\Delta m \sim |V_{td}|^2 \propto \lambda^3 \quad \text{small} \tag{1.40}$$
$$\Delta m_s \sim |V_{ts}|^2 \propto \lambda^2 \quad \text{large} \tag{1.41}$$

The oscillation phases of the processes 1.4, and the oscillation frequency are prime targets in the LHCb program. A possible way to extract the ϕ_s mixing phase is from the "golden" channel $B_s \rightarrow J/\psi\phi$. New physics could contribute with additional mixing phases, and thus is imperative to measure the known phases as good as possible. The "golden" channel gives experimentally a very clean signal due to the presence in the final state of 2 muons as decay products of J/ψ.

The CP asymmetry in $B_s \rightarrow J/\psi\phi$: $A_{CP} = -\sin(\Delta m_s t)\sin(\phi_s - 2\omega) \approx -0.04\sin(\Delta m_s t)$ (1.42)

as the Standard Model prediction for ϕ_s and ω are 0.04 and 0. This is a fast oscillating signal and requires Fourier analysis tools or very precise vertex resolution.

The next two body decay processes would be good candidates to measure the α and γ angles of the first unitarity relation 1.34, and the second unitarity relation 1.35, respectively. for :

1. $B_d \to \pi^+\pi^-$, with asymmetry ratio:

$$A_{CP} = \sin(\Delta m\, t)\sin(2\alpha)$$

and the phase $\omega = \gamma$,
2. $B_s \leftarrow K^+K^-$, with asymmetry ratio:

$$A_{CP} = \sin(\Delta m_s\, t)\sin(2\gamma)$$

and the phase $\omega = \gamma$, too. Here $\gamma = -\arg(\mathrm{V_{ub}})$. The uncertainty in the signal for the two body decay is dominated by large theoretical uncertainties, as the final kaons and pions are superimposed on a huge background.

Other tests could be performed at LHCb ([13]):

1.CP-asymmetry ratio measurement of $B_d \to \phi K_s$. The mode has in the final state a CP-eigenstate. The Standard Model gives:

$$A_{CP}(t) = -sin(\Delta m_s)\sin(\phi_d - 2\omega)$$

with very small decay phase $\omega_{SM} \approx 0$. A enhanced decay phase would point to some new CKM-couplings.

2. $B_s \to \phi\phi$ with very small CP asymmetry, in this reaction the Standard Model estimates favor a small decay phase of the B_s system and the small mixing phase. The reaction has a branching ratio of $(1.4 \pm 0.8)\,10^{-5}$.

3. The unobserved till now $B_s \to \mu^+\mu^-$. This mode is sensitive to Super Symmetric (SUSY) contribution, as its Standard Model estimate is very low.

4. An analog example is the decay $B_d \to K^*\mu^+\mu^-$.

The list is by no means exhaustive, and a "smoking gun", could be found in any of the B^0 or B_s^0 decay modes.

1.4 Conclusion to CP Analysis at LHCb

As the LHC shall become the largest B factory ever, the prospect for CP-violation studies and New Physics at LHCb is very good. Already discovered processes will be measured and new levels of precision will be attained. This would push farther the boundaries or Standard Model applicability range, or alternatively would signal some discrepancy between the Model and the measurements.

Evidence of new physics could be obtained from the old data in conjunction with the LHCb data. As well possible would be completely new discoveries: of some exotic decay modes or enhancement of rare processes due to Super Symmetric partners. SUSY could be just around the corner.

More mundane reaction would be measured, e.g. in Minimum Bias data. In this way valuable cross-section and ratio values would be computed, subsequently providing input for fine-tuning the software tools, e.g. Particle generator, and the theoretical models. As a key example on the last point, ALICE needs to extrapolate the physics from proton-proton collisions to the Lead-Lead collisions. Otherwise it would be impossible to estimate the collective effects present in a Quark Gluon Plasma.

Chapter 2

LHCb Detector

2.1 CERN's Large Hadron Collider Accelerator

The Large Hadron Collider (LHC) is a two ring superconducting accelerator and collider. It is build under-ground at the border between France and Switzerland, in the region of Geneva. As seen in figure 2.1, with one edge the ring almost touches the Léman lake, and on the other borders on the Jura mountain peaks. Build in the LEP tunnel, the accelerator has a circumference length of 26.7 km. Under a mollase overburden, and at depths of 170 m and 45 m underground, the ring accelerates particles close to light speeds. Protons and Lead nuclei will circle the ring circumference and reach momenta of 7 TeV per proton and 2.8 TeV per nucleon (Lead).

Figure 2.1: Aerial view of CERN's LHC with SPS

To put the numbers in perspective the protons will reach a speed of about 10 km/h lower than the photon speed. In relativistic terms the speed of the proton is 0.999999991 c.

LHC uses dipole magnets to keep the protons on a circular trajectory inside the ring and quadrupole magnets to focus the beam. The top value of the magnetic field created by the dipoles is 8.3 Tesla. The field accelerates the initial 450 GeV proton injected by the Super Proton Synchrotron (SPS), to 7 TeV energy. The accelerator quadrupoles are arranged to focus optimally the two beams to common collision points for 4 nodal points, where 4 detectors are placed.

On the ring there are more than 1.200 dipole magnets and about 400 quadrupole magnets. In total, there are 1600 superconducting magnets with 27 tonnes average mass per each, operating at 1.9 K. To keep the superconducting magnets at this extreme temperature, about 96 tonnes of liquid helium are circled through-out the ring, cooling the magnets which are bending the high energy particle beam.

Figures 2.2 show a schematic view of the 15 m long dipole cylinder. Outlined among others are the

dipole coils around the beam pipe, the quadrupole bars, the isolation, thermal shielding, radiation screen to protect from synchrotron emission, iron yoke for the magnet, etc.

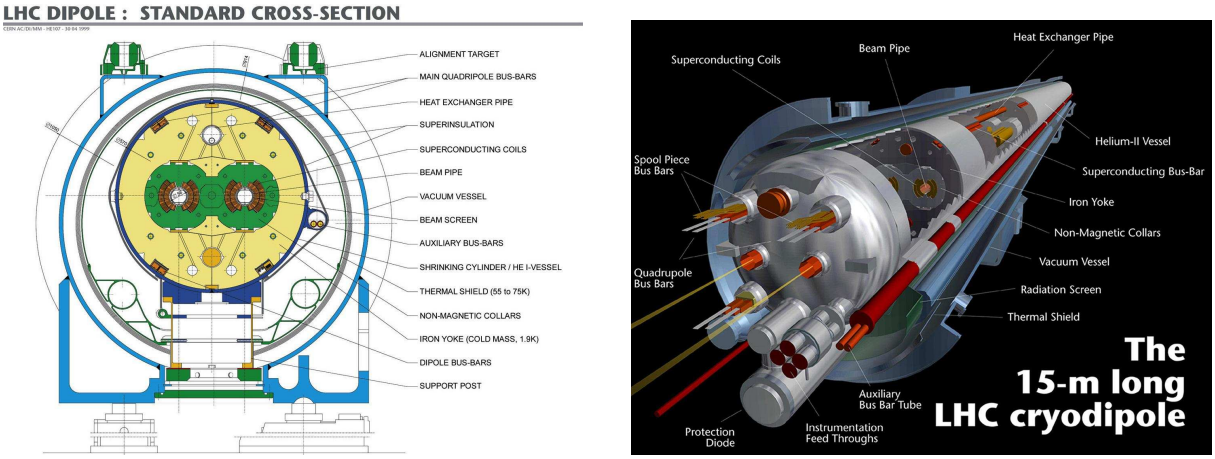


Figure 2.2: A 15 m long LHC dipole with cross-section and length views,

The accelerator functions as the largest existent cryostat. The magnets coils and the cooling pipes need to be isolated from the beam, as well as from the outside. All the electronics and the cooling systems together with the superconductors need to be perfectly shielded, in an event of: a beakdown, a beam dump, and other catastrophic event.

A total 0.584 A beam current corresponds to a beam stored energy of approximately 362 MJ, see [15]. To the value given it must be added the energy stored in the magnets of about 600 MJ, which makes the total close to 1 GJ. This gargantuan energy quantity must be siphoned away safely in the case of a malfunction or an emergency.

At nominal operating parameters the LHC ring should fill 2808 equally spaced bunches. With the speed of proton close to that of light, the average time needed for a circumference run would be of 89 micro-seconds, and the collisions shall not happen at a rate lower than 1 per 25 nano-seconds.

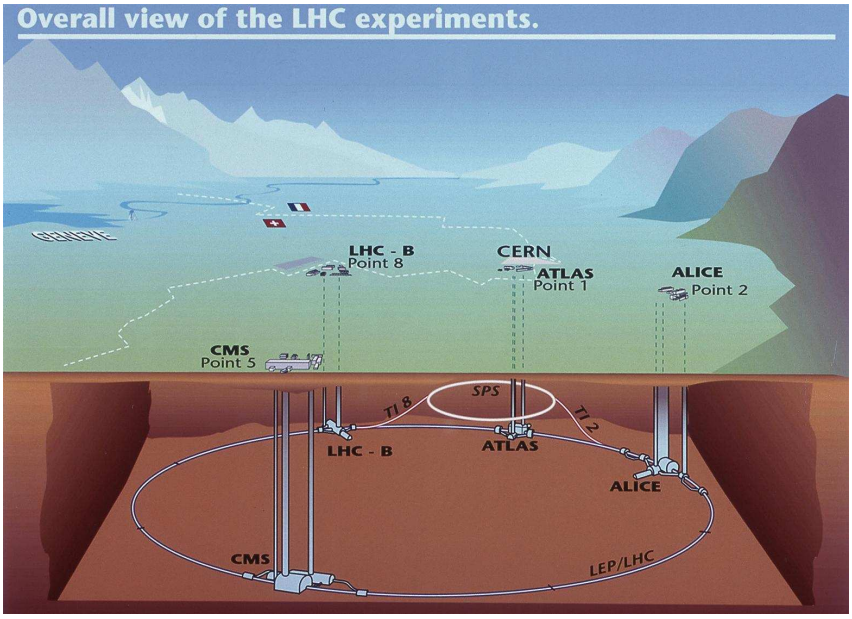


Figure 2.3: Schematic view of the LHC under-ground structure

In figure 2.3 it is displayed a schematic view of the LHC underground ring, and the caverns of 4

detectors: A Toroidal LHC ApparatuS (ATLAS), Compact Muon Solenoid (CMS), A Large Ion Collider Experiment (ALICE), and Large Hadron Collider beauty (LHCb). Each detector gets a certain number of collision events. The decisive quantity is the beam luminosity at the nodal point of each detector. The number of certain type of events per second is directly proportional to the luminosity and to the cross-section of the specified physical process.

$$N_{event} = L\sigma_{event} \tag{2.1}$$

In future, the big experiments ATLAS and CMS are expected to get a peak luminosity of about $10^{34} cm^{-2} s^{-1}$ and the low luminosity LHCb experiment aims for a peak luminosity of $10^{32} cm^{-2} s^{-1}$. The first two experiments are primary concentrated on finding the Higgs boson, and the LHCb physics program is concentrated on the CP violation in the B-sector. This detector is the subject of the next sections.

2.2 LHCb a Single Arm Spectrometer

The dominant mechanism in $b\bar{b}$ production is the gluon-gluon scattering in which two virtual gluons fusion to form a quark anti-quark pair. Given the distribution of gluon energy and momentum inside the proton, in most cases the collision happens between a virtual gluon that carries a significant fraction of proton momentum and a gluon with much less momentum. The produced $b\bar{b}$ system is strongly boosted forward in the beam direction. The hadronization leads to B and $\bar{\mathrm{B}}$ mesons with a conical very narrow angular distribution.

The single arm spectrometer takes advantage of the angular distribution of the produced particles to measure and tag the characteristics of a B event. The single arm spectrometer's structure, means that it can access regions of phase-space which are unreachable for the barrel type detectors as ATLAS and CMS.

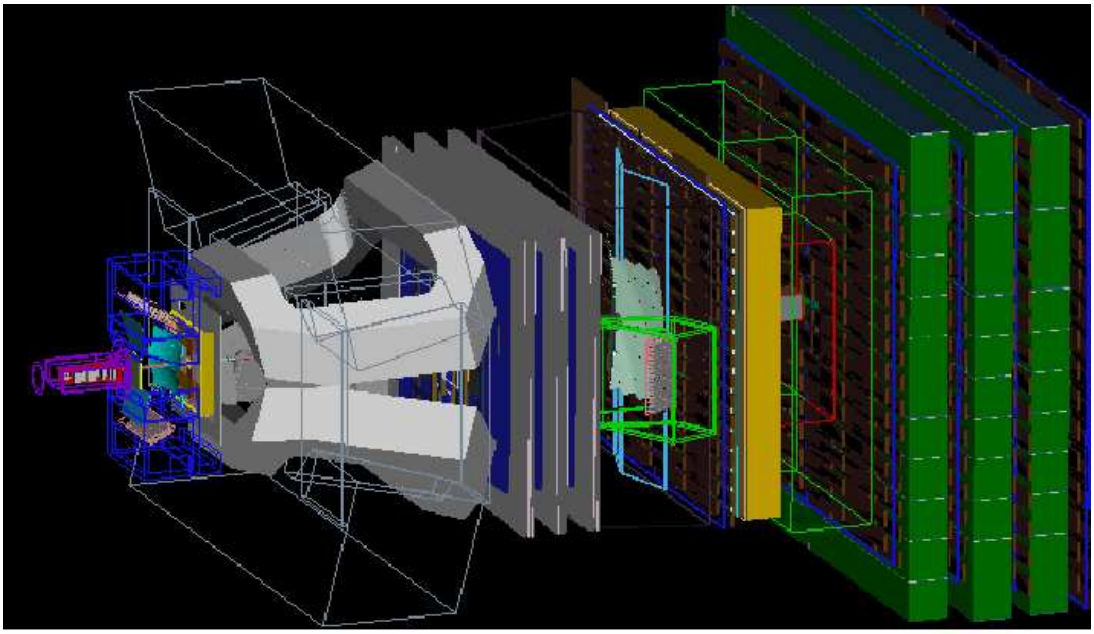


Figure 2.4: Panoramix detector display of the entire LHCb sub-detector setup

LHCb (see reference [16]) is a single-arm spectrometer with an angular geometrical acceptance from approximately 10 mrad to 300 (250) mrad in the bending (non-bending) plane. It was build in the old Delphi Cavern of the LEP era. Important later for alignment purposes, the LHCb system of coordinates in figure 2.5 is defined by: a vertical y axis with the y direction towards the surface, a z direction parallel to

the floor, and tilted with respect to the beam at 0.0036 radians, a right-hand coordinate system convention gives the final axis x.

The LHCb detector is shown in 2.4 or figure 2.5. The structure is split into subdetectors in order from the Interaction Point (IP) downstream:
1. Vertex Locator (VELO),
2. the first Ring Imaging Cherenkov detector (RICH1),
3 the first silicon tracker "Tracker Turicensis" (TT),
4. the magnet yoke,
5. three tracking station with a silicon Inner Tracker (IT) and a straw Outer Tracker (OT),
6. the second Ring Imaging Cherenkov detector detector (RICH2),
7. the first set of muon detector chambers of pads (M1),
8. Scintillator-Pad and Preshower Detector (SPD:PS),
9. Electromagnetic Calorimeter (ECAL),
10. Hadronic Calorimeter (HCAL),
11. 4 other Muon stations (M2-M5).

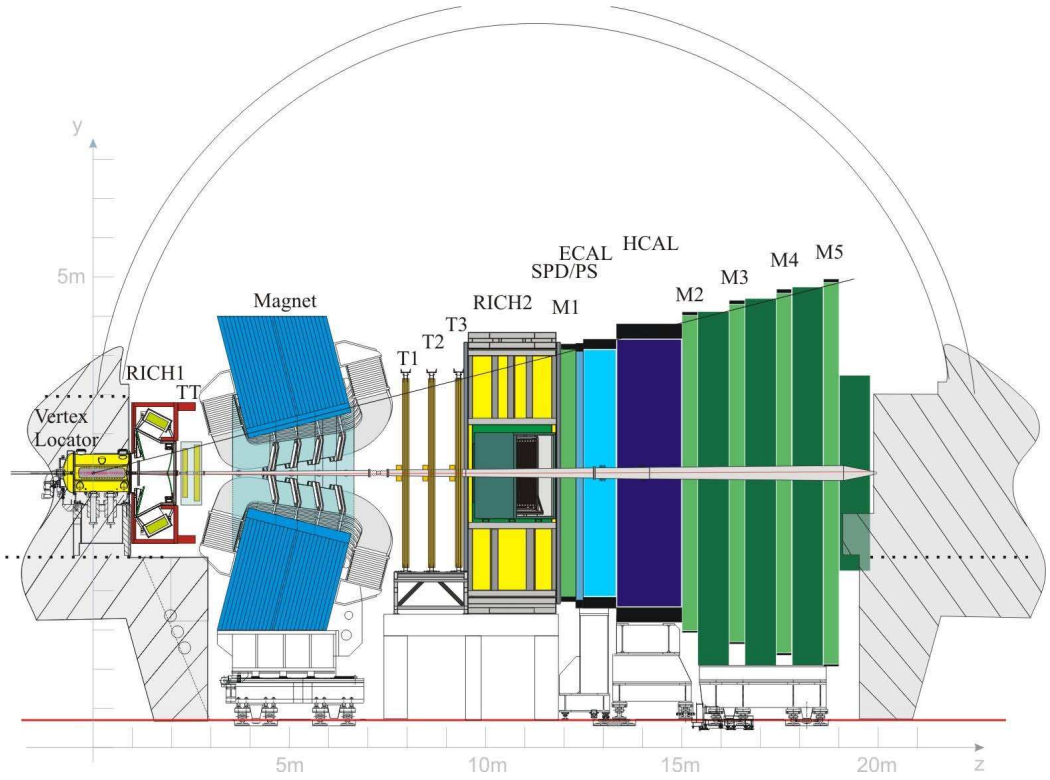


Figure 2.5: LHCb sub-detector setup

The particles are first detected in VELO, a vertex detector based on silicon strip detectors with low pitch. The interaction point is situated at about one third of VELO's length. This subdetectors also take care that a pile-up event with many interactions is flagged.

The particles then reach the RICH1, where for the slow component of the event the particle identity is found. The silicon tracker TT detects the particles, just at the edge of the magnetic field, and can provide in conjunction with the VELO track state the first momentum estimates. The particle momentum is extracted from the curvature in small magnetic field.

The Magnet is a warm magnet design with saddle-shaped coils. The yokes can be seen in the figure 2.4, they are window shaped saddles with sloping poles. The design allows a 4 Tm integrated magnetic field for a 10 m track. The shape chosen insures maximal field intensities in the region between TT and OT-IT, together with an almost null filed at the level of RICH detectors. The total weight of the yoke is

1.5 ktons and the coils add a 54 tons to this. A field mapping was done with Hall probes, and a precise magnetic filed map was obtained.

Important for alignment the magnetic field inside IT has a strong gradient in z. There are also fluctuations in the x-y plane even within the relative small area of the IT.

After emerging from the strong magnetic field region, the particles are detected by the Inner and Outer Trackers. The first is a silicon strip detector, and the second is a drift-time detector. The IT silicon detector has high detection efficiency and small pitch allowing precise measurements. The OT on the other hand has a large acceptance with relatively good position determination, leading to precise momentum estimates for the lower end of the momentum spectrum. The combination between the track states in OT-IT with the track states in the "before magnet" upstream region of the VELO and TT, provide a global tracking fit over all tracking stations. The result of the global fit is a very exact momentum determination for the particle close to the interaction point.

Before stopping in the calorimeters the hadronic and low mass leptonic components are detected by RICH2 Cherenkov detector. The identity of high energy particles is determined.

With its high thickness in radiation lengths, the ECAL calorimeter stops the photons, the electrons, and the positrons. It also determines the energy for these particles. The energy estimate is not as precise as the tracking, but on the other hand ECAL measures the direct photons and the neutral pion decay photons, which make up for a third of the energy flow through the detector

Having filtered away the low mass leptonic part, the HCAL detects and stops the hadronic component. The HCAL clusters and ECAL clusters provide valuable information for the Trigger system.

The most downstream detectors are 4 of the 5 Muon detector station. The Muon detector finds the least interacting charged particles, the muons. These are the weak-component.

2.2.1 The Vertex Locator VELO

VELO uses silicon strip technology. As a ionizing particle crosses the bulk of the silicon sensor, it produces the usual collection of electron-hole pairs. The charges are collected by the strips and a triggering signal is produced via the channels.

The VELO is partitioned in 2 halves, to allow the detector to be in a open position, with the sensors 30 mm apart during injection and 7mm apart during normal running. One VELO half consists of 21 silicon modules with double-sensors. In total there are 42 sensors r-sensors and 42 ϕ-sensors in both halves. Each module measures both transverse x-y coordinates by combining the cylindrical coordinates r and ϕ of each sensor.

300 micrometer thick aluminum foil covers the beam and prevents contamination from the denser VELO's vacuum. There are 2 additional modules per each half for Pile-Up VETO, and only r-sensors are in the latter modules.

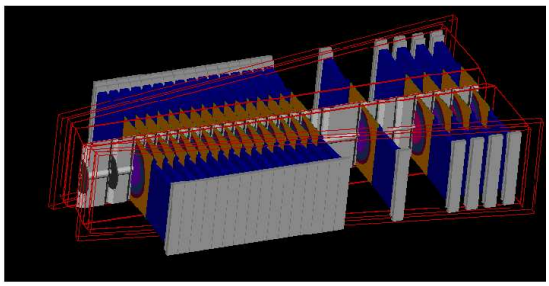
The sensors are split into an inner region closer to the beam and an outer region. The inner region has smaller pitch between strips. The density of strips increases in the inner region.

The technology used for VELO sensors is: n-implants in n-bulk technology. The strip isolation is done by p-spraying. The thickness of the sensors is 300 microns, and the pitch varies from about 40 microns for the inner part to about 100 in the outer-region. The ϕ sensors have a stereo sequence which varies from 10 to 20 degrees from inner to outer region. The average expected occupancy is close to 1 $\%$.

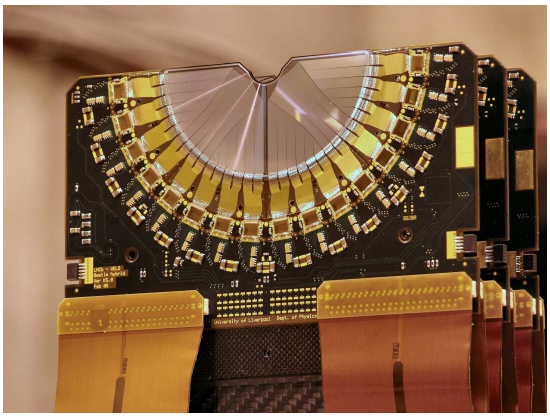
As VELO is retractable, for alignment reasons the mechanism of opening and closing VELO has to be quick and precise. Precision of about few microns are expected for the relative alignment of VELO's halves.

From Trigger's point of view, the VELO is part of the on-line software trigger. The Pile-UP VETO stations allow suppresion of multiple interactions, this is implemented in the hardware trigger, during first level L0 trigger.

Without VELO the precise finding of the vertices close to the interaction point would be problematic at best.



(a) Panoramix display of VELO, The 21 double-sensors of VELO



(b) Photo of a VELO sensor

A sensor is displayed in 2.6(b), outlining the half cylindrical frame. In the Panoramix display 2.6(a) the arrangement of the sensors in 2 rows for each half side is evident.

2.2.2 RICH 1 and 2

Particle identification is a fundamental requirement for LHCb. In most cases the particles are tagged by the RICH detectors and Muon-Calorimeter information. The two RICH detectors cover the full momentum range of LHCb. The RICH1 allows measurements of the low energy component which leaves the detector fiducial volume after the magnetic field. Upstream RICH1 is covering the momentum range between 1 and 60 GeV, using aerogel and C_4F_{10} radiators, whereas RICH2 covers the momentum segment of 15 GeV to 100 GeV and beyond with a C_4F_{10} radiator.

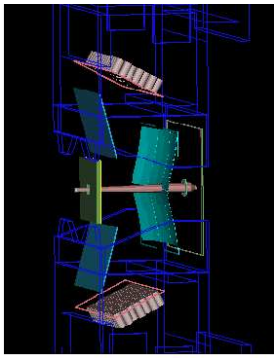


Figure 2.6: 1st RICH, 4 mirror structures and ensemble of photomultipliers on the upper and bottom sides

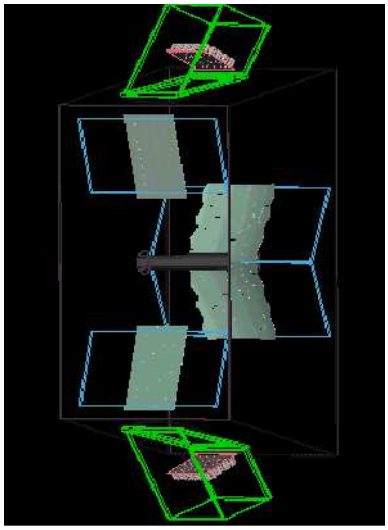


Figure 2.7: 2nd RICH, rotated by 90^0 around z, to have the mirror features clear, photomultipliers are actually placed on the sides and not up-bottom as in the figure

The acceptance is full detector acceptance for RICH1 in the upstream region: 25 milli-radians to 300 mrad (x-horizontal) and 250 mrad (y-vertical). The coverage of the RICH2 is not as complete, since this

detector is downstream: 15 mrad to 120 mrad(x) and 100 mrad (y). Both detectors have an ensemble of spherical and flat mirrors which reflect and focus the Cherenkov light produced by the superluminal particle, see figures 2.6 and 2.7. Magnetically isolated Hybrid Photon Detectors detect the Cherenkov light in the 200-600 nm range.

The pattern recognition tools allow to reconstruct the direction of the particle and the Cherenkov cone-ring. The likelihood function is computed for each particle ID hypothesis. From Monte Carlo studies, the RICH system is expected to provide good particle identification over the entire momentum range. Between 2 and 100 GeV the efficiency to tag a real Kaon with the correct label is 95 %, and the misidentification rate for a pion as Kaon is 5 %. For muons the rate of proper ID is 93 % . To provide these nice values the RICH makes use of a global pattern-recognition and the track information from the Tracking Stations.

2.2.3 Outer Tracker

The Outer Tracker is a collection of modules with straw-tubes [16]. There are two rows of straw-tubes per each module. The tubes are drift chamber cylindrical tubes, and have relative spacing of about 5 mm, making the pitch of about 2.5 mm if the double structure is taken in account. The counting gas filling inside the tubes is a combination of Argon and CO_2 . The straws are central inside the tube's volume.

A better precision - than other-wise available - of about 200 microns is obtained by extracting the drift time coordinate. Drift times of less than 50 ns are assumed. The acceptance is bounded by 300 mrad in x-horizontal coordinate and 250 mrad in y-coordinate. The total active area is about 87 m^2. The OT is divided in 3 stations, and each station has 4 layers with a stereo angle sequence of $\left(0, -5^0, 5^0, 0\right)$. Close to the beam the OT is bordering on the IT acceptance. The entire OT is composed of 168 long modules and 96 short modules. In total there are about 55000 single-straw channels.

Figure 2.8: One of the Outer Tracker Layers

The desired resolution on the $\mathrm{B_s^0} \rightarrow \mathrm{D_s^-}\pi^+$ mass window of $10\mathrm{MeV/c^2}$, would require a relative momentum resolution of $\delta p/p = 0.4$ % . The OT resolution of 200 microns will fullfil the demand on

the tracking reconstruction precision. An similar constraint on the OT performance requires a high track reconstruction efficiency of about 95 %. The latter demand is imposed by the need of reconstructing the high multiplicity B decay events.

2.2.4 ECAL, HCAL, Muon

One of the main tasks of the calorimeters is to provide the electron and photon candidates, together with the transverse-energy estimates for the hadrons. The retrieval of the direct photon and the neutral pion is essential for physics too.

The functioning principles of calorimeters are common. The scintillation light is recovered by the wavelength-shifting fibers and transmitted to the Photo-Multipliers. The HCAL has a higher gain than ECAL, since it receives a lower light yield. The thickness of the ECAL in radiation lengths is 25 and the thickness of HCAL in interaction lengths is only about 5.6, due to space limits and the flexibility of the trigger.

The pad/preshower or SPD/PS detector is build around similar lines with the other calorimeters but with a different technology. All calorimeters are segmented with lower cell size towards the beam-pipe.

ECAL

The electromagnetic calorimeter is build according to the shashlik calorimeter technology, sampling scintillator and lead, with plastic fiber wave shifters. The detector is split in 3 regions with the cell size for each region of: 16, 36, and 147 cm^2 respectively. The energy resolution of:

$$\frac{\sigma_E}{E} = \frac{10\%}{\sqrt{E}} \oplus 1\% \times (\text{E in GeV}) \tag{2.2}$$

corresponding to a resolution of $65\mathrm{MeV/c}^2$ and $75\mathrm{MeV/c}^2$ in 2 most relevant physical processes, see [16].

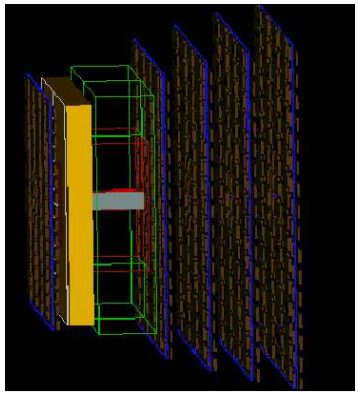


Figure 2.9: ECAL, HCAL and Muon system plotted together, the Muon has 5 stations in brown and outlined with blue, ECAL is the yellow slab, and the HCAL is transparent with green-red contours.

The angular coverage matches the angular coverage of the Tracking system with 300 mrad(x) and 250 mrad(y). A ECAL module is build from 2 mm thick lead , a reflective coating and a 4mm scintillator tile. In depth the 66 successive layers add to 25 X_0 radiation lengths.

HCAL

HCAL on the other hand is made form lower Z iron absorbers layers and scintillation tiles. The main distinctive feature of this calorimeter is that the tiles are lateral and parallel to the beam axis. The fiber wave shifters run parallel to the beam, and at the edges of the tiles. The HCAL is split into an inner and

an outer region with respect to the beam, and each region has a different square cell size of 131 mm and 262.6 mm respectively. The overall dimensions are of $3.6\mathrm{m}^2$ and $14.3\mathrm{m}^2$ for the inner part and overall respectively. The structure is movable and partitioned in two lateral sides each with 26 modules.

The energy resolution is:

$$\frac{\sigma_E}{E} = (65 \pm 5)\%/\sqrt{E} \oplus (9 \pm 2)\%(\text{E in GeV}) \tag{2.3}$$

Muon System

Muon detection is crucial in LHCb as the gold-plated decays $B_d^0 \rightarrow J/\psi\,(\mu^+\mu^-)\,K_s^0$ and $B_s^0 \rightarrow J/\phi\,(\mu^+\mu^-)\,\phi$ both contain a muon pair in the final state.

The Muons system is divided in 5 stations, with the first station from the interaction point, in front of the calorimeters. This choice is done to allow a higher precision in the reconstructed $\mathrm{p_T}$ of the muons. The stations are inter-spaced by iron absorbers 80 cm thick. This allows a selection of the most energetic muons, with a threshold of 6 GeV for the last station. Together with the calorimeters the thickness in radiation lengths from the RICH2 to the last muon station M5, is of 20.

The angular coverage in y is 16 to 258 mrad, and in the bending-plane is of 20 to 306 mrad. This assures a sensitivity of about 20 % for the muons coming from b-inclusive decays. The muon station is divided in lateral plane x-y into 4 regions with diferent granularity. The region sequence of R1, R2, R3, R4 has granularity scale in progression: 1:2:4:8. This balances the occupancy in each region to a common fraction.

The fact that the Muon-system is essential to the Trigger, makes the set of requirements imposed on this detector excruciating. The large flux of particle through this detector, means the efficiency, time resolution, and rate capability need to be optimal. A trigger efficiency of 95 % is demanded per each station, and a time window smaller than the bunch crossing of 25 ns.

2.3 Inner Tracker

The Inner Tracker detector is a silicon detector with single-sided strip sensors. It is located after the magnet in the downstream region, and it is so close to the beam-pipe that it practically embraces it. Though it covers only about 2 % of acceptance, close to 20 % of the tracks that are produced close to the interaction point, are detected in the Inner Tracker. Especially the high energy component of an event traverses the IT-acceptance.

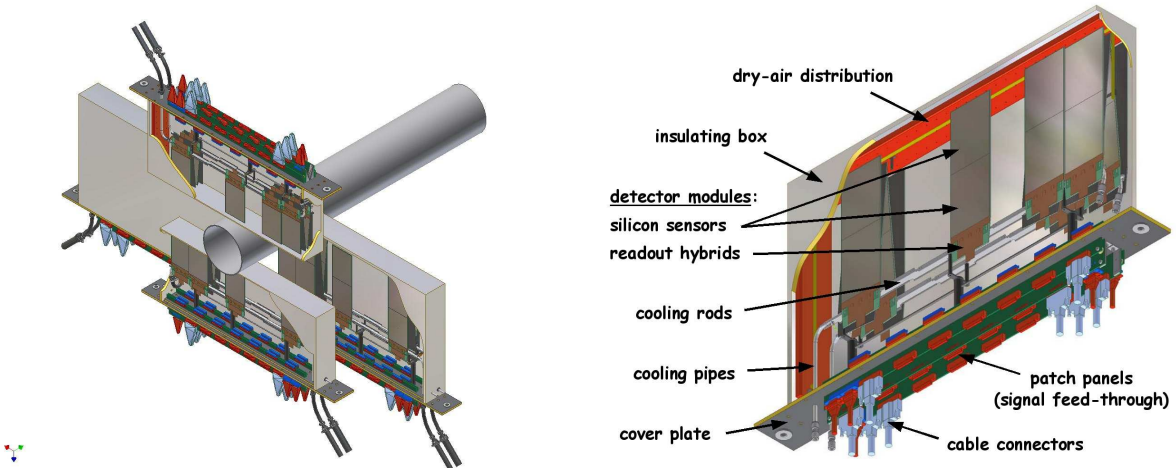
Structure and Granularity

The Inner Tracker is a highly hierarchical, as it has Stations, Boxes, Layers, Ladders, Sensors. The sensors are the lowest level, and ladders are the next. The ladders can be long ladders and short ladders, depending on how many sensors are containing, i.e. the long ladders have 2 silicon strip detectors, whereas the short ladders have only one. The ladders belong to detection layers in one of the 12 IT-Boxes. For one IT station there are 4 boxes, each with 4 layers. The Boxes naming scheme corresponds to the position with respect to beam, hence the Top-Box, the Bottom Box, the Left Box, and the Right Box. In the last year the Left (Right) box was renamed A-Box (C-Box), where the A stands for Access and the C for Cryostat, as the A-Box (C-box) is positioned closer to the entrance (to the cryostat). The long ladders belong to the Right or the Left Boxes, and the short to the Top and Bottom Boxes.

The figures 2.10 display the IT partition at level of a station in boxes, with each box containing 28 ladders arranged in 4 successive layers.

Silicon Sensor in IT

The technology of the IT single-sided layers is based on AC-coupled p^+-on-n strip detector type. The table 2.1 (reference [17])summarizes the IT-sensor characteristics.



(a) IT Station

(b) IT Right or Left Box

Figure 2.10: The 112 IT ladders in a Station, and 28 in a Box.

Technology	p^+-on-n
Thickness	320 μm
Physical dimensions	110 mm $\times$ 78 mm
Length readout strips	108 mm
Number of readout strips	384
Readout strip pitch	198 μm

Table 2.1: Silicon sensor characteristics

There are in total 336 ladders with half short ladders and half long, thus the number of sensors is 504 single-sided silicon sensors. Since the sensors are single sided, it is possible for one sensor to measure in the direction perpendicular to the strip, and the complementary direction is undetermined. To allow for track reconstruction, the IT has a stereo angle sequence of $0^0, -5^0, 5^0, 0^0$, corresponding to a rotation of ladders in each layer of a station. Seven ladders are grouped in a layer, and there are 16 layers in each IT-station. However, since each box in a station is relatively independent, overall we have 4 detection planes with 4 layers in each Station. Now these planes have the stereo sequence just mentioned, called in the IT case: 1st x, u, v, 2nd x.

The x planes have 0^0 stereo angle, and they all measure approximately along the x global direction of the LHCb system of coordinates. The u planes have associated a -5^0 stereo angle, and they measure along a direction slightly tilted compared to x global direction. The v planes have 5^0 stereo angle and are slightly tilted with respect to x global direction, too, but in the opposite direction. as the v layers.

Pitch values of about 200 microns, insure the high reconstruction precision, in the large occupancy domain of the Inner Tracker. The magnetic field through the first station of IT is still relatively strong, and as a consequence the magnetic field gradient along the beam axis is strong.

This work focuses on the alignment of the IT at level of ladders and layers explicitly. Since we expect to know to a very high precision the position of sensors on the ladders, the position of all sub-detector

elements is determined if we align both layers and ladders inside the IT. It is important to remember for the future that the Inner Tracker will have high energy tracks and closed-packed/dense hit distributions.

2.4 LHCb Trigger

The bunch crossing in the interaction point will have a frequency equal to 40 MHz. This number is much to large to attempt a full event storage. Since most of these events will have little of interesting from point of view of physics, it is required to filter the on-line event flow, before storing the relevant information to tape. The trigger system implemented for LHCb is optimized to store with great efficiency a $b\bar{b}$-type event.

The trigger is divided in a Level-0 trigger and a High-Level trigger. The Pile-up vetoing is included in Level-0 trigger. The veto system allows rejection of multiple coincidences within the span of a LHC-bunch, by estimating the number of primary pp interaction per bunch crossing. The Level-0 trigger is implemented at level of hardware, and the high-Level trigger (HLT) at level of a computer farm. The figure 2.11 is a good schematic summary of the trigger setup.

The reduction in interaction rate after the filter of each trigger is:

$$40 \text{ MHz} \rightarrow \boxed{\text{Level 0}} \rightarrow 1\text{MHz} \rightarrow \boxed{\text{HLT}} \rightarrow 2 \text{ kHz}$$

The Level-0 trigger is using information of the highest E_T hadron, electron, and photon clusters in calorimeter. Additionally it uses the two highest p_T muons in the Muon chambers (see [16]). It reduces the event flow from 40 MHz to 1 MHz.

The HLT is using the information provided by the Level-0 to decrease the event flow from 1 MHz to 2kHz. It uses the full event detector data, which is fed to the processing farm for analysis. The HLT is built to confirm the Level-0 candidates, and to reject the uninteresting events. It works by dividing the event sample into alleys, according to the type of trigger decision at Level-0. HLT requests that, the Level-0 candidates must have the Calorimeter clusters or Muon hits associated with T-stations tracks or Velo tracks. Next is the requirement to have a candidate with a combination of two large p_t tracks or/and two tracks with large impact parameters. This already reduces the event flow to 30 kHz. The HLT algorithms applied till now are part of the so called HLT1, opposed to the next set of algorithm that correspond to the second level of this trigger HLT2.

The second level of HLT is just the usual off-line analysis, performed without the time-expensive Kalman Filter method. The cuts in HLT2 are less severe than in the standard off-line tracking and analysis.

In this subsection I made intensive use of the latest-update information existent in bibliographical reference [16].

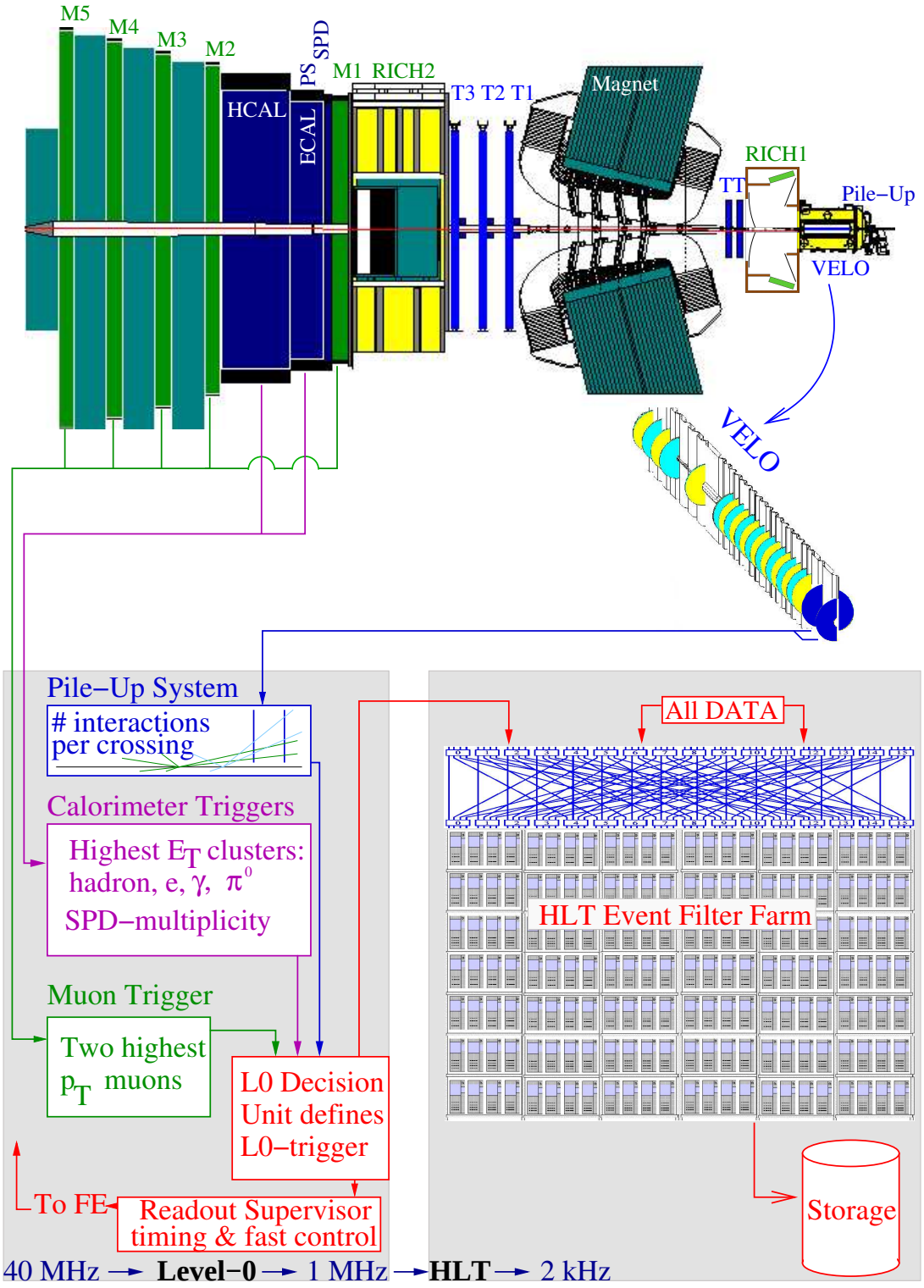


Figure 2.11: LHCb Trigger system

Le bon Dieu est dans le détail

Gustave Flaubert

Chapter 3

Mathematical Algorithms and Tools

3.1 Method of Least Square for a Linear Function

The Least Square Method has been used routinely in the past, by most of high-energy physics community. This has not changed much in the recent years, even with the surge in the number of new algorithms. The method is central in Gauss-Markov theorem [18–20].

Given a set of measurements x_i, with $i \in \mathbb{N}$, Gauss-Markov theorem states :
$\star$ provided that: fit model is linear, the errors or residuals have a null expectation value, are uncorrelated and have equal variance;
$\star$ then the least-square solution is the linear unbiased estimator with the smallest variance[1].

The method can be generalized to include the case when the residuals have unequal variance. The least square sum of the residuals gets a reweighting factor for each term. Generalized Least-Square method: Given a vector of parameters $\mathbf{a}$ then the best linear unbiased estimator is the solution that minimizes the function χ^2:

$$\chi^2 = \sum_{i,j} \frac{\left(U^{ij}_{measurement} - U_{model}\left(\mathbf{a}\right)\right)^2}{\sigma^2_{ij}} \quad \forall \text{ (i,j) track and measurement on track indices} \tag{3.1}$$

with σ^2_{ij} measurement variance.

Le us have a measurement sample $\{u_{ij}\}_{i=1,n;j=1,k}$ population drawn from a parent distribution which is assumed Gaussian with mean $U_{ij}(\mathbf{a})$ and standard deviation σ_{ij}, then the χ^2 function of 3.1 leads to values consistent with a χ^2 probability distribution with N_{dof} number of degrees of freedom.

$$N_{dof} = n * k - dimension\left(\mathbf{a}\right) \tag{3.2}$$

The minimization of the function in 3.1, for a linear model, has as outcome the system of equations[2]:

$$\frac{\partial \chi^2}{\partial \mathbf{a}}\left(\mathbf{a}\right) = 0 \tag{3.3}$$

The system 3.3 tell us that for a given function χ^2 a scalar field defined over the phase-space of $\{a_k\}_{k=1,d}$ set of parameters, its gradient must be null in the point of minimum.

In general the system dimension can be too large to compute directly a solution. In those cases that involve a large number of parameters a reduction in the dimension of the problem must be attempted first. The following section introduces a method that deals with millions of tracks and equations. It does it

[1] A best linear unbiased estimator of the fitted parameters is the one with the smallest mean squared error for every linear combination of the parameters themselves.

[2] $\frac{\partial}{\partial \mathbf{a}}$ is the derivative with respect to a matrix, hence a gradient for a vector as in most cases in this report

by reducing the size of the equation system to a much more manageable dimension, and simultaneously leads to a separation of track equations in independent sets of systems of equations for each track.

3.2 Millepede

A mathematical self-consistent and coherent approach to the alignment problem was achieved by Volker Blobel [21]. He named the method Millepede (Latin for a thousand footed bug, in English millipede or a myriapod). Its elegance resides in the fact that it is less heuristic and numerical than most alignment algorithms on the market. A powerful tool, it has the advantage of emerging naturally and directly from the least square method already described in 3.1.

As any software alignment method, at a basic level it does a minimization of the track residuals. This is achieved in the case of Millepede by minimizing a global χ^2 in the phase space of the alignment and track parameters. The novelty of this approach consists in the following ansatz: the method considers **simultaneously** the track parameters and alignment parameters as variables. The problem of course is that the minimization of such a global χ^2 leads to a system of equation with $N = n \times m + d$ equations, where n is the number of tracks, m the number of track parameters and d the number of alignment parameters. Even in relatively simple systems - e.g., straight track trajectories and 100 alignment parameters - the number of tracks can be however very large at about 10^5 to 10^6 tracks. This leads to a dimension in order of 10^7 for the equation system. The time needed to solve such systems is proportional to N^2, hence this normally takes an unrealistic amount of time.

It is a merit of Millepede that this problem is surpassed. The method transforms the N-dimensional system of linear equations in a d-dimensional system of equations. The resulting system has a number of equations equal to the number of alignment parameters. Reduction was achieved based on a simple and elegant set of linear transformation between the rows of the system matrix.

The transformations are favored by the sparse nature of the system matrix and more exactly by Millepede's implicit assumption of the independence between the track parameters belonging to distinct tracks. In this context the track equations and alignment equations can be separated and solved independently. This disentanglement between track parameters and alignment parameters takes place only for linear systems of equations. In general the alignment problem is non-linear, yet, the problem can be linearized. This is the subject of section 3.3.

Even though it is not the purpose of Millepede to find the track parameters, it is however possible to get them after finding the alignment parameters. Since the measurements of different tracks are not correlated, by extracting the alignment contribution from the measurement values, we decouple the problem of fitting the tracks in n independent track fits. With a dimension of track fit up to 5, each track parameter can be obtained.

Let us first formularize the problem. Take a linear track model:

$$r_j(\mathbf{t}) = \mathbf{t}^T \cdot \vec{\boldsymbol{\delta}}_j = \sum_{jk} t_k \cdot \delta_{jk} \tag{3.4}$$

where r_j is the model function given a measurement j for a particular track with parameters vector $\mathbf{t}^T = (x_0, t_x, y_0, t_y, ...)$. $\vec{\boldsymbol{\delta}}_j$ is the local derivatives' vector for a track, in each measurement j belonging to that track. Sometimes the track parameters are called local parameters, because in the context of the alignment these parameters are local in the sense that they belong only to a given track out of millions. The alignment parameters, on the other hand, enter the residual expressions of all measurements form all tracks, and hence they are global parameters.

For a given track and if the alignment degrees of freedom are considered null, the track parameters are found by minimizing:

$$\chi^2 = \sum_j \frac{\left(u_{measurement,\,j} - \mathbf{t}^T \vec{\boldsymbol{\delta}}_j\right)^2}{\sigma_j^2} \tag{3.5}$$

resulting the local-track system of equation:

$$\begin{cases} \boldsymbol{\Gamma} \cdot \mathbf{t} = \boldsymbol{\beta} \quad \text{with} \\ \text{matrix } \boldsymbol{\Gamma} = \sum_j \dfrac{\vec{\boldsymbol{\delta}}_j \vec{\boldsymbol{\delta}}_j^T}{\sigma_j^2} \quad \forall \text{ j measurement of track} \\ \text{and the vector } \boldsymbol{\beta} = \sum_j \dfrac{\vec{\boldsymbol{\delta}}_j u_{measurement,\,j}}{\sigma_j^2} \end{cases} \tag{3.6}$$

For a linear system of equations, vector $\vec{\boldsymbol{\delta}}_j$ is a sequence of constants. These are the derivatives of the track model with respect to track parameters: $\vec{\boldsymbol{\delta}}_j^T = \frac{\partial r_j}{\partial \mathbf{t}}$.

In the reversed case, considering track parameters as fixed and their contribution already subtracted from each measurement value, then the same reasoning applies for alignment parameters. The sum over all measurements belonging to all tracks:

$$\chi^2 = \sum_j \frac{\left(u_{measurement,\,j} - \mathbf{a}^T \vec{\boldsymbol{\Delta}}_j\right)^2}{\sigma_j^2} \quad \text{, where model is: } r_j = \mathbf{a}^T \vec{\boldsymbol{\Delta}}_j \text{ ,} \tag{3.7}$$

is the function to be minimized in the phase space of the alignment parameters $\mathbf{a}$. $\vec{\boldsymbol{\Delta}}_j$ are the so called "global" derivatives for a measurement j, usually this vector is sparse. The global derivatives are derivatives of the model function with respect to the alignment parameters. The only non-null global derivatives are those derivatives with respect to the alignment parameters appertaining to the sensors where the measurement is defined. E. g., we have 3 sensors to align, each with one alignment degree of freedom per sensor, then: $\vec{\boldsymbol{\Delta}}_1^T = \frac{\partial r_1}{\partial \mathbf{a}} = c_1 \times (1,0,0)$, $\vec{\boldsymbol{\Delta}}_2^T = c_2 \times (0,1,0)$, and $\vec{\boldsymbol{\Delta}}_3^T = c_3 \times (0,0,1)$, c_i being constants.

The minimizing solution is similar to the upper system of equations for a track:

$$\begin{cases} \mathbf{C} \cdot \mathbf{a} = \mathbf{b} \quad \text{with} \\ \text{matrix } \mathbf{C} = \sum_j \dfrac{\vec{\boldsymbol{\Delta}}_j \vec{\boldsymbol{\Delta}}_j^T}{\sigma_j^2} \\ \text{and vector } \mathbf{b} = \sum_j \dfrac{\left(\vec{\boldsymbol{\Delta}}_j u_{measurement,\,j}\right)}{\sigma_j^2} \end{cases} \tag{3.8}$$

Of course, as we have already established, we need a model with variable track and alignment parameters. The following model fulfills these requirements:

$$U_{model}^{ij}\left(\mathbf{t}_i, \mathbf{a}\right) = \vec{\boldsymbol{\delta}}_j^T \cdot \mathbf{t}_i + \vec{\boldsymbol{\Delta}}_j^T \cdot \mathbf{a} \tag{3.9}$$

The local and global derivatives vectors are for model 3.9:

$$\vec{\boldsymbol{\delta}}_{ij}^T = \frac{\partial r_{ij}}{\partial \mathbf{t}_i} \equiv \vec{\boldsymbol{\delta}}_j^T \tag{3.10}$$

$$\vec{\boldsymbol{\Delta}}_{ij}^T = \frac{\partial r_{ij}}{\partial \mathbf{a}} \equiv \vec{\boldsymbol{\Delta}}_j^T \quad \forall \text{ i'th track} \tag{3.11}$$

In the case of the global χ^2, as function of $\mathbf{t}_i$ and $\mathbf{a}$ the minimization is equivalent to the next set of formulae:

$$\begin{cases} \chi^2 = \sum_{ij} \dfrac{\left(u_{ij} - \vec{\boldsymbol{\delta}}_j^T \cdot \mathbf{t}_i - \vec{\boldsymbol{\Delta}}_j^T \mathbf{a}\right)^2}{\sigma_{ij}^2} \\ \forall \text{ i track index and j measurement belonging to it;} \\ \dfrac{\partial \chi^2}{\partial \mathbf{a}} = 2 \sum_{ij} w_{ij} \left[\mathbf{a}^T \vec{\boldsymbol{\Delta}}_j \vec{\boldsymbol{\Delta}}_j^T - \left(u_{ij} - \vec{\boldsymbol{\delta}}_j^T \mathbf{t}_i\right) \vec{\boldsymbol{\Delta}}_j^T\right] = 0 \quad \forall \text{ (i,j)} \\ \text{and if the parameters for distinct tracks are independent:} \\ \dfrac{\partial \chi^2}{\partial \mathbf{t}_i} = 2 \sum_{j} w_{ij} \left[\mathbf{t}_i^T \vec{\boldsymbol{\delta}}_j \vec{\boldsymbol{\delta}}_j^T - \left(u_{ij} - \vec{\boldsymbol{\Delta}}_j^T \mathbf{a}\right) \vec{\boldsymbol{\delta}}_j^T\right] = 0 \\ \text{for a given track i, and } \forall \text{ j measurement in it.} \end{cases} \tag{3.12}$$

Here, the weights $w_{ij} = \frac{1}{\sigma_{ij}^2}$. Just to mention, the precedent derivatives are the generalized matrix derivatives from matrix calculus[3]. The equations 3.12 could be reformulated into the following system of equations:

$$\begin{cases} \sum_{ij} w_{ij} \left(\vec{\boldsymbol{\Delta}}_j \vec{\boldsymbol{\Delta}}_j^T \mathbf{a} + \vec{\boldsymbol{\Delta}}_j \vec{\boldsymbol{\delta}}_j^T \mathbf{t}_i\right) = \sum_{ij} w_{ij} \vec{\boldsymbol{\Delta}}_j u_{ij} \Longleftrightarrow \sum_i \mathbf{C}_i \mathbf{a} + \sum_i \mathbf{G}_i \mathbf{t}_i = \sum_i b_i \\ \sum_j w_{ij} \left(\vec{\boldsymbol{\delta}}_j \vec{\boldsymbol{\delta}}_j^T \mathbf{t}_i + \vec{\boldsymbol{\delta}}_j \vec{\boldsymbol{\Delta}}_j^T \mathbf{a}\right) = \sum_j w_{ij} \vec{\boldsymbol{\delta}}_j u_{ij} \Longleftrightarrow \boldsymbol{\Gamma}_i \mathbf{t}_i + \mathbf{G}_i^T \mathbf{a} = \boldsymbol{\beta}_i \end{cases} \tag{3.13}$$

Notice the beautiful symmetry in the local-track and global-alignment parameters. There are new definitions of matrices:

$$\begin{cases} \mathbf{C}_i = \sum_j w_{ij} \vec{\boldsymbol{\Delta}}_j \vec{\boldsymbol{\Delta}}_j^T & \mathbf{G}_i = \sum_j w_{ij} \vec{\boldsymbol{\Delta}}_j \vec{\boldsymbol{\delta}}_j^T \\ \mathbf{b}_i = \sum_j w_{ij} \vec{\boldsymbol{\Delta}}_j u_{ij} & \boldsymbol{\beta}_i = \sum_j w_{ij} \vec{\boldsymbol{\delta}}_j u_{ij} \\ \boldsymbol{\Gamma}_i = \sum_j w_{ij} \vec{\boldsymbol{\delta}}_j \vec{\boldsymbol{\delta}}_j^T & \end{cases} \tag{3.14}$$

A matrix equation equivalent to 3.13 can now be given:

$$\begin{array}{c|c} 1\times \\ \mathbf{G}_1\boldsymbol{\Gamma}_1^{-1}\times \\ \cdots \\ \mathbf{G}_l\boldsymbol{\Gamma}_l^{-1}\times \\ \cdots \\ \mathbf{G}_k\boldsymbol{\Gamma}_k^{-1}\times \\ \cdots \\ \mathbf{G}_n\boldsymbol{\Gamma}_n^{-1}\times \end{array} \left| \begin{array}{c|cccccccccc} \sum_i \mathbf{C}_i & \mathbf{G}_1 & \cdots & \cdots & \mathbf{G}_l & \cdots & \cdots & \mathbf{G}_k & \cdots & \cdots & \mathbf{G}_n \\ \hline \mathbf{G}_1^T & \boldsymbol{\Gamma}_1 & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \mathbf{G}_l^T & 0 & \cdots & 0 & \boldsymbol{\Gamma}_l & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \mathbf{G}_k^T & 0 & \cdots & \cdots & \cdots & \cdots & 0 & \boldsymbol{\Gamma}_k & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \mathbf{G}_n^T & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & 0 & \boldsymbol{\Gamma}_n \end{array} \right) \left(\begin{array}{c} \mathbf{a} \\ \hline \mathbf{t}_1 \\ \cdots \\ \mathbf{t}_l \\ \cdots \\ \mathbf{t}_k \\ \cdots \\ \mathbf{t}_n \end{array} \right) = \left(\begin{array}{c} \sum_i \mathbf{b}_i \\ \hline \boldsymbol{\beta}_1 \\ \cdots \\ \boldsymbol{\beta}_l \\ \cdots \\ \boldsymbol{\beta}_k \\ \cdots \\ \boldsymbol{\beta}_n \end{array} \right) \tag{3.15}$$

In 3.15, if the subsystem of those matrix rows corresponding to the index *l* is multiplied by $\mathbf{G}_l\boldsymbol{\Gamma}_l^{-1}$ and the result is subtracted from the first rows corresponding to the alignment rows, than the latter system does not depend any longer on track *l* parameters $\mathbf{t}_l$. This can be done with all track parameter rows. The

[3] I am using the usual matrix derivative rules $\frac{\partial(\mathbf{Y}^T\mathbf{Z})}{\partial \mathbf{X}} = \mathbf{Z}^T \frac{\partial \mathbf{Y}}{\partial \mathbf{X}} + \mathbf{Y}^T \frac{\partial \mathbf{Z}}{\partial \mathbf{X}}$

multiplication factors are given in the extreme right side of equation 3.15 for each row ensemble. The procedure leads to the simplified system:

$$\left(\begin{array}{c|ccc} \sum_i \left(\mathbf{C}_i - \mathbf{G}_i\mathbf{\Gamma}_i^{-1}\mathbf{G}_i^T\right) & 0 & \cdots & 0 \\ \hline \cdots & \cdots & \cdots & \cdots \\ \mathbf{G}_k^T & \cdots & \mathbf{\Gamma}_k & \cdots \\ \cdots & \cdots & \cdots & \cdots \end{array}\right) \left(\begin{array}{c} \mathbf{a} \\ \hline \cdots \\ \mathbf{t}_k \\ \cdots \end{array}\right) = \left(\begin{array}{c} \sum_i \left(\mathbf{b}_i - \mathbf{G}_i\mathbf{\Gamma}_i^{-1}\boldsymbol{\beta}_k\right) \\ \hline \cdots \\ \boldsymbol{\beta}_k \\ \cdots \end{array}\right) \tag{3.16}$$

The track parameters' part in equation 3.16 is separated by the horizontal line and is the same as the track parameters' part in 3.15. However the part of the matrix equation which is in the upper side with respect to the horizontal line, has been changed and is no longer coupled to the track parameters. Since for this linear case, we are interested only on the alignment parameters, the initial system of equations 3.15 has been reduced to a system of dimension equal to the number of alignment parameters. In practice, a system of equations with a dimension within millions is reduced to one with a dimension of few hundreds.

Isolating the interesting part of 3.16:

$$\mathbf{C}'\mathbf{a} = \mathbf{b}' \quad \text{with} \begin{cases} \mathbf{C}' = \sum_i \left(\mathbf{C}_i - \mathbf{G}_i\mathbf{\Gamma}_i^{-1}\mathbf{G}_i^T\right) \\ \mathbf{b}' = \sum_i \left(\mathbf{b}_i - \mathbf{G}_i\mathbf{\Gamma}_i^{-1}\boldsymbol{\beta}_i\right) \end{cases} \tag{3.17}$$

The alignment parameters are given from 3.17, by: $\mathbf{a} = \mathbf{C}'^{-1}\mathbf{b}$, with the matrix $\mathbf{C}'^{-1}$ as the covariance matrix of alignment vector $\mathbf{a}$. The inversion can be performed, since the matrix dimension is manageable, or alternatively, a diagonalisation could be used instead as a preliminary step towards solving 3.17.

3.2.1 Lagrange Multiplier Method

In general, the alignment problem of the equation 3.17 has a singular $\mathbf{C}'$ matrix. This is due to the existence of the transformations - in the phase space of the track and alignment parameters - that keep the minimizing χ^2 quantity constant. Thus there are an infinity of possible solutions, all minimizing simultaneously χ^2. It is a typical not well defined problem. To define well the alignment problem, it is needed to understand that a system of detectors without any external reference is susceptible to suffer global transformations under alignment. There are 12 a priory possible global transformations, 3 translation, 3 rotations, 3 shearings, and 3 scalings. I shall discuss these transformation in the future chapters. For now it is sufficient to give the recipes to deal with these global modes.

The method of Lagrange multipliers as given in [21] uses Lagrange multipliers to constrain global degrees of freedom. A global translation could be fixed by requiring that the detector system should not move in average, e.g., $\sum_j \delta x_j = 0$ fixes the average x shift for the individual alignment parameters δx_j. This type of constraints can be in general formulated as :

$$\mathbf{a}^T \cdot \mathbf{f} = f_0 \tag{3.18}$$

Observe the linear relationship of these constraints, and hence the linearity requirement of Lagrange multiplier method is fulfilled. Via the Lagrange multiplier λ the upper constraints could be included in the definition of the χ^2 function from the equation 3.7 as:

$$\begin{cases} F = \sum_{ij} w_{ij} \left(u_{ij} - \vec{\boldsymbol{\Delta}}_j^T \mathbf{a} \right)^2 + \lambda \left(\mathbf{a}^T \mathbf{f} - f_0 \right) \\ \dfrac{\partial F}{\partial \mathbf{a}} = 0 \\ \dfrac{\partial F}{\partial \lambda} = 0 \end{cases} \iff \left(\begin{array}{c|c} \mathbf{C} & \mathbf{f} \\ \hline \mathbf{f}^T & 0 \end{array} \right) \left(\begin{array}{c} \mathbf{a} \\ \hline \lambda \end{array} \right) = \left(\begin{array}{c} b \\ \hline f_0 \end{array} \right) \quad (3.19)$$

The new system of equations 3.19 has a dimension larger by 1 than the old system. This extra line and column insure that the alignment parameters respect the constrain relation of 3.18. In the same time the Lagrange multiplier could be considered just as an additional alignment parameter. Similarly, the Lagrange multiplier method could be applied for the Millepede's system from 3.15.

There is a particular choice of constraints that I found very useful. To prevent any of the 12 global modes it suffices to fix just one of the alignment parameters at the time. Thus 12 alignment parameters fixed instead of 12 average relationships. The two ways of constraining a system are mathematically equivalent. The latter method has the small advantage that it does not increase the dimensionality of the $\mathbf{C}'$ matrix but it is reducing it. In practical application, the so called "weak" modes appear. They are transformation modes that leave the global χ^2 almost unchanged, and they cause in the $\mathbf{C}'$ matrix a near-singular behavior. Due to additional perturbations of the system introduced by physical biases in the measurements and their weights, the importance of near-singular values of $\mathbf{C}'$ becomes paramount. It is thus useful to have a lower dimensionality of the alignment system, because it allows the tagging of the $\mathbf{C}'$ near-singular values more readily.

The Lagrange multiplier method and the method of parameter a priory fixing are implemented in the LHCb alignment software.

This was in a nutshell most of the mathematical foundation behind Millepede. Now it follows to go beyond the mere linear formalism and extend the method to a more realistic non-linear case.

3.2.2 Example of a Millepede Matrix

It is possible to take an example that is at the time very simple and still very realistic. Given 3 parallel and equally spaced detectors, there is an ensemble of tracks all with hits in each detector. This simplistic example is plotted in 3.1. It is a realistic setup because if we need to align the Inner Tracker detector at level of stations, we have a somewhat similar setup. Till now, the cosmic muons of real data, can align IT only at the level of stations[4]. This setup shows that to well define the alignment problem, only $\Delta\tau_2$ one of the 3 alignment parameters can be aligned, the other 2 must be a priory fixed, otherwise the global shift of all stations and the global shearing are not fixed[5].

For simplicity consider the measurement resolution $\sigma_{ij} = 1$. This makes the weights for each measurement $w_{ij} = 1$. The problem is bi-dimensional only, with τ and z as axes. If the distance between layers T1, T2, and T3 is z_0, and if we take the absolute z of T1 as $z = 0$, than the local and global derivatives are:

$$\begin{cases} \vec{\boldsymbol{\delta}}_1^T = (1, 0) & \vec{\boldsymbol{\delta}}_2^T = (1, z_0) & \vec{\boldsymbol{\delta}}_3^T = (1, 2z_0) \\ \vec{\boldsymbol{\Delta}}_1 = 0 & \vec{\boldsymbol{\Delta}}_2 = 1 & \vec{\boldsymbol{\Delta}}_3 = 0. \end{cases} \quad (3.20)$$

The precedent relations 3.14 lead to the following expressions:

[4] Actually there are 2 types of boxes in the IT stations that were aligned, each type independently of the other.

[5] The global transformation will be discussed in the next chapters.

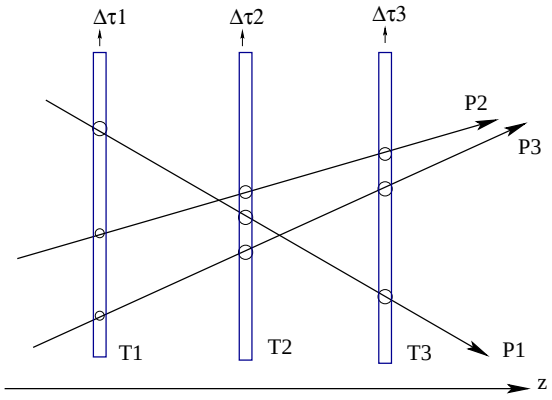


Figure 3.1: 3 parallel and equally spaced detectors

$$\left\{ \begin{array}{ccc} \mathbf{\Gamma}_i = \begin{pmatrix} 3 & 3z_0 \\ 3z_0 & 5z_0^2 \end{pmatrix} & \mathbf{G}_i = (1, z_0) & \boldsymbol{\beta}_i = \begin{pmatrix} \sum_j u_{ij} \\ z_0 \left(u_{i2} + 2u_{i3}\right) \end{pmatrix} \\ \mathbf{C}_i = 1 & \mathbf{b}_i = u_{i2} & \end{array} \right. \tag{3.21}$$

It follow that the equation 3.15 has the expression:

$$\left(\begin{array}{c|cccccc} \sum_i 1 & (1, z_0) & \cdots & \cdots & \cdots & (1, z_0) \\ \hline \begin{pmatrix} 1 \\ z_0 \end{pmatrix} & \begin{pmatrix} 3 & 3z_0 \\ 3z_0 & 5z_0^2 \end{pmatrix} & 0 & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \begin{pmatrix} 1 \\ z_0 \end{pmatrix} & \cdots & 0 & \begin{pmatrix} 3 & 3z_0 \\ 3z_0 & 5z_0^2 \end{pmatrix} & 0 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \begin{pmatrix} 1 \\ z_0 \end{pmatrix} & \cdots & \cdots & \cdots & 0 & \begin{pmatrix} 3 & 3z_0 \\ 3z_0 & 5z_0^2 \end{pmatrix} \end{array} \right) \left(\begin{array}{c} \Delta\tau_2 \\ \hline \begin{pmatrix} x_0^{(1)} \\ t_x^{(1)} \end{pmatrix} \\ \cdots \\ \begin{pmatrix} x_0^{(l)} \\ t_x^{(l)} \end{pmatrix} \\ \cdots \\ \begin{pmatrix} x_0^{(n)} \\ t_x^{(n)} \end{pmatrix} \end{array} \right) =$$

$$\left(\begin{array}{c|ccccccc} u_{i2} & \sum_j u_{1j}, & z_0 \left(u_{12} + 2u_{13}\right) & \cdots & \sum_j u_{lj}, & z_0 \left(u_{l2} + 2u_{l3}\right) & \cdots & \sum_j u_{nj}, \quad z_0 \left(u_{n2} + 2u_{n3}\right) \end{array} \right)^T \tag{3.22}$$

Solving 3.22 gives: $\mathbf{C}' = \sum_i \frac{2}{3}$ and $\mathbf{b}' = \sum_i \left(2u_{i2} - u_{i1} - u_{i3}\right)/3$, which in turn give the alignment:

$$\Delta\tau_2 = \frac{\sum_i \left(2u_{i2} - u_{i1} - u_{i3}\right)}{\sum_i 2} \tag{3.23}$$

3.3 Nonlinear Problems and an "Enlarged" Millepede

One of my personal contribution to the LHCb alignment framework was a linearization method with a stable iterative procedure[6]. The method can be used with Millepede which is embedded in an iterative loop. The linearization of a non-linear model allows the use of the linear least squares method, and the algorithm[7] computes corrections to the alignment and track parameters at each iterative step. The method outlined here is similar to the iterative methods included in the latest updates of the Millepede,

[6]This is correct for the alignment that uses Millepede, there is an additional concurrent algorithm which to my knowledge is similar to the one outlined here.

[7]I refer here to a mathematical algorithm and not to what the software group calls a "Gaudi Algorithm", which is more or less a software implementation of first

e.g. Millepede II[8] [22], it was however developed independently [23–25].

To provide the mathematical framework for this approach I start again form the χ^2 function of 3.12. In general, the local and global derivatives, $\vec{\boldsymbol{\delta}}$ and $\vec{\boldsymbol{\Delta}}_j$, respectively, depend on the track parameters and even on the global-alignment parameters themselves:

$$\vec{\boldsymbol{\delta}}_j \longrightarrow \vec{\boldsymbol{\delta}}_{ij}\left(\mathbf{t}_i, \mathbf{a}\right) \tag{3.24}$$

$$\vec{\boldsymbol{\Delta}}_j \longrightarrow \vec{\boldsymbol{\Delta}}_{ij}\left(\mathbf{t}_i, \mathbf{a}\right) \tag{3.25}$$

This happens as a result of the non-linear model function. It has the general form:

$$\chi^2 = \sum_{ij} w_{ij}\left(u_{ij} - \vec{\boldsymbol{\delta}}_{ij}^{T}\left(\mathbf{t}_i, \mathbf{a}\right)\cdot \mathbf{t}_i - \vec{\boldsymbol{\Delta}}_{ij}^{T}\left(\mathbf{t}_i, \mathbf{a}\right)\cdot \mathbf{a}\right)^2 = \sum_{ij} w_{ij}\left[u_{ij} - U_{model}^{ij}\left(\mathbf{t}_i, \mathbf{a}\right)\right]^2 \tag{3.26}$$

To search for the minimum of the upper function 3.26, there are several methods available "on the market", and some of them are given in the references [26] and [20]. Part of these algorithms involve computing the χ^2 at certain points in the phase-space, with some particular scheme for choosing iteratively sequences of points. These methods work directly with the actual χ^2 map, and take advantage of the local topology of the chi square function near the estimation point, e.g., grid or gradient methods. Other methods, call semi-analytical methods, make prior approximations to the χ^2 and take advantage that in the vicinity of the minimum point the function is parabolic-like[9]. These are the methods that have fast convergence rate, and are choosing their own iterative step. They are ideally suited for our task at hand because the misalignments are considered small with respect to the other geometrical parameters. Thus we are entitled to think that by choosing as a first guess the null alignment parameters, we start the iterative procedure very close to the minimum of the real χ^2 function, and that the first parabolic approximation to the real function is a sufficient approximation[10]. It remains, then, to just let the algorithm converge to that solution which minimizes the real χ^2 function. This, however, is not always true, as it will be explained at the end of the current section.

To really extend the Millepede scheme to the non-linear case, we found useful to use only one of the semi-analytical methods. The method itself Taylor-expands not the χ^2 but the model function directly to the first order. It is equivalent to a method that uses the second-order Taylor expansion of the χ^2 function near the starting point in the phase-space of the alignment and track parameters. Let the starting values be $\left(\left\{\mathbf{t}_i^{\star}\right\}_{i=1,N}, \mathbf{a}^{\star}\right)$ for all track and alignment parameters, with N the number of tracks used to align the detector. Then the Taylor first-order expansion of the model function is:

$$\begin{aligned} U_{model}^{ij}\left(\mathbf{t}_i, \mathbf{a}\right) \;\approx\; & U_{model}^{ij}\left(\mathbf{t}_i^{\star}, \mathbf{a}^{\star}\right) + \left(\frac{\partial U^{ij}}{\partial \mathbf{t}_i}\left(\mathbf{t}_i^{\star}, \mathbf{a}^{\star}\right)\right)\delta\mathbf{t}_i + \\ & \left(\frac{\partial U^{ij}}{\partial \mathbf{a}}\left(\mathbf{t}_i^{\star}, \mathbf{a}^{\star}\right)\right)\delta\mathbf{a} + O\left(\delta\mathbf{t}_i^2, \delta\mathbf{a}^2, \delta\mathbf{t}_i\cdot\delta\mathbf{a}^T\right) \\ & \text{with: } \delta\mathbf{a} = \mathbf{a} - \mathbf{a}^{\star} \text{ and } \delta\mathbf{t}_i = \mathbf{t}_i - \mathbf{t}_i^{\star} \end{aligned} \tag{3.27}$$

The partial derivative vectors $\left(\frac{\partial U^{ij}}{\partial \mathbf{t}_i}\left(\mathbf{t}_i^{\star}, \mathbf{a}^{\star}\right)\right)$ and $\left(\frac{\partial U^{ij}}{\partial \mathbf{a}}\left(\mathbf{t}_i^{\star}, \mathbf{a}^{\star}\right)\right)$ play in the linearized model function form of the equation 3.27, the same role as the constant vectors $\vec{\boldsymbol{\Delta}}_j$ and $\vec{\boldsymbol{\delta}}_j$ in the equation 3.12. It

[8] To my knowledge Millepede II includes a Newton-Raphson method similar to the Newton-Raphson method shown in the text.

[9] the parabolic approximation emerges from the minimum conditions, the first order derivatives approach null value near the minimum

[10] Sufficient in the sens that it provides an approximation that is within a micron of the real value, or for rotations within an equivalent value.

follows that the linearized model function can be used in the Millepede method to extract the corrections $\left(\{\delta\mathbf{t}_i\}_{i=1,N}, \delta\mathbf{a}\right)$ to the initial values.

$$\begin{aligned}\chi^2 \approx \Xi^{(1)} &= \sum_{ij} w_{ij} \left\{ u_{ij} - U^{ij}_{model}\left(\mathbf{t}_i^\star, \mathbf{a}^\star\right) - \left[\frac{\partial U^{ij}}{\partial \mathbf{t}_i}\left(\mathbf{t}_i^\star, \mathbf{a}^\star\right)\right]\delta\mathbf{t}_i - \left[\frac{\partial U^{ij}}{\partial \mathbf{a}}\left(\mathbf{t}_i^\star, \mathbf{a}^\star\right)\right]\delta\mathbf{a} \right\}^2 \\ &= \sum_{ij} w_{ij} \left[\delta u_{ij}^{(1)} - \left(\vec{\boldsymbol{\delta}}_{ij}^{(1)}\right)^T \delta\mathbf{t}_i - \left(\vec{\boldsymbol{\Delta}}_{ij}^{(1)}\right)^T \delta\mathbf{a} \right]^2 \end{aligned} \tag{3.28}$$

To be able to iterate the procedure just described, we apply the Newton-Raphson method [20, 26]. The method just outlined substitutes the real χ^2 function with a second order approximation $\Xi^{(1)}$, and then finds its minimum with Millepede. In this way we compute corrections to both alignment and track parameters. The parameters themselves are updated with these corrections, and henceforward ready to be used in a subsequent iteration. A next Newton-Raphson step must use this corrected values to compute a new χ^2 approximation. This last function gets usually closer to the real χ^2 and implicitly its minimization produce corrections to the track-alignment parameters which approach the reality. A next step is initiated again, and the newly updated parameters used, and so on. The procedure continues until a convergence criterion is fulfill or till an upper-bound iteration index is reached. This is equivalent to the following mathematical iterative scheme:

$$\left\{\begin{array}{ll}
\text{1. given starting values:} & \left\{\mathbf{t}_i^{(n)}\right\}_{i=1,N}, \mathbf{a}^{(n)} \\
\text{2. subtract:} & \delta u_{ij}^{(n)} = u_{ij} - U^{ij}_{model}\left(\mathbf{t}_i^{(n)}, \mathbf{a}^{(n)}\right), \\
\text{3. compute} & \left(\vec{\boldsymbol{\delta}}_{ij}^{(n)}\right)^T = \dfrac{\partial U^{ij}}{\partial \mathbf{t}_i}\left(\mathbf{t}_i^{(n)}, \mathbf{a}^{(n)}\right) \;\forall\, i = 1, N \\
\text{local-global derivatives:} & \\
 & \left(\vec{\boldsymbol{\Delta}}_{ij}^{(n)}\right)^T = \dfrac{\partial U^{ij}}{\partial \mathbf{a}}\left(\mathbf{t}_i^{(n)}, \mathbf{a}^{(n)}\right) \\
\text{4. approximate:} & \chi^2 \approx \Xi^{(n)} = \sum_{ij} w_{ij}\left[\delta u_{ij}^{(n)} - \left(\vec{\boldsymbol{\delta}}_{ij}^{(n)}\right)^T \delta\mathbf{t}_i^{(n)} - \left(\vec{\boldsymbol{\Delta}}_{ij}^{(n)}\right)^T \delta\mathbf{a}^{(n)}\right]^2 \\
 & \delta\mathbf{t}_i^{(n)} = \mathbf{t}_i - \mathbf{t}_i^{(n)},\; \delta\mathbf{a}^{(n)} = \mathbf{a} - \mathbf{a}^{(n)},\; \delta u_{ij}^{(n)} = u_{ij} - U^{ij}_{model}\left(\mathbf{t}_i^{(n)}, \mathbf{a}^{(n)}\right) \\
\text{5. minimize:} & \boxed{\text{Millepede}} \\
\text{6. retrieve and update} & \mathbf{a}^{(n+1)} = \mathbf{a}^{(n)} + \delta\mathbf{a}^{(n)} \\
\text{alignment:} & \\
\text{7. disentangle alignment} & \delta u_{ij}^{(n)} \rightarrow \delta u_{ij}^{(n)} - \left(\vec{\boldsymbol{\Delta}}_{ij}^{(1)}\right)^T \delta\mathbf{a} = \delta\mathcal{U}_{ij}^{(n)} \Rightarrow \\
\text{from measurements:} & \boldsymbol{\beta}_i^{(n)} = \sum_j \dfrac{\left(\vec{\boldsymbol{\delta}}_j^{(n)} \delta\mathcal{U}_{ij}^{(n)}\right)}{\sigma_j^2}, \quad \boldsymbol{\Gamma}^{(n)} = \sum_j \dfrac{\vec{\boldsymbol{\delta}}_j^{n}\left(\vec{\boldsymbol{\delta}}_j^{(n)}\right)^T}{\sigma_j^2} \\
\text{8. do all local fits:} & \text{minimum of the local } \chi^2_{local} \Rightarrow \delta\mathbf{t}_i^{(n)} = \left(\boldsymbol{\Gamma}_i^{(n)}\right)^{-1} \boldsymbol{\beta}_i^{(n)} \\
\text{9.update track parameters:} & \mathbf{t}_i^{(n+1)} = \mathbf{t}_i^{(n)} + \delta\mathbf{t}_i^{(n)}
\end{array}\right. \tag{3.29}$$

In the precedent scheme, at point 7, the $\boldsymbol{\beta}_i^{(n)}$ and $\boldsymbol{\Gamma}_i^{(n)}$ matrices are equivalent to those form equation 3.6.

The alignment and track parameters[11] $\left(\left\{\mathbf{t}_i^{(n+1)}\right\}_{i=1,N}, \mathbf{a}^{(n+1)}\right)$ are ready to restart the loop 3.29. An equivalent logical scheme for the Newton-Raphson step is:

n-1 → **Derivatives** → **Millepede Global fit** → **Local Fit for each track**

- Derivatives: computes derivative values for Millepede
- Millepede Global fit: finds alignment parameters
- Local Fit for each track: finds the track parameters

→ n+1

Two nice examples of Newton-Raphson being applied are plotted courtesy of Mathematica [27], 3.2.

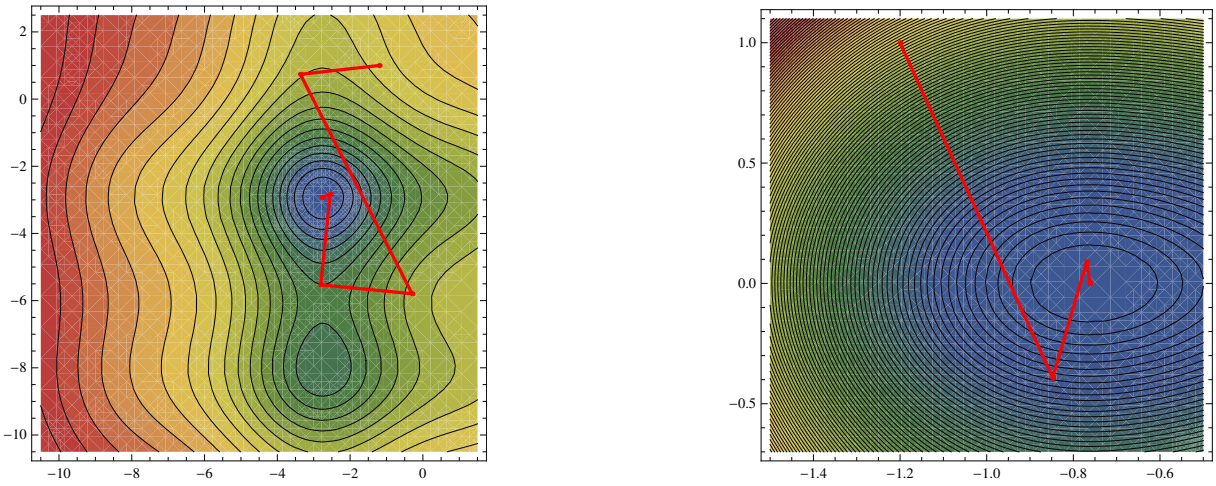


Figure 3.2: Plots of Newton-Raphson steps with the topological contours for the surface of the minimized function. Right Side of plot the function: $f(x,y) = 0.25x^2 + y^2/9 + 6 + x + y + \cos(x) + \sin(y+1)$, and in the Left Side: $g(x,y) = x^2 + y^2/9 + 20 + 0.1x + y + \exp(\cos(x)) + \exp(-\sin(y))$.

3.3.1 Quality Cuts and the Removal of Outliers

In most application of the upper method we can expect biases in the distribution of the residuals $u_{ij} - U^{ij}_{model}$, even for a linear case. These biases can appear due to the physical processes that perturb the track trajectory or just due to stochasticity of the detection process. It is useful to impose quality-cuts on the track sample, and at each iteration depending on the alignment quality, avoid the use of tracks that do not fulfill some quality criteria. In alignment, we found it useful to cut on the local χ^2_{local} value of a fitted track, with χ^2 calculated at point 8 in the 3.29 scheme. The procedure removes outliers in the χ^2 distribution of all fitted tracks. To not bias the alignment parameters, the cut can be soft or non-existent for the first iterations, and then for consecutive iterations the cut becomes harder as the alignment improves quality. The global χ^2 of 3.12 is the quantity that allowed us to gauge the alignment quality form one iteration to the next. To the upper logical scheme is added a new logical box.

n-1 → **Derivative feed** → **Millepede Global fit** → **Local Fit for each track** → **track χ^2 cut flags outliers** → n+1

Care was exercised in using the added logic unit. We are changing the track sample used for alignment between iterations. In these conditions, the Newton-Raphson method is applied only once for a

[11] In this section the number of tracks is N, and the (n) index stands for the n-th iteration

fixed sample. This might create problems in extreme cases: if the track sample changes very rapidly during iterations, or the alignment matrix $\mathbf{C}'$ has one or more "weak modes". In these cases it was useful to preserve the alignment sample over more than one iteration[12], adding in this way an extra loop inside the initial iterative loop.

The idea of outlier removal by χ^2_{local} cuts was borrowed from Millepede, but used a little differently within Newton-Raphson.

3.3.2 Adaptive Step Size, the Quasi-Newton Step

As seen in the 3.2 figures, there might happen that the minimized function exhibits local minima in addition to the global minimum. This fact generates problems in practical applications especially when the dimension of the phase-space increases significantly. On such hypersurfaces, the topology becomes very complicated and the iterative procedure might end in a local minimum instead of the global minimum. To correct this potential problem, an adaptive-step-size method was implemented, that is similar to the quasi-Newton methods found in the "Numerical Recipes" book [26], or more exactly to the Levenberg-Marquardt algorithm from the same reference.

Even though our extended Millepede method chooses by construction its own alignment step $\delta\mathbf{a}$, it is apparent that for not well behaved χ^2-topologies near the absolute minimum this procedure could run into problems. An other factor is the possibility of divergence[13]. In running our implementation of Generalized Millepede, local minima and divergences were observed for some extreme cases. It was proven useful to use, in the case of a divergence, the following method:
$\star$ for an increase of global χ^2 function from one iteration to an other, revert to the old track parameters of previous iteration;
$\star$ change alignment values by half $\delta\mathbf{a}$ and repeat last iteration.

Since no update of alignment parameters is done till the global χ^2 is reduced, the last method prevents divergence by default. However there is a more fundamental reason why to use this approach. The Newton step of $\left(\{\delta\mathbf{t}_i\}_{1=1,N}\,\delta\mathbf{a}\right)$ is in a descent direction of the global χ^2 [26]. This means that there will always be a constant, which multiplied with this vector, will provide a suitable correction. This added correction will minimize the true $\chi^2\left(\{\mathbf{t}_i\}_{1=1,N}\ ,\ \mathbf{a}\right)$[14]. This is one of the methods recommended by the author of Millepede, too. To avoid local minima, different starting points of the iteration loop are chosen and the results compared.

For the convergence criteria there is no a priory universal formula. However, we implemented some monitoring on the alignment parameter value as it changes from one iteration to other. Similarly, the global χ^2 and its change during consecutive iterations are monitored. When the changes in the alignment parameters and χ^2 function become negligible the iterations are stopped.

Usually, I needed more restrictive convergence criteria to obtain realistic error estimates. The alignment values proved on the other hand more stable. In general it was not the non-linearities that caused the difficulties. I traced back the divergence problems to the biases in the residuals and to the presence of quasi-singularities in the $\mathbf{C}'$ matrix. These causes coupled with the quality cuts during the iterative procedure were found responsible for at least most of the divergent cases observed during several runs[15].

In the figure 3.3 the quasi-Newton step of this section is structured in a logical-unit based scheme.

[12] routinely 3 iteration were used with a constant sample

[13] this could be due to: biases in the residues distributions, singular or quasi-singular modes of $\mathbf{C}'$

[14] for a more detailed explanation the reference [26] Numerical Recipes, sections on Non-linear Least Squares and Nonlinear Systems of Equations, could be used.

[15] This is out of few hundred runs that performed flawlessly. The divergences observed were explained, and solved by improving the track quality and number, or by fixing the observed "weak modes"

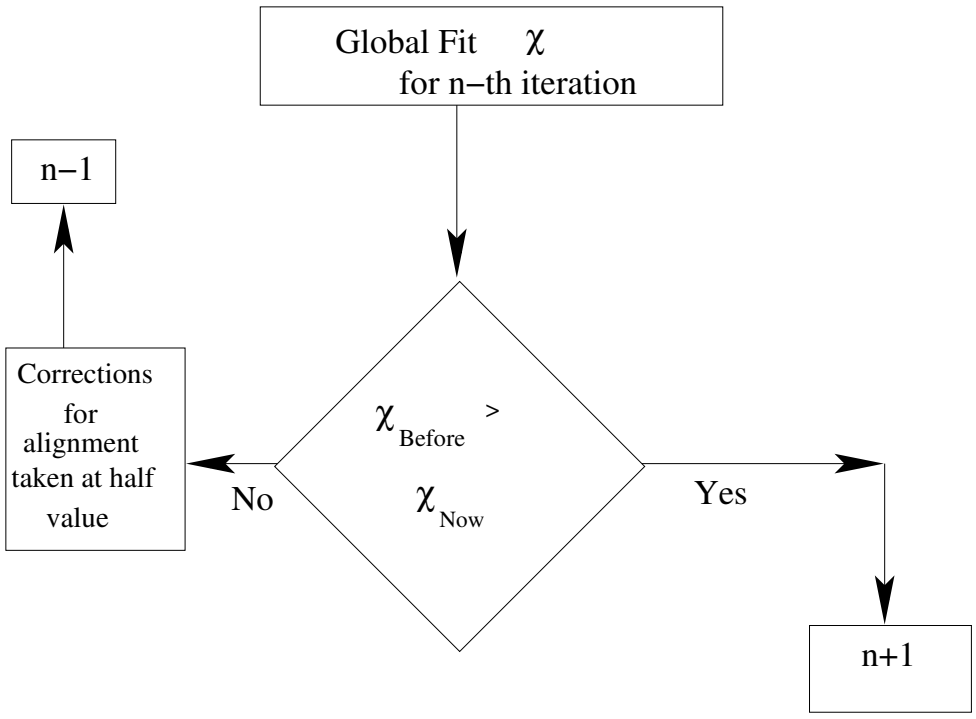


Figure 3.3: Extra quasi-Newton step for the Newton-Raphson iteration

"HC SVNT DRACONES" (Latin spelling for: hic sunt dracones) or "Hic sunt leones"

Anonymous

Chapter 4

Alignment Models and Tools

The model functions outlined in the following chapter are the results of an independent analysis by the report author. Even though I found several models in the "literature", however most were too simplistic, and a minority too lapidary described. It was thus required to start pretty close to "scratch" in developing a realistic model. Relying myself on some very general set of ground rules, I was able at least to some degree, to translate the proposed objective in reality. The following chapter introduces some of the alignment models used in aligning the LHCb's Inner Tracker. From seven tested models, I am describing first the simplest one used in a Toy Monte Carlo, after which I give the ones that I found the most useful in aligning the Inner Tracker with or without magnetic field. The latter two models were tested in full Monte Carlo simulations for different types of events.

4.1 Simple Alignment Model for a Toy Monte Carlo

For "historical" reasons, I am starting with the most minimal model that I could develop. Although it is far from realistic, this model proved most useful in understanding the later more realistic models. After all, this is the purpose of the "Toys".

In a first approach to an analytic alignment model we can make the following simplifying assumptions:
⋆ neglect the box structure and the beam, take 12 square detection layers;
⋆ the detection planes are parallel and perpendicular to a common axis z;
⋆ the middle of each square is situated along the z axis, thus co-linear;
⋆ the misalignments are applied to this "nominal geometry";
⋆ the strips on the detection surfaces are arranged in the IT-appropriate sequence: X1, U, V, X2 for each station, hence a stereo angle sequence of $\left(0, -5^0, 5^0, 0\right)$.

Within these assumptions, I took the definition of a measurement as **determined by the point in space where a rectilinear (parabolic[1]) track trajectory intersects the surface of the detection layer**. These positions allow to extract a numerical value corresponding to the position of the closest strip with respect to a reference strip from the sensor-plane of the detector. A smearing of the resulting value can be done to take into account the finite resolution [2].

Solving the problem of an intersection between a rectangular surface and a linear trajectory, a model is obtained that depends on the stereo angle for the plane in question. For a track index i, and a measurement on the track with index j, we have the track parameters $\mathbf{t} = \left(x_0^i, t_x^i, y_0^i, t_y^i, Q^i\right)$ and alignment parameters $\delta\tau_j$, δz_j, $\delta\alpha_j$, $\delta\beta_j$, $\delta\gamma_j$. The alignment parameters that do not correspond to the measurement's sensor, and therefore have the index different from j, do not enter the model function for the given measurement. Q depends on the curvature of the track when the magnetic field is non-null, otherwise it just zero.

[1] rectilinear for case with magnet-off, and parabolic for the case with magnet-on
[2] The toy considers all measurement resolutions constant and equal

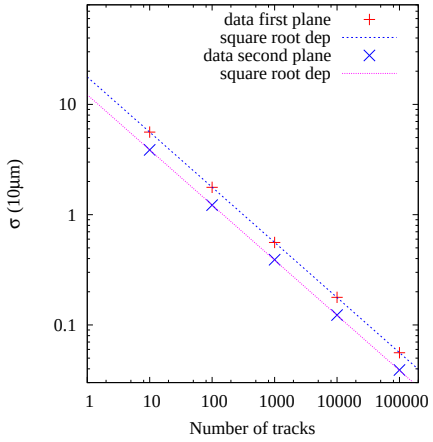
$$
\begin{aligned}
U_{model} = &\left(Q^i z_j^2 + z_j\, t_x^i + x_0^i\right)\, \cos\alpha_j - \left(z_j\, t_y^i + y_0^i\right)\, \sin\alpha_j - \delta\tau_j - \\
&-\delta\alpha_j \left(Q^i z_j^2 \sin\alpha_j + z_j\, t_x^i \sin\alpha_j + z_j\, t_y^i \cos\alpha_j + x_0^i \sin\alpha_j + y_0^i \cos\alpha_j\right) + \\
&+\delta z_j \left(t_x^i \cos\alpha_j + 2Q^i z_j \cos\alpha_j - t_y^i \sin\alpha_j\right) - \\
&-\delta\beta_j \left(Q^i z_j^2 + t_x^i z_j + x_0^i\right) \left(t_x^i \cos\alpha_j - t_y^i \sin\alpha_j + 2Q^i z_j \cos\alpha_j\right) - \\
&-\delta\gamma_j \left(t_y^i z_j + y_0^i\right) \left(t_x^i \cos\alpha_j - t_y^i \sin\alpha_j + 2Q^i z_j \cos\alpha_j\right)
\end{aligned}
\tag{4.1}
$$

$\delta\tau$ is the misalignment in the direction of the measurement; $\delta\alpha$ is the rotation around the beam axis, which, in the LHCb system of coordinates, is the z axis[3]; $\delta\beta$ is the rotation around the y axis; $\delta\gamma$ is the rotation around the x axis; and δz is the shift along the z axis. I took the rotation pivot in the center of each layer. The order of transformations is arbitrary, since I always approximate the misalignments with infinitesimal transformations[4].

Of course, there are a few problems with this model: first it does not consider a realistic hierarchical structure for IT[5], second the planes need not to be parallel, third the detection layers and ladders have non-coaxial centers, and the list continues.

Although this model is of limited value, it taught us a great deal about the possible global transformations that can not be fixed by the internal alignment of IT alone. In the same time, it allowed us to test, for the first time, the Millepede method in a more controlled context than we would have had if using the much more realistic and complex LHCb Monte Carlo.

An additional benefit from this model is that it incorporates together with the global modes specified in the previous chapter (see 3.2.1), some of the "weak" modes[6] found with the more realistic models. The fact that this model is a third order polynomial helped understand some of the difficulties that could arise from possible divergences and oscillations in the iterative results[7].



The alignment linear degree of freedom is fitted using 10 to 100000 tracks. The table bellow gives the dependence of four alignment parameter resolutions for 4 distinct detector planes.

N	$\sigma_1(\mu m)$	$\sigma_2(\mu m)$	$\sigma_5(\mu m)$	$\sigma_6(\mu m)$
10	56.1	38.7	60.5	54.8
100	17.7	12.2	19.1	17.3
10^3	5.6	3.9	6.1	5.5
10^4	1.78	1.23	1.92	1.74
10^5	0.56	0.39	0.61	0.55

Figure 4.1: The alignment precision dependence on the number of tracks in the alignment sample

Considering first only the linear part of the model 4.1, hence the $\delta\tau$ part of the model, and aligning only for this degree of freedom, I get by varying the number of track the usual square root law as displayed by the figure 4.1 and its added table.

In preparing the ground for a realistic alignment using the LHCb Monte Carlo data, I used the Toy Monte Carlo to align in all 5 possible degrees of freedom, $\delta\tau$, δz, $\delta\alpha$, $\delta\beta$, and $\delta\gamma$, respectively.

[3]For the LHCb system of coordinates the z-axis is actually tilted with respect to the beam axis by about 0.0036 rad.

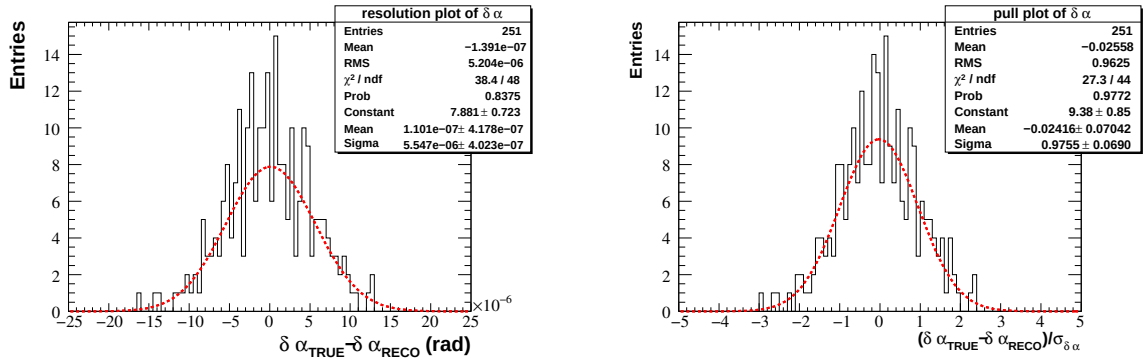
[4]The translations and rotations commute for infinitesimally small misalignments.

[5]IT has several hierarchical levels of which: Stations, Boxes, Layers, Ladders, and Sensors are the relevant for alignment

[6]"weak modes" are modes that can be only loosely constrained by an alignment procedure, due to a lack of connecting tracks between parts of the detector, or due to the field or geometry effects, see sections 5.4 or 6.4.4

[7]Here, we are used quality cuts on the track sample within the iterative procedure

To check the Millepede error propagation I made 251 independent alignments, each with independent track samples and independent misalignments. The results are shown in the pull plots[8] and resolutions plots of 4.2, and in the appendix 8.1, 8.2, 8.3, 8.4.



(a) Resolution plot

(b) Pull plot

Figure 4.2: Resolution and pull plots for $\delta\alpha$ in a Toy Monte Carlo

The pull plots are Normal distributions, and hence the previews plots show an exact agreement between the error estimates of Millepede and the true errors. The resolution plots have the same feature as the pulls. The global χ^2 of the alignment fit is given in Figure 4.3 for the last iterations before convergence. The number of degrees of freedom for the fit is given in the same figure on the left. The large number is due to the fact that about 100000 tracks were used in the alignment, and each track has 4 parameters.

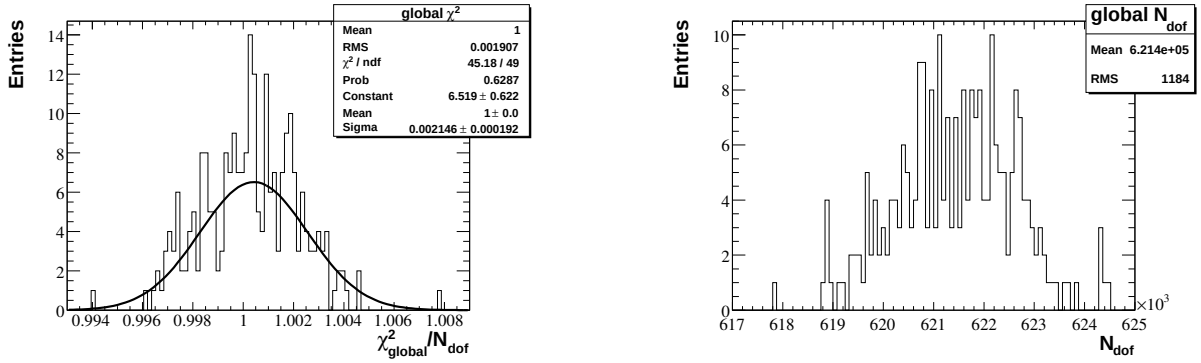


Figure 4.3: The reduced global χ^2 and the number of degrees of freedom for 251 toy Monte Carlo independent runs, each with 10^5 tracks

The χ^2 functions of the 251 runs are distributed in the final iteration of each run according to the probability density function of the χ^2-distribution. The latter function, in the limit of a very large number of degrees of freedom, approaches a rescaled Normal function with a root mean square of $\sqrt{2}/\sqrt{N_{dof}}$ equal in our case with about 0.0018 on average. This is almost exactly the observed root mean square of the right plot of 4.3.

As a concluding comment of this section, I have observed that for a large number of alignment parameters of non-linear kind, the Newton-Raphson iterative procedure begins to require quasi-Newtons steps, to avoid ending in a local minimum or even divergence. This was a warning for the future full LHCb Monte Carlo alignments, that we might encounter problems in aligning the less sensitive degrees

[8] Pull plots, in this context, are histograms of the error distributions scaled by the error estimates.

of freedom as δz and $\delta\gamma$. In some cases, for these degrees of freedom, the Millepede $\mathbf{C}'$ matrix develops close to singular behavior, due to the modes that are only weakly constrained[9]. The problem was solved with quasi-Newton steps that stabilize the results, or by increasing the number of tracks. This was not surprising, since the quasi-Newton method described in the previous chapter is adding robustness to the Newton-Raphson procedure.

4.2 Finding a Model in the Absence of the Field

A realistic alignment model was achieved by taking into consideration a few common-sense facts:
⋆ we do not know a priori the exact position of sensors within an IT sub-detector: IT-layers, IT-boxes, IT-stations;
⋆ these sensors have themselves 6 possible degrees of freedom that can take very different values depending on which part of the detector are we in;
⋆ one degree of freedom for a sensor can not be even in principle extracted, the measurement complementary-direction is not fixed;
⋆ the misalignments must be much lower than some detector dimension, e.g., the stereo angle of a U/V plane must be much larger than any possible rotational-misalignment, and the distance between the IT layers must be much larger than the misalignments along the beam.

Having the upper requirements fulfilled for an alignment model is not trivial at best. In order to accomplish this "wish list", I had to keep the number of simplifying assumptions to minimum, and define the problem as generally as possible. This is the opposite path to the one used in the last section, where before constructing the model we took as many as possible simplifying assumptions.

To begin, we can take a straight track trajectory for the particles. In the absence of the magnetic field and for particles that do not suffer dramatically from random scattering[10], this is almost an exact form for a real track trajectory.

$$\text{i'th track parameter vector: } \mathbf{t}_i = \left(x_0^i, t_x^i, y_0^i, t_y^i\right) \Longrightarrow \left\{ \begin{array}{rcl} x\left(z\right) & = & x_0^i + t_x^i z \\ y\left(z\right) & = & y_0^i + t_y^i z \end{array} \right. \tag{4.2}$$

Thus, in the LHCb system of coordinates, the line is parametrized by the variable z coordinate axis. The newly parametrized line is defined by the origin vector $\mathbf{O}_{origin} = \left(x_0^i, y_0^i, 0\right)$ and by the track direction versor: $\mathbf{V} = \left(t_x^i, t_y^i, 1\right) / \sqrt{\left(t_x^i\right)^2 + \left(t_y^i\right)^2 + 1}$.

$$\text{Any point on the line is given by: } \mathbf{R}_{track} = \mathbf{O}_{origin} + \mathbf{V} \times z \tag{4.3}$$

Before explaining the parametrization of the sensor, I make an argument in favor of choosing the measurement model as the point of intersection between the track and the sensor surface. A measurement model can be defined alternatively by the distance of closest approach between the particle track trajectory and the trajectory of the triggered IT-silicon-strip. It is however in my opinion impossible to a priori prefer one definition over the other, since as seen in the plot 4.4 approximately 50% of tracks have the closest approach point outside the silicon bulk of the sensor.

In the literature the distance of closest approach was sometimes favored, e. g., see [28]. However even then it was for a double sided silicon sensor, where different sides of the same silicon layer measure along different directions. For those sensor the distance of closest approach is the obvious choice, since the measurement is in fact an extrapolation towards the center of the silicon layer. Yet, there is no similar requirement for the single-sided IT sensors. With the number of tracks going through the left side of a strip being equal a priory to the number of tracks passing through the right, any differences between the definitions become negligible anyway. The choice of the model was influenced by the computational requirements, as it will become clear in the next paragraphs.

[9] The presence of very small values in the spectrum of the $\mathbf{C}'$ eigenvalues
[10] This physical process is sometime called Molière scattering, for reference see PDG Review [8]

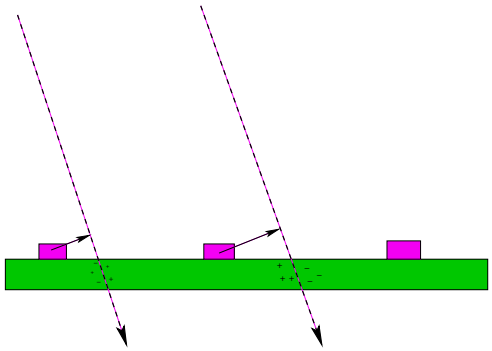


Figure 4.4: Sensor layer with two tracks, the track closest approach to the sensor strip is outside silicon bulk

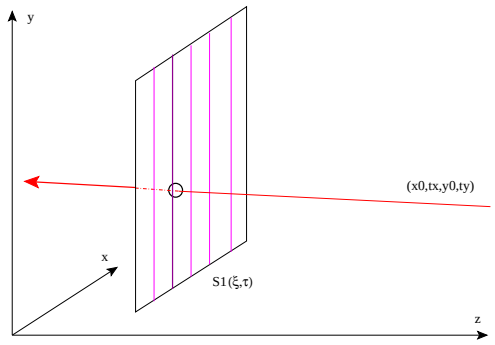


Figure 4.5: Track crossing the surface of a sensor with strips displayed in magenta. The track is parametrized by the vector (x0,tx,y0,ty), and the sensor surface by the versors $\vec{\xi}, \vec{\tau}$ and the center of the sensor

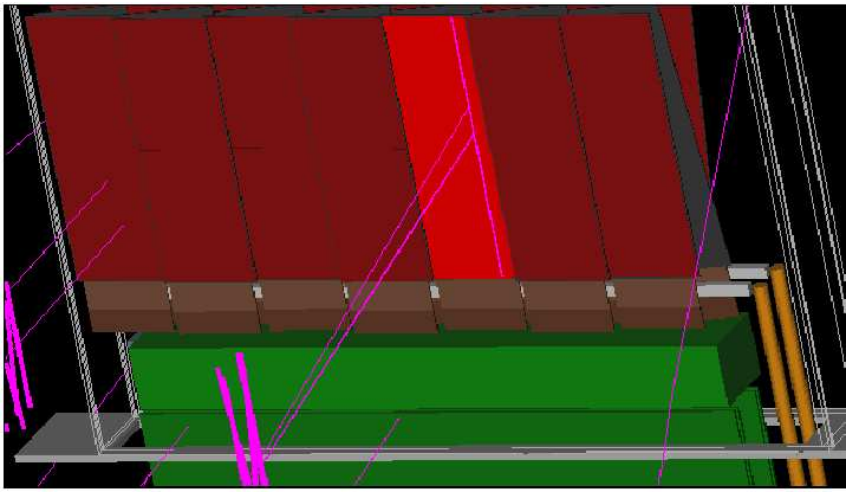


Figure 4.6: One box of IT with 4 sensors triggered by a particle with a long track. Importance of hierarchy is outlined. Triggered strips are in magenta, with the same color for tracks. The first 7 sensors of the Box are shown together with the partially visible gray and red ladders belonging to the layers behind the first layer of sensors. The front sensor with a triggered strip is highlighted with lighter red.

In the figure 4.6 we see a Panoramix[11] display of a Minimum Bias event with one of the IT-Boxes as defined in the Geometry Database of the LHCb software. Four sensors in the Box are triggered by the passing of a charge particle through the silicon. One sensor in the plot is outlined in lighter red, and the triggered strip in magenta. In the foreground there are additional clusters of strips with the corresponding sensors not displayed. The outer Tracker track segments are shown in the background, too. This figure gives a felling of how important the geometrical hierarchy is in IT.

Assuming the Object to be aligned is the Box from 4.6, then it is required to provide an exact description of the sensor structure inside the IT-Box. We do this by choosing to define the sensor position relative to the center of the object to be aligned, in the case at hand the geometrical box' center as defined in the LHCb Geometry Database. Form there we extract the sensor relative positions, too. Any translations or rotations of the box translate in a set of translations and rotations of the component sensors. The sensor surface, as defined in the figure 4.5, is parametrized in terms of:

[11] LHCb Detector and Event Graphical Display Package, it can interface to most analysis, geometry, reconstruction, simulation, and digitization packages

a) sensor center $\mathbf{R}_c = (X_c, Y_c, Z_c)$ relative to the center of the object to be aligned $\mathbf{R}_p = (X_p, Y_p, Z_p)$;
b) surface-defining versors[12]: $\vec{\xi} = (\xi_x, \xi_y, \xi_z)$, $\vec{\tau} = (\tau_x, \tau_y, \tau_z)$.
Any point on the surface of the sensor is defined :

$$\mathbf{R}_{surface} = \vec{\xi} \cdot \xi + \vec{\tau} \cdot \tau + \mathbf{R}_c + \mathbf{R}_p \tag{4.4}$$

with the ξ and τ as variables of the parametrization.

With the two points, one on track and one on surface, being equal $\mathbf{R}_{surface} = \mathbf{R}_{track}$, the problems is transformed in a system of three equations. Dropping the index i from the track parameter vector, and referring to a generic track with a measurement on a sensor shifted and rotated by $\mathbf{a} = (\delta\tau, \delta z, \delta\alpha, \delta\beta, \delta\gamma)$ with respect to $\mathbf{R}_p$ alignable object position:

$$\left\{ \begin{array}{rcll} \tau_x^r(\mathbf{a}) \cdot \tau + \xi_x^r(\mathbf{a}) \cdot \xi + X_c^r(\mathbf{a}) + X_p & = & t_x \cdot z + x_0 & \\ \tau_y^r(\mathbf{a}) \cdot \tau + \xi_y^r(\mathbf{a}) \cdot \xi + Y_c^r(\mathbf{a}) + Y_p & = & t_y \cdot z + y_0 & \Longleftrightarrow \\ \tau_z^r(\mathbf{a}) \cdot \tau + \xi_z^r(\mathbf{a}) \cdot \xi + Z_c^r(\mathbf{a}) + Z_p & = & z & \end{array} \right. \tag{4.5}$$

$$\mathbf{A}^r(\mathbf{a}, \mathbf{t}) \times \begin{pmatrix} \tau \\ \xi \\ z \end{pmatrix} = \mathbf{R}_c^r(\mathbf{a}) + \mathbf{R}_p - \mathbf{O}_{origin}(x_0, y_0)$$

The "r" index on the members of $\vec{\xi}$, $\vec{\tau}$ and $\mathbf{R}_c$ stands for "real" surface orientation and position. Opposed to the "nominal" position and surface orientation, these quantities depend on the to-be-determined alignment parameters $\mathbf{a}$. The inversion of the 4.5 provides a set of three quantities (τ, ξ, z) as function of both track parameters and alignment parameters. $\mathbf{R}_p$ is assumed in the "Nominal" geometry.

If we choose $\vec{\xi}$ to be in the "true" direction of the strips, and the complementary **direction $\vec{\tau}$ as the measurement direction**, then, from the three previous quantities only the first one is relevant, being in fact the expression of the Alignment Model, function of the track and alignment parameters. The only thing left to define is the set of coordinate transformations from a "nominal" to a misaligned-real geometry.

In the following equations and paragraphs I use the basic assumption that the solutions to alignment parameters span a range which is much smaller than other detector-geometry quantities. E.g. the rotations $\delta\alpha$, $\delta\beta$, and $\delta\gamma$ are considered to be much smaller than 5^0 stereo angle, and the shifts $\delta\tau$ and δz much smaller than the dimensions of sensors and their spacing along the beam. Hence, the transformations of sensors under misalignment could be considered infinitesimal. A direct result is that the misalignment rotation matrix has the form:

$$\mathbf{\Theta} = \begin{pmatrix} 1 & -\delta\alpha & \delta\beta \\ \delta\alpha & 1 & -\delta\gamma \\ -\delta\beta & \delta\gamma & 1 \end{pmatrix} \tag{4.6}$$

One added advantage of the infinitesimal rotations is that we no longer have to fret about ordering them.

To map the "nominal" to real geometry, the next set of relations are considered:

[12] Versors are vectors with unit magnitude.

$$\vec{\tau}^{\,r} = \mathbf{\Theta}\,\vec{\tau} = (-\delta\alpha\tau_y + \delta\beta\tau_z + \tau_x\,,\ \delta\alpha\tau_x - \delta\gamma\tau_z + \tau_y\,,\ -\delta\beta\tau_x + \delta\gamma\tau_y + \tau_z)^T \tag{4.7}$$

$$\vec{\xi}^{\,r} = \mathbf{\Theta}\,\vec{\xi} = (-\delta\alpha\xi_y + \delta\beta\xi_z + \xi_x\,,\ \delta\alpha\xi_x - \delta\gamma\xi_z + \xi_y\,,\ -\delta\beta\xi_x + \delta\gamma\xi_y + \xi_z)^T \tag{4.8}$$

$$\begin{aligned}\mathbf{R}_c^r &= \mathbf{\Theta}\,\mathbf{R}_c + \delta z\cdot(0,0,1)^T = \\ &= (-\delta\alpha Y_c + \delta\beta Z_c + X_c\,,\ \delta\alpha X_c - \delta\gamma Z_c + Y_c\,,\ \delta z - \delta\beta X_c + \delta\gamma Y_c + Z_c)^T\end{aligned} \tag{4.9}$$

where $\vec{\xi}$, $\vec{\tau}$, $\mathbf{R}_c$ quantities are defined in "Nominal" geometry, and are equivalent to the same set of quantities in the "r-real" geometry with null misalignments. The Nominal Geometry is the assumed geometry of the detector before the alignment[13], and thus any quantities defined in this geometry are assumed known and constant . An additional transformation, the δz shift in the z direction, was added in 4.9.

The relations 4.7, 4.8, and 4.9, together with the equation 4.5 allow to extract the measurement-alignment model function $\tau\,(x_0, t_x, y_0, t_y, \delta\tau, \delta z, \delta\alpha, \delta\beta, \delta\gamma)$ with the versor $\vec{\tau}^{\,r}$ in the direction of measurement. The $\delta\tau$ dependence of the model is yet to be included. To do this last step, we can notice that the τ model is defined as a measurement model in the measurement direction perpendicular to the sensor strips, which is equal to the direction of the versor $\vec{\tau}^{\,r}$. Since I define $\delta\tau$ as the misalignment-shift in the measurement direction, it means the quantity $\delta\tau$ is just added to the expression extracted from the previously mentioned set of equations.

Summarizing, the final Alignment Model for a null magnetic field is given in 4.10 for 2 degrees of freedom, and later in the chapter in the section 4.5 equation 4.24 for all 5 degrees of freedom. A $U_{model}\,(\mathbf{t}, \mathbf{a})$ notation was included as a reminder that Millepede considers the model function as depending on all alignment parameters and not only on those belonging to a certain sensor to which a measurement is associated.

$$\begin{aligned}&\tau\,(x_0, t_x, y_0, t_y, \delta\tau, \delta z, \delta\alpha, \delta\beta, \delta\gamma) = U_{model}(\mathbf{t}, \mathbf{a}) = \\ &\delta\tau + \delta\alpha\Bigg\{(\tau_x\xi_z t_x + \tau_y\xi_z t_y - \tau_z\xi_x t_x - \tau_z\xi_y t_y)\times \\ &\Bigg[\frac{\xi_x t_y Z_c + \xi_x t_y Z_p + \xi_x y_0 - \xi_x Y_c - \xi_x Y_p - \xi_y t_x Z_c - \xi_y t_x Z_p - \xi_y x_0 + \xi_y X_c + \xi_y X_p - \xi_z t_x y_0}{(-\tau_x\xi_y + \tau_x\xi_z t_y + \tau_y\xi_x - \tau_y\xi_z t_x - \tau_z\xi_x t_y + \tau_z\xi_y t_x)^2} + \\ &\frac{\xi_z t_x Y_c + \xi_z t_x Y_p + \xi_z t_y x_0 - \xi_z t_y X_c - \xi_z t_y X_p}{(-\tau_x\xi_y + \tau_x\xi_z t_y + \tau_y\xi_x - \tau_y\xi_z t_x - \tau_z\xi_x t_y + \tau_z\xi_y t_x)^2}\Bigg] + \\ &\frac{-\xi_x t_x Z_c - \xi_x t_x Z_p - \xi_x x_0 + \xi_x X_p - \xi_y t_y Z_c - \xi_y t_y Z_p - \xi_y y_0 + \xi_y Y_p + \xi_z t_x X_c + \xi_z t_y Y_c}{-\tau_x\xi_y + \tau_x\xi_z t_y + \tau_y\xi_x - \tau_y\xi_z t_x - \tau_z\xi_x t_y + \tau_z\xi_y t_x}\Bigg\} + \\ &\frac{\xi_x t_y Z_c + \xi_x t_y Z_p + \xi_x y_0 - \xi_x Y_c - \xi_x Y_p - \xi_y t_x Z_c - \xi_y t_x Z_p - \xi_y x_0}{-\tau_x\xi_y + \tau_x\xi_z t_y + \tau_y\xi_x - \tau_y\xi_z t_x - \tau_z\xi_x t_y + \tau_z\xi_y t_x} + \\ &\frac{\xi_y X_c + \xi_y X_p - \xi_z t_x y_0 + \xi_z t_x Y_c + \xi_z t_x Y_p + \xi_z t_y x_0 - \xi_z t_y X_c - \xi_z t_y X_p}{-\tau_x\xi_y + \tau_x\xi_z t_y + \tau_y\xi_x - \tau_y\xi_z t_x - \tau_z\xi_x t_y + \tau_z\xi_y t_x}\end{aligned} \tag{4.10}$$

The model is consistent with choice of transformations and the ordering given in the equation below.

$$\text{A general misalignment transformation: } \mathbf{T} = T_{\delta\tau}\times T_{\delta z}\times\mathbf{\Theta}\,(\delta\gamma)\times\mathbf{\Theta}\,(\delta\beta)\times\mathbf{\Theta}\,(\delta\alpha) \tag{4.11}$$

[13] It is considered that the Nominal Geometry includes the best possible geometrical knowledge on the detector setup before the alignment

The $T_{\delta\tau}$ and $T_{\delta z}$ represent translations after rotation sequence $(\delta\alpha, \delta\beta, \delta\gamma)$. The missing degree of freedom corresponding to ξ in the system of equations 4.5 is the shift in the direction parallel to the strip trajectory. The $\delta\xi$ misalignment is undetermined since it has no impact on the distributions of the measurement residuals.

The order of the transformations, 4.11, can be chosen arbitrary for a set of infinitesimal translation and rotations. However, to preserve the definition of $\delta\tau$ as the shift along the true measurement direction[14] $\vec{\tau}^{\,r}$, I keep the ordering implicitly throughout the thesis. This choice of transformation ordering also makes the calculus a little easier.

4.3 Model for a Case with Magnetic Field

It is convenient to drop again the track and measurement indices, since the Alignment Model for a fixed measurement on a track depends only on:
a) the parameters of the track to which the measurement belongs;
b) the alignment parameters for the object to which the measuring sensor is associated.
Hence the model function for a measurement j belonging to a track i can be written generically:

$$U^{ij}_{model}\left(\{\mathbf{t}_i\}_{i=1,N}\,,\,\mathbf{a}\right) \longrightarrow \tau_{model}\left(x_0, t_x, y_0, t_y, Q, \delta\tau, \delta\alpha, \delta\beta, \delta\gamma, \delta z\right) \tag{4.12}$$

where the track vector $\mathbf{t}_i^T = (x_0, t_x, y_0, t_y, Q)$, and the alignment vector is reduced to those alignment parameters that correspond to the measurement: $\mathbf{a}^T = (\delta\tau, \delta\alpha, \delta\beta, \delta\gamma, \delta z)_j$. Notice the presence of an extra track parameter Q implicitly depending on the charge, particle momentum, and magnetic field intensity along the particle trajectory.

As in the absence of the field, the Alignment Model for a given measurement on a fixed track can be extracted from the intersection of the track trajectory with the sensor surface. The magnetic field in the Inner Tracker is highly inhomogeneous, and its gradient along beam axis is high. For these reasons a simple parabolic-trajectory track was found to be insufficient, and a cubic trajectory was used instead. This somewhat complicates the system of equations 4.5. As we shall see later, instead of a 3-dimensional system of linear equations, there are 2 linear equations plus a non-linear equation of 3-rd order. The parametrized form of a cubic trajectory was taken from the expression used by one of the Pattern-Recognition Algorithms of the LHCb software, see [29].

Even though the cubic analytical model is less exact than the numerical result of a Kalman-Filter apparatus[15], it is much faster. In the hypothesis of an enough large sample of events, it is possible to eliminate any bias from the measurement residuals[16]. On a long-term, the cubic model might prove itself more useful to alignment than its Kalman counterpart, since the object of alignment is to extract the alignment parameters for which the resolution is inversely proportional to the square-root of number of tracks. There is an added reason for preferring the analytical form, given that a result based on this function shall be independent of the Kalman-Filter Tracking Method used in tracking. To my understanding, this non-correlation assumption between the tracking biases and alignment biases, might prove useful later in physical analysis[17] .

The cubic form discussed has the following expression:

$$\begin{cases} x(z) &= x_0 + t_x\,(z - z_0) + Q \times \left[(z - z_0)^2 + d \cdot (z - z_0)^3\right] \\ y(z) &= y_0 + t_y\,(z - z_0) \end{cases} \tag{4.13}$$

[14]The direction on the sensor surface which is perpendicular to Inner Tracker strips in the "real" geometrical setup.

[15]This is the standard track fitter in LHCb

[16]It might happen, as we shall see later, that a unit surface of the IT surface receives on average much more positive or negative particles than the opposite-charge particles, then the measurement residuals are biased in a preferential direction due to the deviation of the true track trajectory from the assumed cubic form

[17]There are opposing views within the alignment group on this subject, but the objections refer more to the Outer Tracker drift time calibration than to the IT-strip-cluster-based measurements.

The constant parameters are: $z_0 = 8520$ mm as the middle of the second IT-station, and $d = 0.377 \times 10^{-3}\text{mm}^{-1}$. The track expression of the equations 4.13 is a parametrized cubic trajectory for a particle in an inhomogeneous magnetic field. This trajectory intersected with the rectangular surface of the sensor gives a system of equations similar to 4.5, but this time non-linear in (τ, ξ, z):

$$\tau_x^r \tau + \xi_x^r \xi + X_c^r + X_p = d \cdot Q\,(z - z_0)^3 + Q\,(z - z_0)^2 + t_x\,(z - z_0) + x_0 \tag{4.14}$$

$$\tau_y^r \tau + \xi_y^r \xi + Y_c^r + Y_p = t_y\,(z - z_0) + y_0 \tag{4.15}$$

$$\tau_z^r \tau + \xi_z^r \xi + Z_c^r + Z_p = z \tag{4.16}$$

As in last section the $\tau\,(\mathbf{t}, \mathbf{a})$ is the measurement-alignment model function[18] that we are after. The other functions, ξ and z, are not relevant for alignment.

To solve the equation system 4.14 through 4.16 for τ, we can extract the ξ and z dependence on τ from 4.15 and 4.16 respectively, and rewrite 4.14 in terms of τ only. The resulting equation has a third order polynomial in τ, $P\,(\tau)$, and with $X_s = X_c^r + X_p$, $Y_s = Y_c^r + Y_p$, and $Z_s = Z_c^r + Z_p$ as the sensor true position in the LHCb system of coordinates, then:

$$\begin{aligned} P\,(\tau) = x_0 - X_s - \tau\tau_x^r + \frac{\xi_x^r\left(-y_0 + Y_s + t_y z_0 - t_y Z_s + \tau\tau_y^r - t_y\tau\tau_z^r\right)}{\xi_y^r - t_y\xi_z^r} + \\ \frac{t_x\left(-z_0\xi_y^r + Z_s\xi_y^r + y_0\xi_z^r - Y_s\xi_z^r - \xi_z^r\tau\tau_y^r + \xi_y^r\tau\tau_z^r\right)}{(\xi_y^r - t_y\xi_z^r)} + \\ \frac{Q(-z_0\xi_y^r + Z_s\xi_y^r + y_0\xi_z^r - Y_s\xi_z^r - \xi_z^r\tau\tau_y^r + \xi_y^r\tau\tau_z^r)^2}{(\xi_y^r - t_y\xi_z^r)^2} + \\ \frac{d\cdot Q(-z_0\xi_y^r + Z_s\xi_y^r + y_0\xi_z^r - Y_s\xi_z^r - \xi_z^r\tau\tau_y^r + \xi_y^r\tau\tau_z^r)^3}{(\xi_y^r - t_y\xi_z^r)^3} = \\ = D_0 + D_1\tau + D_2\tau^2 + D_3\tau^3 = 0 \end{aligned} \tag{4.17}$$

with $P\,(\tau) = 0$ equation that determines the form of the Model function. The general solution of 4.17 is too complicated to be of use in alignment, however the two approximation of 4.17, to first and to second order in τ, are manageable, albeit with some difficulty...

In practice the third order polynomial can be easily approximated with a first order polynomial or a second order polynomial, because the third order term is typically null in c++ single precision on 32-bit machines.

Both approximation were tried, and the results were used to align the Inner Tracker with 5 million high energy[19] tracks sample in the final iteration. The alignment results of the two models were compared and the observed values were found identical for the first 6 relevant digits. For this reason I shall introduce only the model based on the first order approximation:

$$\tau = -\frac{D_0}{D_1} \tag{4.18}$$

which expanded in terms of the versors components and sensor center position:

[18] The Model function has dependence: on the alignment vector $\mathbf{a}$ or more exactly on the alignment parameters $(\delta\tau, \delta z, \delta\alpha, \delta\beta, \delta\gamma)$ associated with the measurement, and on the track parameters $\mathbf{t} = (x_0, t_x, y_0, t_y, Q)$

[19] A cut was place around 20 GeV or alternatively at 5 GeV.

$$\tau = \Bigg(-x_0 + X_s - \frac{Q(\xi_y^r + 2t_y\xi_z^r)(-z_0\xi_y^r + Z_s\xi_y^r + y_0\xi_z^r - Y_s\xi_z^r)^2}{\left(\xi_y^r\right)^3} - \frac{d \cdot Q(\xi_y^r + 3t_y\xi_z^r)(-z_0\xi_y^r + Z_s\xi_y^r + y_0\xi_z^r - Y_s\xi_z^r)^3}{\left(\xi_y^r\right)^4} - \frac{(\xi_y^r + t_y\xi_z^r)((-y_0 + Y_s + t_y z_0 - t_y Z_s)\xi_x^r + t_x(-z_0\xi_y^r + Z_s\xi_y^r + y_0\xi_z^r - Y_s\xi_z^r))}{\left(\xi_y^r\right)^2} \Bigg) \times \Bigg(-\tau_x^r + \frac{2Q(\xi_y^r + 2t_y\xi_z^r)(-z_0\xi_y^r + Z_s\xi_y^r + y_0\xi_z^r - Y_s\xi_z^r)(-\xi_z^r\tau_y^r + \xi_y^r\tau_z^r)}{\left(\xi_y^r\right)^3} + \frac{3d \cdot Q(\xi_y^r + 3t_y\xi_z^r)(-z_0\xi_y^r + Z_s\xi_y^r + y_0\xi_z^r - Y_s\xi_z^r)^2(-\xi_z^r\tau_y^r + \xi_y^r\tau_z^r)}{\left(\xi_y^r\right)^4} + \frac{(\xi_y^r + t_y\xi_z^r)(\xi_x^r(\tau_y^r - t_y\tau_z^r) + t_x(-\xi_z^r\tau_y^r + \xi_y^r\tau_z^r))}{\left(\xi_y^r\right)^2} \Bigg)^{-1} \quad (4.19)$$

4.19 model form is missing the dependence on the alignment parameters for the versors $\vec{\tau}^{\,r}$, $\vec{\xi}^{\,r}$ and for the sensor center position (X_s, Y_s, Z_s). These dependences were given already in the previous section of this chapter: equations 4.7, 4.8, and 4.9, respectively[20]. Including the latter relations in the model 4.19 and upholding the transformation order 4.11, we just have to add the final term $\delta\tau$ in the Model expression. As before I interpret $\delta\tau$ as the shift in measurement direction, and hence in the direction pointed by the versor $\vec{\tau}^{\,r}$ on the sensor surface.

As in the case when the magnet is turned off, the model just deduced is farther simplified by **keeping only those terms that are of first order in alignment parameters**. This is a direct consequence of the hypothesis that the alignment parameters are considered infinitesimally small. The final Model function is not linear however since it has a complicated dependence on track parameters.

Due to the complexity of the final model form, I have to split the model in 5 terms.

$$\begin{aligned} \tau_{model}\left(x_0, t_x, y_0, t_y, Q, \delta\tau, \delta\alpha, \delta\beta, \delta\gamma, \delta z\right) &= U'_{model}\left(\delta\tau\right) + U''_{model}\left(\delta\alpha\right) + \\ &+ U'''_{model}\left(\delta\beta\right) + U'^{v}_{model}\left(\delta\gamma\right) + U^{v}_{model}\left(\delta z\right) \end{aligned} \quad (4.20)$$

The first term shall be a sum of: the part which is independent of alignment parameters and the $\delta\tau$ alignment parameter. This term is given below:

[20] These 3 equations provide the transformation from the Nominal geometry to the misaligned geometry in the LHCb system of coordinates.

$$
\begin{aligned}
U'_{model}(\delta\tau) \;=\; & \delta\tau + \Bigg(-x_0 + X_c + X_p - \frac{Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2}{\xi_y^3} - \\
& \frac{dQ(\xi_y + 3\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^3}{\xi_y^4} - \frac{1}{\xi_y^2}(\xi_y + \xi_z t_y)\times \\
& \big(t_x(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) + \xi_x(-y_0 + Y_c + Y_p + t_y z_0 - \\
& t_y(Z_c + Z_p))\big)\Bigg) \Bigg/ \Bigg(-\tau_x + \frac{(\xi_y + \xi_z t_y)((\tau_z\xi_y - \tau_y\xi_z)t_x + \xi_x(\tau_y - \tau_z t_y))}{\xi_y^2} + \\
& \frac{2(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))}{\xi_y^3} + \\
& \frac{1}{\xi_y^4} 3d(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 3\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2\Bigg)
\end{aligned}
\tag{4.21}
$$

The term $U''_{model}(\delta\alpha)$ follows at the end of the chapter in the section 4.6, and the last 3 terms are given in the appendix' section 8.2.

4.3.1 The Model Derivatives

To use the Alignment Model of the previous sections in Millepede, the model forms were implemented into C++ as functions and methods of the class "Derivatives". In accordance to the recommendations of Numerical Recipes group [26], I do not compute numerically the derivatives, but I compute from the Model function the analytical expressions for the derivatives with respect to each track parameter and each alignment parameter. These local and global derivatives implemented as C++ functions, supply the necessary values to construct the global $\mathbf{C}'$ matrix for Millepede (equation 3.15, and for the derivatives use 3.14).

4.4 Alignment Model versus Measurement Definition

4.4.1 Tracks, Clusters, Measurements

To align we need to open a potential "Pandora box", which consists of real data. For now in the absence of the real data, which would be used to align the real detector, we must use the best Monte Carlo data available. This is the case of IT[21], for OT on the other hand there are enough cosmic muons coming into the much larger acceptance area of the Outer Tracker.

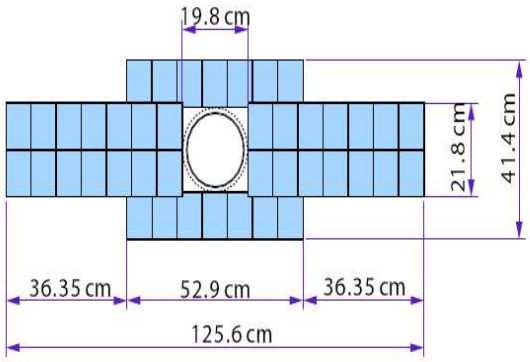
To align a Monte Carlo simulated detector, I had to use the analytical expressions of the previous section. This however gave me only the local-global derivatives, and not the measurement values themselves. To complete the χ^2-function description, I need to compute some quantities corresponding to the LHCb-detector measurables. To do the last step, I have to give a definition for an LHCb Inner Tracker measurement. This definition shall be, if not identical, at least very similar to the standard definition used by the LHCb Tracking.

The Inner Tracker is a silicon strip based detector. The sensors of IT are single-sided strip silicon sensors, meaning that for a silicon bulk of few hundred microns thickness, on one side there are equally spaced strips, as in 4.7, 4.8 or 4.4.

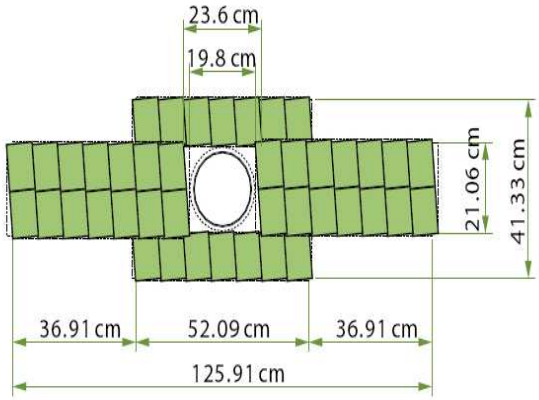
For LHCb Silicon Trackers, the residual charge deposited by a physical particle is collected by one or many strips from a sensor. This association of one or multiple strips represents an IT cluster. Each

[21] some studies were carried out by external groups, even for IT, with a few hundred IT muon tracks, but that data was not available in time for this report

cluster can be viewed as an ensemble of neighboring strips, each with a certain fraction of the total cluster-charge associated. Based on these charge factions, a global weighted position can be given for a measurement cluster. This weighted position is the weighted sum over the positions of all strips from the cluster. The procedure gives different estimated resolutions depending on how many strips are in a cluster, and what are the charge factions.



(a) IT: the $u \approx x$ plane



(b) IT: the $u \approx \cos 5^0 + y \sin 5^0$ plane

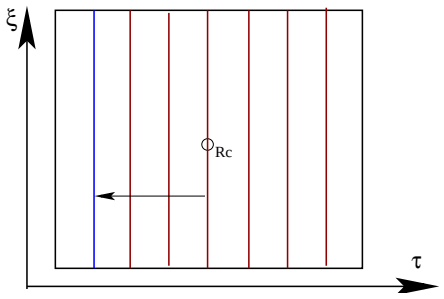


Figure 4.7: Silicon strips with $\alpha = 0^0$, the exact orientation in 3D is not a priory known.

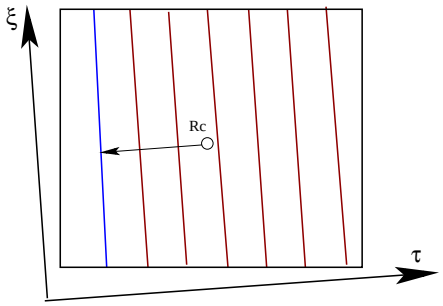
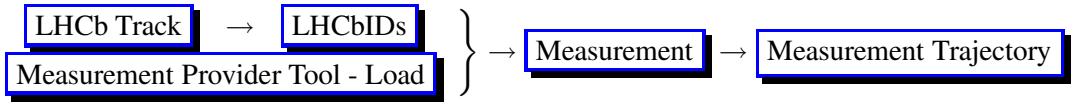


Figure 4.8: IT silicon strips with $\alpha = 5^0$, the exact orientation in 3D is not a priory known.

In case of a measurement in IT, the second thing we get from data after Track pointers, are the identity pointers of each Cluster on a track, or the LHCbIDs. These pointers are pointing in fact to a particular strip in the cluster. The software can make available from the already reconstructed data, via the Track objects, the Measurement objects associated with a given LHCbID strip.

The Measurements objects are loaded via a Measurement Provider Tool [30], which creates them on demand from the disk-stored file information. These Measurements, contain the weighted position of the IT cluster, plus the error estimation of this position. The cluster position, is available in form of a Measurement Trajectory object. The software defined trajectory gives us a direction, which is the strip direction in 3D, plus a reference position. It is these quantities that I use to define a measurement value, and the associated error estimates are the uncertainties placed on these numerical values.



Having an associated analytic trajectory for the measurement, I define the measurement value as

the distance on the sensor surface between: the Measurement Trajectory parametrized in the LHCb coordinates, and a reference trajectory that has the same direction as the Measurement Trajectory, but with the reference center of trajectory taken in the sensor center. I could have taken a different definition, and taken the reference trajectory with a different center, but the Models deduced in the previous section have this last assumption incorporated.

As long as the measurements trajectories are defined as straight lines, the upper definition is consistent with the usual measurement definition from the Tracking software. As in the figures 4.7 and 4.8, the measurement is always defined in the $\vec{\tau}$ direction and the strips trajectory give the $\vec{\xi}$ direction. These versors are the ones already used in the previous sections.

4.4.2 Geometry Database, Measurement Values and Misaligned Geometries

Each software Tool and Algorithm, which requests an information on the geometry of the detector, is communicating with what are called the DDDB and LHCBCOND databases. Avoiding discussion on some alignment-unrelated subjects like the differences between the Transient and Permanent geometries, it is sufficient to say that all the relevant information is stored in the SQL-Lite database files: "DDDB.db" and "LHCBCOND.db". In the last year a new file has been defined, the "SIMCOND.db", and it is identical in structure with the LHCBCOND.db, except that is provided for simulations, whereas the LHCBCOND Database shall be used in the future only for reconstructing real data[22].

The DDDB - i.e., the geometry database - is containing all the structure, material, and geometry information on the LHCb detector. Its information is used in Simulation, Digitization and Reconstruction phases. LHCBCOND/SIMCOND, called and the Conditions Database, is structured to contain the Alignment, Calibration, Channel, and Readout information. The alignment results shall be stored in the LHCBCOND database and used later for tracking.

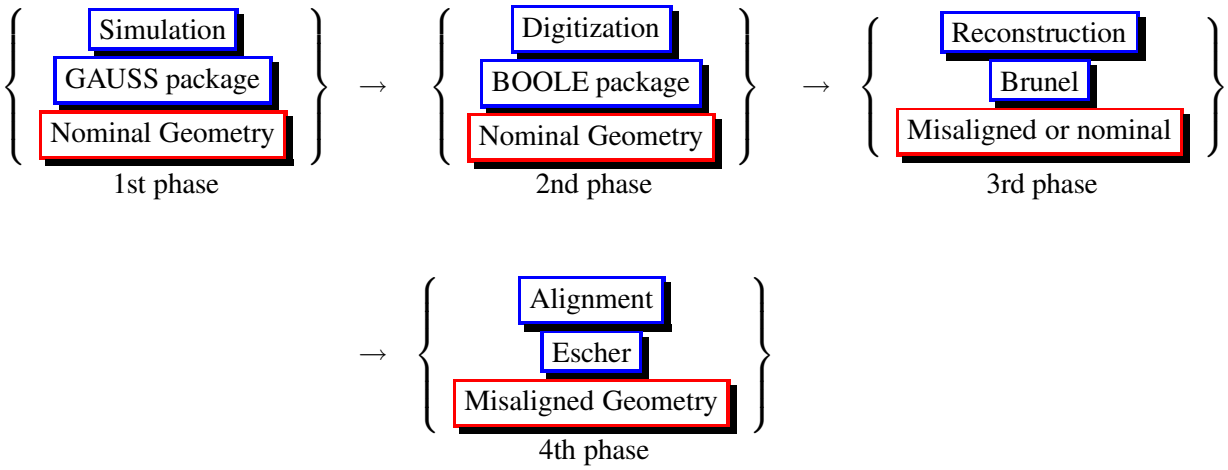
What we called the Nominal Geometry is the geometry before alignment. Providing, a reconstructed geometry that includes the alignment results shall be a final step in the alignment procedure.

To estimate the effectiveness of our alignment procedure, a deliberately misaligned LHCBCOND/SIMCOND is created. These databases rely on xml ASCII files that code all the geometry-structural information. These files are loaded in the DDDB and LHCBCOND/SIMCOND SQL-Lite files. The misalignments can be inputed at level of the xml files, in ASCII form, or directly in the SQL-Lite files. Unfortunately, the procedure evolved from a relatively straight-forward method which I implemented, to a much more convoluted approach, due to the evolution of the database services and geometry description. To keep up with the changes in the LHCb software, the tools of generating a misaligned geometry in LHCBCOND/SIMCOND had to change in time quite rapidly. This required several times input from co-workers [31], [32].

Normally, to test correctly the alignment procedure on misaligned geometries, all the initial phases of Simulation and Digitization should have used misaligned geometries, and the Reconstruction phase the unique Nominal Geometry. Yet, this is not possible. It is very difficult, to create geometrical setups, where some of the detector volumes do not overlap. This is why the LHCb Alignment group does not use misaligned geometries in the initial phases of Simulation and Digitization, and for the same reasons none of the LHC big experiment groups do not either. At the simulation phase, where the Monte Carlo particles are propagated in the detector, it might happen that for a given particle the propagation step would lead it in a region of the detector where two detector volumes overlap. If this occurs, the program might crush or at least the particle is dropped altogether. It is exactly to avoid this fact that we keep the Nominal Geometry valid for Simulation and implicitly for Digitization. **Only at the Reconstruction-Alignment phase we use a misaligned geometry**. The scheme below gives: the Monte Carlo phases,

[22] There would be in addition the ONLINE database, but this is for the online processing

their corresponding LHCb software packages, and the used geometry at each step.



In the final 4th phase, the alignment phase, I am using the misaligned geometry, unless I want to check that I recover the Nominal Geometry in the case of no misalignments. The 3rd and 4th phases are both reconstruction phases, since both use tracking algorithms although different ones.

Usually to avoid the effect of the misalignments on the Pattern Recognition, the Reconstruction was done with the Nominal Geometry. This allowed to disentangle the misalignment and Pattern-Recognition effects from the "pure" alignment effects. The few cases in which the reconstruction[23] was done with a misaligned geometry, taught us that for a small misalignment up to 1 mm, the effects coming from Pattern-Recognition do not disrupt the alignment. It is for very large misalignments, that the loss in track finding efficiency starts to play a dominant role. These extreme cases are not to be solved by Millepede, anyway. The alignment solution for large misalignments is direct alignment using measurement residual distributions and their mean as output alignment parameters. In this report I limit myself to the case where the Pattern-Recognition efficiency loss effects are not dominant.

The computed measurements, as defined in the previous subsection 4.4.1, contain encoded in the sequence of values the needed alignment information. According to what misalignment values were given in the geometry database SIMCOND/LHCBCOND, the shape of the reconstructed residual distributions will distort from the "ideal" - Nominal Geometry case - distributions. The decoding is done by Alignment via 3th chapter's method, and the alignment parameter values are used to correct the misaligned geometry database.

4.4.3 Transformation from the Detector Local Misalignments to the LHCb Global Misalignments Values

The LHCBCOND.db stores values that correspond to a set of 6 transformation in the coordinate system of the sub-detector component, e.g., in the table 4.1(a) the transformation values apply for the Top-Box Layer V from the first Station. These are not transformation in the Global-LHCb system of coordinates. Since each detector subcomponent has its own system of coordinates, to make heads-and-tail of these values, we need to transform to the system of the LHCb coordinates or to any other global-common system.

This choice of transformation would seem strange from an alignment point of view, were all transformations are computed in a common coordinate system [24]. However, someone must keep in mind that these databases were conceived first for Geant simulations, where a parent detector volume decides where each daughter element volume is placed in space. It is this kind of requirements that explain

[23] Reconstruction usually takes a lot more time than an alignment job.

[24] This statement neglects at this time the existence of alignment global degrees of freedom and "weak" modes.

<condition classID="6" name="ITT1TopLayerV">
<paramVector name="dPosXYZ" type="double">0.122689 -0.0808714 0</paramVector>
<paramVector name="dRotXYZ" type="double">0 -0.00121675 -0.00258766</paramVector>
<paramVector name="pivotXYZ" type="double">0 0 0</paramVector>
</condition>

(a) A misaligned Geometry Conditions database line - LHCBCOND, before alignment. The Nominal Geometry has all values in LHCBCOND null. The example shows that the Top Layer V of the first station has been shifted in x and y, and rotated with respect to Z and Y local axis. The z shift was null as it was the rotation around x. These being the least sensitive degrees of freedom

<condition classID="6" name="ITT1TopLayerV">
<paramVector name="dPosXYZ" type="double"> -0.00598 -0.0698 0.0675 </paramVector>
<paramVector name="dRotXYZ" type="double"> 0.000462 8.96e-05 3.2e-05 </paramVector>
<paramVector name="pivotXYZ" type="double">0 0 0</paramVector>
</condition>

(b) A misaligned Geometry Conditions database line - LHCBCOND, after alignment. Same setup as the one in 4.1(a). Alignment was done in all 5 Degrees of freedom with relatively few tracks, 6 μm residue misalignment for the x shift

Table 4.1: Two tables with LHCBCOND lines, exact rendering

	δx (mm)	δy(mm)	δz(mm)	$\delta\alpha$(rad)	$\delta\beta$ (rad)	$\delta\gamma$ (rad)
Before	0.122689	-0.0808714	0	-0.00258766	-0.00121675	0
After	-0.00598	-0.0698	0.0675	3.2×10^{-5}	8.96×10^{-5}	0.000462

Table 4.2: Summary of the values given in 4.1(a) and 4.1(b) for before and after alignment, respectively; relevant for the local coordinate system of the detector

	$\delta\tau$ (mm)	δz(mm)	$\delta\alpha$(rad)	$\delta\beta$ (rad)	$\delta\gamma$ (rad)
True	-0.129159	0.0002912	-0.002583	0.0012260	3.157×10^{-19}
Reconstructed	-0.129270	-0.064442	-0.002611	0.00131	0.00066

Table 4.3: Summary of the values given in 4.1(a), and 4.1(b) converted in the LHCb coordinate system, reconstructed and true this time.

the type of transformations that govern the LHCb geometry as it is given in the DDDB and LHCB-COND/SIMCOND databases.

Tables 4.1(a) and 4.1(b) display the result of an alignment for 5 degrees of freedom per detector-component. The 4.1(a) gives the sub-detector position before alignment. The first six numbers are in order: shift in x, shift in y, shift in z, rotation with respect to the x axis, rotation with respect to the y axis, rotation with respect to the z axis. The last 3 numbers are dummies.

In the table 4.1(b), the numbers were updated with the alignment corrections. The δx shift has a residual bias of about 6 μm, from the initial value of about 122 μm. To update the LHCBCOND database, the Millepede values, which are in a global detector frame, were converted to the local values associated with the coordinate system of each sub-detector.

In what concerns the other degrees of freedom, the precision found was very good for rotations around z and y, which are aligned to a precision of some tens of micro-rads. The alignment of the two other degrees of freedom - the shift in z (δz), and the rotation around x ($\delta\gamma$) - have lesser precision, due to the alignment insensitivity to this type of coordinates. Plus, we can expect that the resolutions for z-shifts and x-rotations will be determined by the fluctuations in the other degrees of freedom. The last

effect is dictated by the correlations between all 5 degrees of freedom, and the non-linear aspects of the problem. As I have already mentioned in the previous chapter, this two extra coordinates are very hard to fix in an alignment with tracks originating in the primary vertex[25]. Usually it is better to just fix δz and $\delta\gamma$ a priory, unless we acquire a very good sample of tracks with low random scattering and of very high energy.

δy is more an arbitrary value, than a real alignment value. Since the shift along a strip is undetermined, the last value is more a projection of $\delta\tau$ onto the y axis. As always, $\delta\tau$ is the alignment in direction of measurement.

The additional tables, 4.2 and 4.3, contain explicitly the misalignment values for the same sub-detector. It is possible to compare the values in one system of reference, local for 4.2, with the values in the other, the global reference system for 4.3.

In defining the set of coordinate transformation in Alignment I have used the following conventions:
a) the transformations are applied in a common system of all IT sensors;
b) rotations are performed first, and then translations;
c) rotations are done with the rotation pivot in the center of the sub-detector which must be aligned;
d) Roll-Pitch-Yaw system of coordinates assumed, and implicitly the same ordering and definitions, ZYX order of rotation[26].

Let us define the following quantities: an infinitesimal translation vector $\delta\vec{T}$ in the local sub-detector system, an infinitesimally small rotation matrix $\delta\mathbf{R}$ in the same system, an infinitesimal translation vector $\vec{\delta}$ in the global LHCb system, and a rotation matrix δD made of infinitesimally small rotational alignment parameters, respectively. The position of a sub-detector in LHCb is determined by the transformation between local-subdetector system to the LHCb Global system. This last transformation can be expressed in terms of $\mathbf{R}$ rotation matrix and $\vec{T}$ transformation vector, this time these are far from infinitesimally small, e.g., z shifts have several m. For the last transformation, the order of rotations and translations is reversed. This is because the transformation discussed is an inverse transformation with respect to the standard LHCb Geometry transformation from Global to Local.

Finally I was able to put everything together in the matrix relations 4.22, which gives the transformation rules from local values to the global LHCb values - the values that Millepede reconstructs. 4.23 gives the same relations but reversed. We can use 4.23 relations to convert the alignment values to the local values, and use the latter to change the LHCBCOND database.

$$\begin{cases} \vec{\delta} &= -\mathbf{R}\delta\vec{T} \\ \delta\mathbf{D} &= \mathbf{R}\delta\mathbf{R}^{-1}\mathbf{R}^{-1} \end{cases} \tag{4.22}$$

$$\begin{cases} \delta\vec{T} &= -\mathbf{R}^{-1}\vec{\delta} \\ \delta\mathbf{R} &= \mathbf{R}^{-1}\left(\delta\mathbf{D}\right)^{-1}\mathbf{R} \end{cases} \tag{4.23}$$

[25] In alignment the two degrees of freedom cause near-singular behavior of the $\mathbf{C}'$-alignment matrix.

[26] as explained in the previous sections, I choose these conventions to make separation of variables easier in the model function

4.5 Formulae B-off

Full model formula for no magnetic field case:

$$\tau\left(x_0,t_x,y_0,t_y,\delta\tau,\delta z,\delta\alpha,\delta\beta,\delta\gamma\right)=U_{model}(\mathbf{t},\mathbf{a})=$$

$$\delta\tau+\delta\alpha\Bigg\{(\tau_x\xi_z t_x+\tau_y\xi_z t_y-\tau_z\xi_x t_x-\tau_z\xi_y t_y)\times$$

$$\Bigg[\frac{\xi_x t_y Z_c+\xi_x t_y Z_p+\xi_x y_0-\xi_x Y_c-\xi_x Y_p-\xi_y t_x Z_c-\xi_y t_x Z_p-\xi_y x_0+\xi_y X_c+\xi_y X_p-\xi_z t_x y_0}{(-\tau_x\xi_y+\tau_x\xi_z t_y+\tau_y\xi_x-\tau_y\xi_z t_x-\tau_z\xi_x t_y+\tau_z\xi_y t_x)^2}+$$

$$\frac{\xi_z t_x Y_c+\xi_z t_x Y_p+\xi_z t_y x_0-\xi_z t_y X_c-\xi_z t_y X_p}{(-\tau_x\xi_y+\tau_x\xi_z t_y+\tau_y\xi_x-\tau_y\xi_z t_x-\tau_z\xi_x t_y+\tau_z\xi_y t_x)^2}\Bigg]+$$

$$\frac{-\xi_x t_x Z_c-\xi_x t_x Z_p-\xi_x x_0+\xi_x X_p-\xi_y t_y Z_c-\xi_y t_y Z_p-\xi_y y_0+\xi_y Y_p+\xi_z t_x X_c+\xi_z t_y Y_c}{-\tau_x\xi_y+\tau_x\xi_z t_y+\tau_y\xi_x-\tau_y\xi_z t_x-\tau_z\xi_x t_y+\tau_z\xi_y t_x}\Bigg\}+$$

$$\frac{\xi_x t_y Z_c+\xi_x t_y Z_p+\xi_x y_0-\xi_x Y_c-\xi_x Y_p-\xi_y t_x Z_c-\xi_y t_x Z_p-\xi_y x_0}{-\tau_x\xi_y+\tau_x\xi_z t_y+\tau_y\xi_x-\tau_y\xi_z t_x-\tau_z\xi_x t_y+\tau_z\xi_y t_x}+$$

$$\frac{\xi_y X_c+\xi_y X_p-\xi_z t_x y_0+\xi_z t_x Y_c+\xi_z t_x Y_p+\xi_z t_y x_0-\xi_z t_y X_c-\xi_z t_y X_p}{-\tau_x\xi_y+\tau_x\xi_z t_y+\tau_y\xi_x-\tau_y\xi_z t_x-\tau_z\xi_x t_y+\tau_z\xi_y t_x}+$$

$$\delta z\frac{(\xi_x t_y-\xi_y t_x)(-\tau_x\xi_y+\tau_x\xi_z t_y+\tau_y\xi_x-\tau_y\xi_z t_x-\tau_z\xi_x t_y+\tau_z\xi_y t_x)}{(-\tau_x\xi_y+\tau_x\xi_z t_y+\tau_y\xi_x-\tau_y\xi_z t_x-\tau_z\xi_x t_y+\tau_z\xi_y t_x)^2}+$$

$$\delta\beta\left\{(\tau_x\xi_y t_x-\tau_y\xi_x t_x-\tau_y\xi_z+\tau_z\xi_y)\times\right.$$

$$(\xi_x t_y Z_c+\xi_x t_y Z_p+\xi_x y_0-\xi_x Y_c-\xi_x Y_p-\xi_y t_x Z_c-\xi_y t_x Z_p-\xi_y x_0+$$

$$\xi_y X_c+\xi_y X_p-\xi_z t_x y_0+\xi_z t_x Y_c+\xi_z t_x Y_p+\xi_z t_y x_0-\xi_z t_y X_c-\xi_z t_y X_p)+$$

$$(-\tau_x\xi_y+\tau_x\xi_z t_y+\tau_y\xi_x-\tau_y\xi_z t_x-\tau_z\xi_x t_y+\tau_z\xi_y t_x)\times$$

$$\left.(\xi_x t_x y_0-\xi_x t_x Y_c-\xi_x t_x Y_p-\xi_x t_y x_0+\xi_x t_y X_p+\xi_y t_x X_c+\xi_y Z_c+\xi_z t_y Z_p+\xi_z y_0-\xi_z Y_c-\xi_z Y_p)\right\}\times$$

$$\frac{1}{(-\tau_x\xi_y+\tau_x\xi_z t_y+\tau_y\xi_x-\tau_y\xi_z t_x-\tau_z\xi_x t_y+\tau_z\xi_y t_x)^2}+$$

$$\delta\gamma\left\{(-\tau_x\xi_y+\tau_x\xi_z t_y+\tau_y\xi_x-\tau_y\xi_z t_x-\tau_z\xi_x t_y+\tau_z\xi_y t_x)\times\right.$$

$$(\xi_x t_y Y_c+\xi_x Z_c-\xi_y t_x y_0+\xi_y t_x Y_p+\xi_y t_y x_0-\xi_y t_y X_c-\xi_y t_y X_p+\xi_z t_x Z_p+\xi_z x_0-\xi_z X_c-\xi_z X_p)+$$

$$(-\tau_x\xi_y t_y-\tau_x\xi_z+\tau_y\xi_x t_y+\tau_z\xi_x)\times$$

$$(\xi_x t_y Z_c+\xi_x t_y Z_p+\xi_x y_0-\xi_x Y_c-\xi_x Y_p-\xi_y t_x Z_c-\xi_y t_x Z_p-\xi_y x_0+\xi_y X_c+$$

$$\left.\xi_y X_p-\xi_z t_x y_0+\xi_z t_x Y_c+\xi_z t_x Y_p+\xi_z t_y x_0-\xi_z t_y X_c-\xi_z t_y X_p)\right\}\times$$

$$\frac{1}{(-\tau_x\xi_y+\tau_x\xi_z t_y+\tau_y\xi_x-\tau_y\xi_z t_x-\tau_z\xi_x t_y+\tau_z\xi_y t_x)^2} \tag{4.24}$$

4.6 Formulae, B-on

The second most important part of the Model is given bellow. It misses the terms depending on $\delta\beta$, $\delta\gamma$, and δz, which are given in the appendix, see 8.2.

$$
\begin{aligned}
&U''_{model}\left(\delta\alpha\right) = \delta\alpha\times \\
&\Bigg\{-\Bigg[\Bigg(\tau_y + \frac{\xi_x((\tau_z\xi_y - \tau_y\xi_z)t_x + \xi_x(\tau_y - \tau_z t_y))}{\xi_y^2} - \frac{2\xi_x(\xi_y + \xi_z t_y)((\tau_z\xi_y - \tau_y\xi_z)t_x + \xi_x(\tau_y - \tau_z t_y))}{\xi_y^3} + \\
&\frac{(\xi_y + \xi_z t_y)(\tau_x\xi_x + (\tau_z\xi_x - \tau_x\xi_z)t_x - \xi_y(\tau_y - \tau_z t_y))}{\xi_y^2} + \\
&\frac{1}{\xi_y^3}2(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 2\xi_z t_y)(-\xi_z X_c - \xi_x z_0 + \xi_x(Z_c + Z_p)) + \\
&\frac{2\xi_x(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))}{\xi_y^3} + \\
&\frac{1}{\xi_y^3}2(\tau_z\xi_x - \tau_x\xi_z)Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) - \\
&\frac{1}{\xi_y^4}6\xi_x(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) + \\
&\frac{1}{\xi_y^4}6d(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 3\xi_z t_y)(-\xi_z X_c - \xi_x z_0 + \xi_x(Z_c + Z_p))\times \\
&(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) + \frac{3d\xi_x(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2}{\xi_y^4} + \\
&\frac{1}{\xi_y^4}3d(\tau_z\xi_x - \tau_x\xi_z)Q(\xi_y + 3\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2 - \\
&\frac{1}{\xi_y^5}12d\xi_x(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 3\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2\Bigg)\times \\
&\Bigg(-x_0 + X_c + X_p - \frac{Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2}{\xi_y^3} - \\
&\frac{dQ(\xi_y + 3\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^3}{\xi_y^4} - \\
&\frac{1}{\xi_y^2}(\xi_y + \xi_z t_y)(t_x(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) + \xi_x(-y_0 + Y_c + Y_p + t_y z_0 - t_y(Z_c + Z_p)))\Bigg)\Bigg]\Bigg/ \\
&\Bigg(-\tau_x + \frac{(\xi_y + \xi_z t_y)((\tau_z\xi_y - \tau_y\xi_z)t_x + \xi_x(\tau_y - \tau_z t_y))}{\xi_y^2} + \\
&\frac{1}{\xi_y^3}2(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) + \\
&\frac{1}{\xi_y^4}3d(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 3\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2\Bigg)^2 + \\
&\Bigg(-Y_c - \frac{1}{\xi_y^3}2Q(\xi_y + 2\xi_z t_y)(-\xi_z X_c - \xi_x z_0 + \xi_x(Z_c + Z_p))(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) - \\
&\frac{\xi_x Q(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2}{\xi_y^3} + \frac{3\xi_x Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2}{\xi_y^4} - \\
&\frac{1}{\xi_y^4}3dQ(\xi_y + 3\xi_z t_y)(-\xi_z X_c - \xi_x z_0 + \xi_x(Z_c + Z_p))(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2 - \\
&\frac{d\xi_x Q(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^3}{\xi_y^4} + \frac{4d\xi_x Q(\xi_y + 3\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^3}{\xi_y^5} -
\end{aligned}
$$

$$\frac{1}{\xi_y^2}\xi_x(t_x(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) + \xi_x(-y_0 + Y_c + Y_p + t_y z_0 - t_y(Z_c + Z_p)))+$$
$$\frac{1}{\xi_y^3}2\xi_x(\xi_y + \xi_z t_y)(t_x(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) + \xi_x(-y_0 + Y_c + Y_p + t_y z_0 - t_y(Z_c + Z_p)))-$$
$$\frac{1}{\xi_y^2}(\xi_y + \xi_z t_y)(\xi_x X_c + t_x(-\xi_z X_c - \xi_x z_0 + \xi_x(Z_c + Z_p))-$$
$$\xi_y(-y_0 + Y_c + Y_p + t_y z_0 - t_y(Z_c + Z_p)))\Bigg) \Bigg/$$
$$\Bigg(-\tau_x + \frac{(\xi_y + \xi_z t_y)((\tau_z\xi_y - \tau_y\xi_z)t_x + \xi_x(\tau_y - \tau_z t_y))}{\xi_y^2}+$$
$$\frac{1}{\xi_y^3}2(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))+$$
$$\frac{1}{\xi_y^4}3d(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 3\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2\Bigg)\Bigg\}$$

Chapter 5

Alignment Results in the Absence of the Magnetic Field

5.1 A Priori Constraints for Magnet-off

As I shall explain more extensively later in Section 5.4, there are Degrees of Freedom that can not be fixed by an IT-alignment alone. If someone is restricted to the information coming from IT measurements, he or she lacks the required information to fix global degrees of freedom, e.g., a global shift in the LHCb x coordinate of all IT sensors. In mathematical terms, these kind of transformations leave the global χ^2 unchanged. The simplest measurement model for x alignment is:

$$u_{\text{model}}\left(\text{track i}, \text{module j}\right) = x_0^i + t_x^i \cdot z^j + \delta\tau^j$$

where x_0^{i}, $\text{t}_{\text{x}}^{\text{i}}$ are the parameters for i'th track, and $\delta\tau^{\text{j}}$ is the alignment parameter for an alignment object[1]. Since for X-layers the measurement is done approximately in x direction, the alignment parameter is here: $\delta\tau^j \approx \delta x^j$.

From this very simple model, it is straightforward to already infer the existence of 2 global degrees of freedom:

$$\begin{aligned} &\text{1st mode} \quad \begin{cases} x_0^i \longrightarrow x_0^i - \delta_{shift},\ \forall\,\text{i track index, simultaneously with} \\ \delta\tau^j + \delta_{shift},\ \forall\, j \text{ align-able object with null stereo angle} \end{cases} \Longrightarrow \\ &\chi^2 = \sum_{ij} \frac{\left(u^{ij}_{\text{measured}} - u_{\text{model}}(i,j)\right)^2}{\sigma^2_{\text{measurement}}} \quad \text{Invariant under transformation} \end{aligned} \tag{5.1}$$

Transformation 1 is a global mode or global degree of freedom equivalent to a global shift in x direction of all IT sensors.

$$\begin{aligned} &\text{2nd mode} \quad \begin{cases} t_x^i \longrightarrow t_x^i - \delta_{shear},\ \forall\,\text{i track index, simultaneously with} \\ \delta\tau^j + \delta_{shear} \times z^j,\ \forall\, j \text{ align-able object with null stereo angle} \end{cases} \Longrightarrow \\ &\chi^2 = \sum_{ij} \frac{\left(u^{ij}_{\text{measured}} - u_{\text{model}}(i,j)\right)^2}{\sigma^2_{\text{measurement}}} \quad \text{Invariant} \end{aligned} \tag{5.2}$$

Transformation 2 is a shearing in x of all X-layers and leaves the global χ^2 invariant because it exists

[1] In case of IT: X-layers with null stereo angle, boxes, sensors with 0^0 stereo angle, tracking stations, etc.

a transformation in track parameters phase-space that compensates. Stressing once more, since we look for solution in the common phase-space of alignment and track parameters, it is obvious that the χ^2 is bound to be invariant under some transformations, and is up to the global-alignment to select those solutions that are the most realistic.

By extension, it is easy to see that in addition to the global shift and shearing in x there must be equivalent modes in y coordinate. Thus in the end, for shifts along measurement direction, it is required either to a priori fix 4 Degrees of Freedom, or use the mathematically-equivalent method of Lagrange multipliers [21]. Due to the stereo sequence, it is required to fix at least 2 degrees that correspond to an U or a V layer.

In addition to these global shifts, there are 3 rotational degree of freedom corresponding to 3 LHCb coordinate axis. The rotation around the z-axis, is approximately the uncertainty on the stereo angle, $\delta\alpha$. There are global rotations around LHCb x-axis and y-axis, too.

Given the model in equation 4.1, and neglecting for simplicity the stereo angle by taking $\alpha = 0$ as for an X layer:

$$U_{model} = t_x^i z^j + x_0^i + \delta\alpha_j \cdot \left(z_j\, t_y^i + y_0^i\right) \tag{5.3}$$

it is easily seen that the following mode leaves the global χ^2 unchanged:

$$\text{5th mode} \quad \begin{cases} \delta\alpha_j \longrightarrow \delta\alpha_j + \Delta \\ x_0^i \longrightarrow x_0^i - \Delta \cdot y_0^i \\ t_x^i \longrightarrow t_x^i - \Delta \cdot t_y^i \qquad \forall\,(\mathrm{i,j}) \\ y_0^i \longrightarrow y_0^i \\ t_y^i \longrightarrow t_y^i \end{cases} \tag{5.4}$$

For a shift in the z coordinate direction, there are two global modes: a global shift in z and, additionally, a scale factor in z. Again, discarding the stereo angle, the track model becomes:

$$u_{\mathrm{model}}\left(\mathrm{track\,i, module\,j}\right) = x_0^i + \left(z^j + \delta z^j\right) t_x^i + \delta\tau^j \tag{5.5}$$

and the two modes that preserve the alignment χ^2 value are;

$$\text{8th mode} \quad \begin{cases} \delta z_j \longrightarrow \delta z_j + \Delta \\ x_0^i \longrightarrow x_0^i - \Delta \cdot t_x^i \qquad \forall\,(\mathrm{i,j}) \\ t_x^i \longrightarrow t_x^i \end{cases} \tag{5.6}$$

$$\text{9th mode} \quad \begin{cases} \delta z_j \longrightarrow \delta z_j + \Delta \cdot (z_j + \delta z_j) \\ x_0^i \longrightarrow x_0^i \qquad\qquad \forall\,(\mathrm{i,j}) \\ t_x^i \longrightarrow \frac{t_x^i}{1+\Delta} \end{cases} \tag{5.7}$$

5.2 Monte Carlo Results

5.2.1 IT Quadrants, Stacks of 12 Active Layers, Track Quality Requirements

To simplify somewhat the problem of IT-alignment, as a first approach we need to decouple the problem in 4 separate alignments for each type of box. As seen in figure 5.1 The IT is split in quadrants around the beam-pipe:

1. Top side for Top Boxes,
2. Bottom side for Bottom Boxes,
3. Left side for Left Boxes, in the last year this was renamed A-Box for access;
4. Right side for Right Boxes, renamed C-Box for Cryostat.

A particle of high energy produced at the primary vertex can not cross through two sides simultaneously, unless it crosses the so called overlap regions between boxes. However the relative acceptance of the overlap regions versus the IT total acceptance is within 1 %. Thus for now we can just discard the information provided by this "overlap" tracks, and consider each quadrant to be independet.

Figure 5.1: IT Boxes around the beam-pipe

The tracks from the primary vertex connect up to 12 layers of a single IT side. The IT quadrants are from tracking point of view, 4 stacks of 12 active layers, as seen by an observer sitting in the primary vertex position.

Of course in magnetic field or if a track suffers a catastrophic collision it might occur that the particle trajectory will connect different quadrants of IT. However these tracks have low momentum and/or add to the outliers of measurement-residual distributions, i.e., they do not fulfill the quality requirements needed for alignment.

A priori, there are some straightforward criteria to distinguish between a good track and a bad track from alignment point of view. The track needs to have high momentum , enough hits in IT, and should be relatively clean, thus not a ghost or a clone[2]. The high momentum requirement, reduces the bias in the alignment results that is otherwise produced by the Molière random scattering . From the formula given in [8]: for a particle of 14 GeV the random scattering has a caracteristic $\theta_0 \approx 0.3\,\text{mrad}$ for an IT thickness of approximatelly 0.1 radiation lengths.

$$\theta_0 = \frac{13.6\,\text{MeV}}{\beta cp} z\sqrt{x/X_0}\,[1 + 0.038\ln(x/X_0)] \tag{5.8}$$

a)with $\theta_x \approx \theta_0$ the x-plane random angle scattering - a cumulative quantity after multiple Coulomb scatterings;
b) β, p , z the kinematic variables and charge of a particle;
c) x the thickness of material, X_0 radiation length in material.

This in turns give a bias RMS of around 0.3 mm in third IT station, larger than 0.198 mm pitch spacing. However, it is not only this number that is worrisome, since the dominant effect on the alignment results could be very well due not to the main random scattering distribution, but to its stochastic tail. In the latter case some outliers can in principle perturb irremediably the Newton-Raphson iterative step, making the iterative procedure more slowly convergent, or even in extreme cases[3] throwing the iterative procedure completely off-tracks and generating divergences.

5.2.2 χ^2-cut a Method of Track Quality Control, Quasi-Newton Step

In absence of a magnetic field the main factor that affects the otherwise rectilinear trajectories of high energy particles is the random scattering. This subsection outlines a method to control the effect of the Molière scattering. With the more general introduction in the method already given in subsection 3.3.1, of chapter 3, I concentrate in this subsection on the method application and description.

[2]Ghost track appears when less than 70% of hits on a track belong a single particle, clones appear when track segments belong to more than one reconstructed particle

[3]when weakly constrained degrees of freedom are taken in account, e. g. shifts in z-beam direction

In absence of field, there is no way to gauge the particle energy based on tracking alone. As seen in equation 5.8, for a highly relativistic particle, the random scattering depends inversely proportional on particle momentum. Confronted with this difficulties, and not wanting to include for now directly the information from other subdetectors[4], we have to consider the individual track χ^2. For straight trajectories, track χ^2s follow a χ^2 distribution, but because of random scattering we can expect an additional smearing for each point of the χ^2 distribution. The smearing must be smaller for fast particles, and larger for slow particles. To diminish the effect on alignment values, we can include in the iterative procedure an extra logic unit, which accepts as useful for alignment only for those tracks that have a track χ^2 smaller than a cut value.

$$\chi^2_{\text{track}} = \sum_j \frac{\left[u_{\text{j measurement}} - U_{\text{model}}\left(\mathbf{t}_i, \mathbf{a}_j\right)\right]^2}{\sigma_j^2} < \chi^2_0 \tag{5.9}$$

I begin usually with a very large cut value [5], and I finish in the last cycle with a value equivalent to a 3 σ cut. Just to be clear on the last point, in the hypothetical absence of random scattering, the probability to reject a track in the last iteration would be $1 - Erf\left(3/\sqrt{2}\right) \approx 0.0027$.

The choice of intermediate cut-values is less obvious and it depends on the track sample used, number and type of alignment degrees of freedom, and required convergency speed. Till now I can not propose a universal scheme that works for all cases. It is however my conclusion that: a short cut-value sequence is preferred for samples with particles at high momentum, and a longer sequence is needed to align for samples with mostly slow particles. Since we are at this point in a magnet-off case, the possibility to distinguish between these two sample types comes from the amount of the rejected tracks in the Newton-Raphson iteration. The slow particles are in a very high fraction rejected, whereas the fast particles are less.

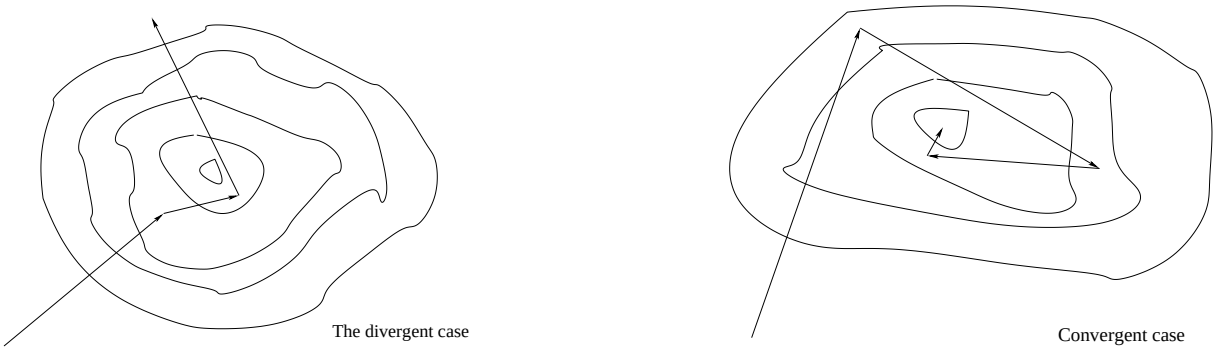


Figure 5.2: Schematic representation of the global-χ^2 map topology in the phase space of the alignment and track parameters. Left plot shows a divergent case and the right plot shows a convergent case.

The method just described, performed well for alignments that extract the most sensitive Degrees of freedom: $\delta\tau$ shifts along local measurement direction, $\delta\alpha$ rotation with respect to z axis[6] (the rotation pivot is considered in the layer or the ladder center, depending which sub-detector is to be align, IT-layer or IT-ladder), and rotation around LHCb y axis[7] (vertical cavern axis).

However, even for a sample made of 100 GeV muons, the alignment with the described method fails if the less sensitive degrees of freedom - $\delta\gamma$ rotation around x axis and z shifts - are considered. The symptoms point to an overshooting of the global minimum in the iterative procedure, this in turn leads to large fluctuation in the track-alignment parameters. The track χ^2s become considerably worse over one iteration and this increases the track rejection till it includes all track in the sample creating a divergence.

[4] The LHCb calorimeters provide energy estimates for particles

[5] the first iteration should include all but few of the track sample

[6] LHCb z axis is slightly tilted with respect to beam axis, by about 0.0036 rads.

[7] this for layers only, for ladders the rotational arm is too short to observe effects in measurement-residual distributions

The graphs in 5.2 show a schematic representation of one failed and one successful alignment. The failure occurs due to:
1. track rejection at low cut-values, e. g. track-χ^2 quality cut at 23.6 for an 8 degrees of freedom track fit.
2. large fluctuation in the alignment and track parameters close to the global minimum of the true global χ^2, fluctuation which occur over the span of one iteration.

The cure in this case was provided by the recipe described in subsection 3.3.2 of chapter 3. The adding of a monitoring logic unit as in Figure 3.3, prevents divergences. In addition, the cut value of the track-χ^2 quality cut is preserved till the convergence is achieved and then changed for a new Newton-Raphson iteration cycle. The last requirement, prevents rejection of a large number of tracks in the vicinity of global χ^2 minimum.

Rejecting the results of those iterations that increase the global χ^2 from one iteration to the next[8], especially in the vicinity of global minimum, we add quasi-Newton steps which slowly converge the alignment result to the needed solution at global minimum.

In this approach, to the alignment values of the previous iteration are added only half values of the computed alignment corrections, and the track parameters are not updated. The iterative procedure starts again, and the procedure is repeated till a smaller global-χ^2 is found. Alternatively, the track parameters are updated too in the quasi-Newton step, by adding the half values of the computed corrections. Yet, from the results obtained, I can not give a quantitative reason, for choosing between these two procedures. It is true, however, that it is the latter procedure which really chooses a solution along the Newton step direction in the phase-space of the alignment and track parameters.

An alternative to the track χ^2 quality-cut method can be implemented too. It is based on retrieving the calorimeter energy deposits for each track. This provides a way to estimate the particle energies. The corresponding momentum value is then used in the trajectory extrapolation through the IT stations, and in computing at each measurement an updated measurement error[9]. The new measurement weights go in computing the new Molière-unbiased alignment matrix that should lead to unbiased alignment values. Though this is in principle a better approach, it requires a much larger computing time. Extrapolating a track through the IT diverse material structure takes a long time. Thus given the very good results obtained with the χ^2 quality cut method - which allows alignment with millions of tracks over half day running time on one CPU core - I used the second method only for more academical reasons. If in the future it would be proven that this method is necessary, it is straightforward to implement this scheme in the current LHCb software framework.

In the long run, I think it is more important to have available the calorimeter cluster energy for a given track, and use this to reject tracks below a certain energy and to distinguish between pions and electrons by having the hadron calorimeter confirmation checked. For example. a sample of 50 to 100 GeV pions should be ideal for the alignment of even the most insensitive degree of freedom without having to worry about random scattering or electron bremsstrahlung. This simple cuts, might prove more useful then recomputing the measurement errors to account for random scattering[10].

5.2.3 Particle Gun Data, 100 GeV Muons

The particle gun is a possible particle generator of LHCb Monte Carlo software. It is the simplest generator. It is just generating particles at a certain specified place in LHCb, and it assigns to them a certain energy, momentum and direction. The newly produced particles are propagated through the LHCb detector, in a Simulation Phase. In the last chapter I considered Generation Phase and Simulation Phase under the same umbrella of "Simulation"-denomination.

[8] global χ^2 includes the measurements from all track that pass the quality cut

[9] to the measurement error is added in quadrature the estimated random scattering contribution

[10] Especially true if the stochastic tails of random scattering and electron bremsstrahlung give the dominant effect on the alignment biases

To reduce the amount of random scattering, I have planned to start with a close to "ideal" sample for initial detector alignment. This approach would allow to disentangle between alignment effects and contributions from random scattering in the measurement-residual distributions, by decreasing the latter to negligible.

With this in mind, I have generated a local mass production of about a quarter million events, each event with between 10 to 20 primary vertex muons at about 100 GeV. The resulting particles were put through a sequence of algorithms as given in section 4.4.2. To propagate the muons in LHCb, I used the Gauss LHCb software package. This step includes the generation phase, too. The results of Gauss phase are "sim" files, which among other informations, have stored the energy deposited by a particle in each crossed silicon sensor.

The energy deposit must be converted to a digital electronic signal. To accomplish the digitization step, the Boole package was used, and the triggered strip's LHCbID stored together with the charge fraction associated to a single strip from a possible Inner Tracker cluster. The output of this phase is in form of "digi"s files.

At this level an LHCb software user has access to the Inner-Tracker clusters, which allow to start a complete reconstruction phase. Without going into the full description of the last phase, it would suffice to mention that at this step, the Pattern-Recognition algorithms and the Tracking algorithms are usually used. The first set of algorithms allow extraction of a "proto"-track or a "seed", and the latter algorithms complete the reconstruction phase by providing a fully reconstructed track which gets stored in a track container of a "dst" file[11].

The "Dst" files, generated by the full Monte Carlo sequence just given, are the input of an alignment phase. The alignment phase is implemented in Escher package of the LHCb software, which consists of alignment algorithms and tools. The output of the alignment is the LHCBCOND database that includes updated positions of the LHCb sensors[12].

Except for the alignment phase, a "Nominal Geometry" was used throughout the simulation-digitization-reconstruction chain. The LHCBCOND/SIMCOND database of this geometry has null misalignments. In the alignment phase, the reconstruction starts from non-null misalignments, and it should converge[13] to null values, approaching Nominal Geometry setup with each iteration.

All these Monte Carlo phases or steps were accomplished for the muon sample. I tried to have restricted the generated muons to the maximum available geometrical phase of the Inner Tracker. Never the less, most muons were propagated through the OT geometrical acceptance, or lost in the beam pipe. Still, a large fraction amounting to 870000 tracks are IT tracks, which were reconstructed and kept by the upper scheme. The reconstructed tracks passed some final quality checks at the level of the alignment phase. The mentioned quality checks are not the χ^2-track quality cut, but e. g. cuts that reject the tracks which have less than 9 IT-hits.

IT Layer Alignment, 3 Degrees of Freedom

In the next part I have used only part of the Gun Particle track sample, of about 290000 tracks.

After 27 Newton-Raphson iteration steps, without quasi-Newton, and with track χ^2 cut changing from one iteration to the next, I got very good alignment precision and results for 3 Degrees of freedom, $\delta\alpha$, $\delta\beta$, and $\delta\tau$, respectively. In figures 5.4 and 5.5, the final Newton-Raphson results are given for the three Degrees of freedom. The alignment was done for 48 IT detector Layers (each layer has 7 single or double sensors) partitioned in 4 virtually independent quadrants. In the 5.4 and 5.5, the lower axis index runs over from 0 to 47:
0) First Station's Top Box first layer x;
1) First Station's Top Box layer U;

[11] Data Summary Tape file

[12] Actually the alignment updates refer to either Inner Tracker layers or ladders, the latter being equivalent to sensor position updates as long as the sensor and ladder possible deformations are not included in the alignment model.

[13] And it does converge to null or close to null misalignments!

2) First Station's Top Box layer V;
3) First Station's Top Box second layer X;
4-15) similar sequences for the Bottom Box, A-Box, C-Box of the 1st Station;
16-47) similar sequences as the upper for 2nd and 3rd Stations.

As seen in the Figures 5.4 and 5.5, some of the layers are fixed a priori. In this case 2 global translation, 2 shearings and 1 rotation are fixed per each quadrant. The choice is:
a) $\delta\tau = 0$ for first x layer, first u layer, last v layer, last x layer, e.g., in the Top Boxes (same for the others);
b) $\delta\alpha$ for last x layer;
c) $\delta\beta$ for last x layer.

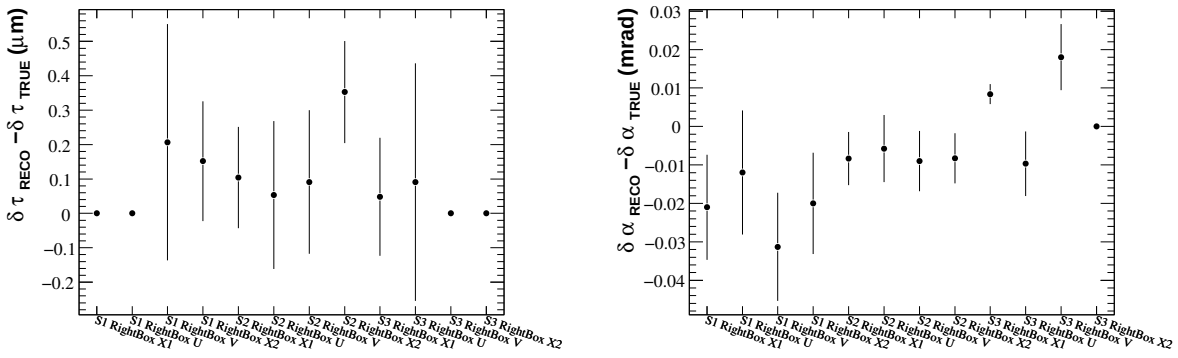
The results for the 12 layers of one quadrant are summarized in figures 5.3 a and b. As in the case of 5.4 it is easy to see that no alignment parameter $\delta\tau$ exceeds 1 μm error.

On the average, in the final iteration I get about 0.29 μm alignment precision for the alignment in direction of measurement $\delta\tau$. The corresponding values for rotations are: 0.02 mrad $\delta\alpha$, and 0.094 mrad $\delta\beta$, respectively.

In Appendix' subsection 8.3.1 in figures 8.5(a) to 8.5(d) I give the error plots and quasi-pull plots. The figures 8.5(a) and 8.5(c) are not exactly pull-plots, since the displayed distributions contain all non-a priori fixed layer degrees of freedom $\delta\tau$ or $\delta\alpha$, with some of the degrees of freedom correlated. The distributions are for only one alignment run, but because the IT is split into independent quadrants, the results of these quasi-pull plots should hold at least qualitative relevance regarding the alignment effectiveness.

Indeed, the quasi-pulls approach the expected normal distribution. The Root Mean Squares of the 8.5(a) and 8.5(c) are both close to 1, pointing to a successful alignment and a correct error propagation from the measurement error level to the alignment parameter uncertainty.

To highlight the fluctuation in the alignment parameters with iteration step, I have selected 3 layers and plotted their alignment parameters $\delta\alpha$ as function of iteration in figures of 5.6. It is obvious from these plots that the χ^2-track quality cut does not make a significant impact on the alignment quality. This is to be expected, however, since the cut was designed to eliminate those particle tracks suffering from multiple scattering, and the effect of this physical process is negligible for a 100 GeV muon. In the same plots it is obvious, for both errors and their pulls, that the final results were convergent.



(a) $\delta\tau$ results

(b) $\delta\alpha$ results

Figure 5.3: Alignment results after 27 iterations, with the corresponding Station, Box, Layer index label

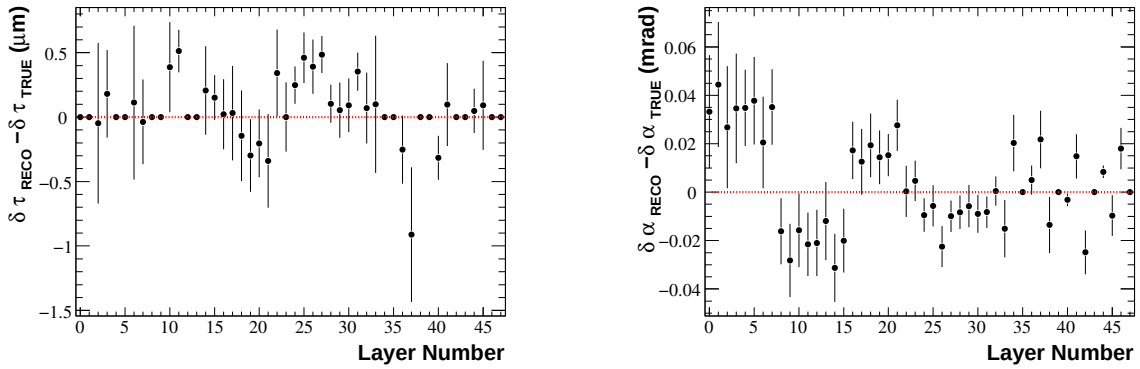


Figure 5.4: Alignment results after 27 iterations, shifts along measurement direction ($\delta\tau$) and rotation around beam axis $\delta\alpha$, error bars given by the alignment method.

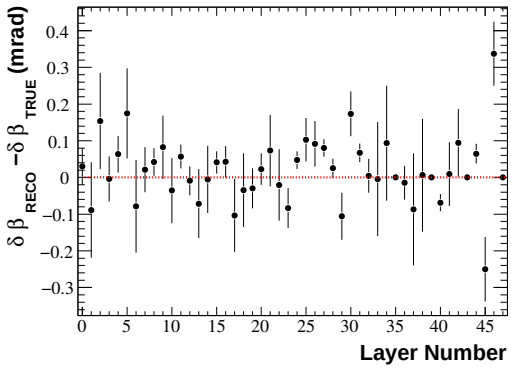


Figure 5.5: Alignment results after 27 iterations, rotation around y LHCb-vertical axis ($\delta\beta$), error bars given by the alignment method.

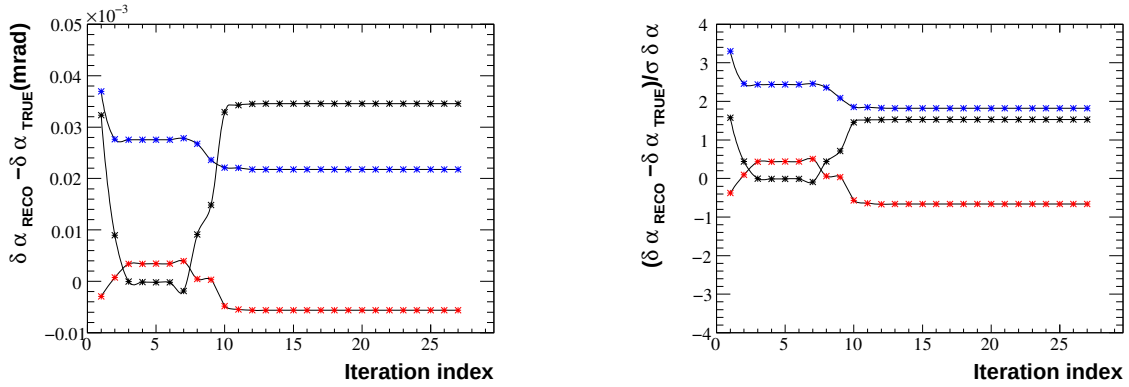


Figure 5.6: Alignment progression with iterations for rotation around beam axis, the effect of the χ^2 quality cut is seen after the 7th iteration when the bias due to random scattering is reduced and the convergence becomes evident. The plotted quantities correspond to: Station 1 Top Box Layer X2, Station 2 Left/A-Box Layer U, Station 3 Bottom Box Layer U. (rank 3, 25, 37)

The evolution of the global alignment-χ^2 with the iteration number is given in figure 5.7. On the global χ^2 plot is superimposed the evolution of the track number, which represents those tracks that pass the track-χ^2 cut. In the final iteration, the number of degrees of freedom for alignment is given in

equation 5.10:

$$\left.\begin{array}{rcl} N_{dof} & = & 2.054 \times 10^6 \\ \chi^2_{global} & = & 2.027 \times 10^6 \end{array}\right\} \Longrightarrow \frac{\chi^2}{N_{dof}} = 0.987 \quad \text{and} \quad RMS_{\chi^2} = 2030 \tag{5.10}$$

The apparent 13σ deviation from a χ^2 distribution with 2.054×10^6 degrees of freedom is obviously the effect of the track-χ^2 cut which cuts away the tails of the measurement residuals for those tracks used in alignment at each iteration. To conclude this subsection, I give in the figure 5.8 the shape of the track-

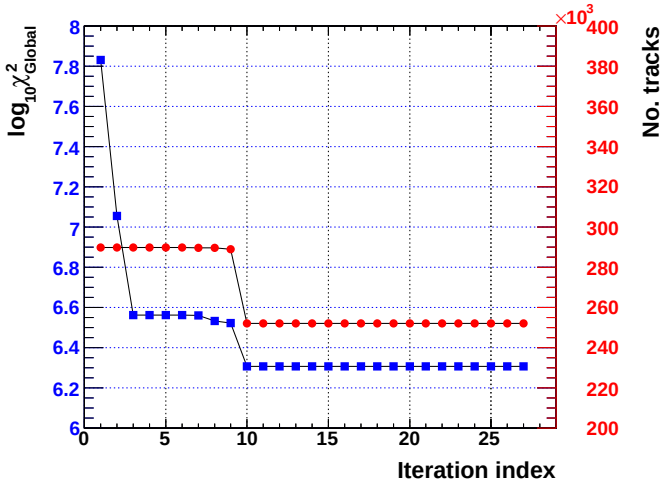
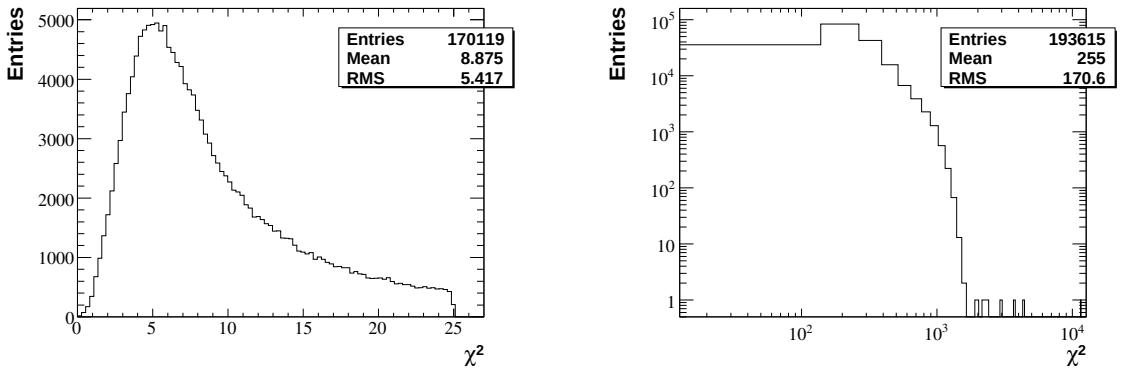


Figure 5.7: Alignment progression with iterations: global χ^2 of alignment and number of tracks used in alignment. The latter quantity changes as the quality cut on the track's χ^2 becomes harder - in the last iteration it corresponds to a 3.15 σ equivalent probability of keeping a track

χ^2 distribution before and after alignment. As apparent in the picture the distribution values are slightly smeared with respect to what we would expect, and the tail is broader. The effect of random scattering still evident in this smearing, even for this high energy track sample. An additional contribution to this smearing comes from the alignment parameters and track parameter correlations.



(a) After 27 Newton steps

(b) Before alignment

Figure 5.8: Track estimated χ^2 in the last iteration, here the 27th. Quality cut equivalent with a 3.15 σ probability rejection. Only tracks with 8 degrees of freedom per fit were included.

.

Pull Plots and Resolution Plots

The sample of approximately a quarter-million Particle Gun events was split in 34 independent subsamples. Each sample gets about 23000-27000 tracks usable in alignment. The results for 2 τ degrees of freedom are shown in plot 5.9, whereas for 3 α and 3 β are shown in 5.10 and 5.11, respectively.

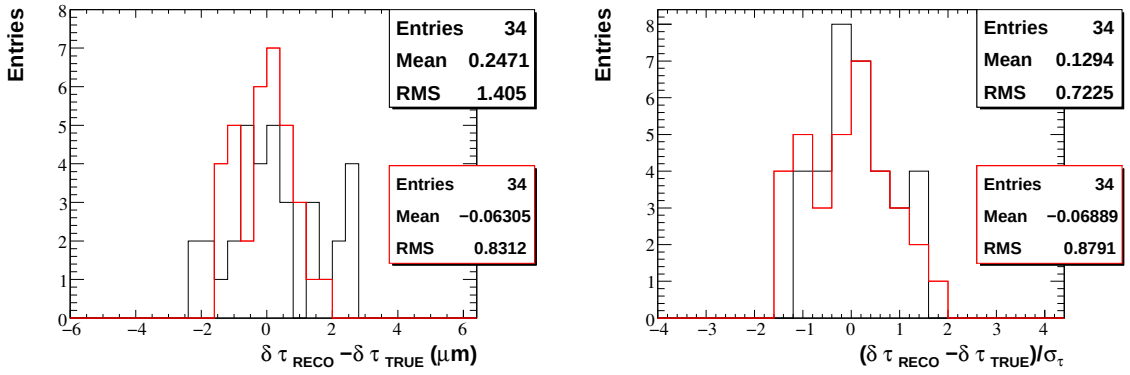


Figure 5.9: Pull plots and resolution plots of $\delta\tau$ alignment for X1 and U layers in the Top Box of the last Station

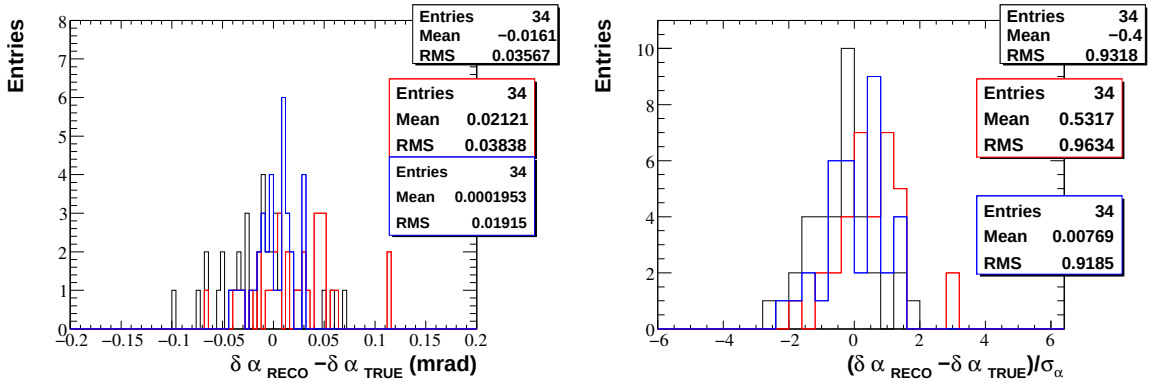


Figure 5.10: Pull plots and resolution plots of $\delta\alpha$ alignment for X1, V and U layers in the Top Box of the last Station

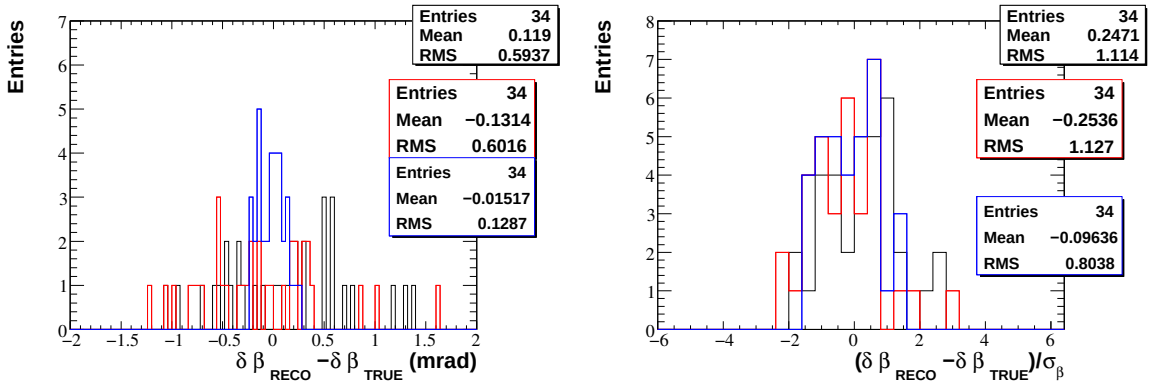


Figure 5.11: Pull plots and resolution plots of $\delta\beta$ alignment for X1, V and U layers in the Top Box of the last Station

This time the plots are the real pull-plots due to the independence between each of the 34 track samples. In the 3 last mentioned plots the histograms have 34 entries, one entry for each alignment run. The pull plots are consistent with the Normal distribution, and the associated resolution plots have RMS within 1 micron for alignment of $\delta\tau$. The $\delta\alpha$ has an RMS of about 0.04 mrad, for U and V layers, and 0.02 mrad for the X layer. The latter relatively low value can be explained by the layer closeness to the layer that is fixed in $\delta\alpha$. This layer is an X layer and is decisive in the error estimates assignment to the neighboring layers. The other layers in the first and second station get a larger RMS for the same distributions, because they have a larger lever-arm with respect to the layer which is fixed in this degree of freedom.

The agreement between the pull plots and the Normal distribution proves that the error propagation worked, and the alignment converged to the expected results. In other words, this consistency check - between the value I put in the geometry database and the alignment values that were reconstructed - was successful.

5.2.4 Two Extra Degrees of Freedom and the Quasi-Newton Steps

To prove the validity of the full model function given in section 4.5 of chapter 4, it is needed to include the last two determinable degrees of freedom: δz shift in LHCb z direction, and $\delta\gamma$ rotation around x axis. The alignment of the LHCb Inner Tracker is largely insensitive to these degrees of freedom, and the inclusion might trigger divergences. The alignment Millepede $\mathbf{C}'$ matrix has low eigenvalues coming from eigen-modes that are mixed modes of all 5 Degrees of freedom . This makes the matrix close to singular, and thus large fluctuation in the alignment parameters are to be expected during the Newton-Raphson iterative phase. The large fluctuations occur for the track parameters too, since it exists a strong correlation between the two types of parameters.

As a result of this large fluctuation, close to 3σ equivalent track-χ^2 cut, it might happen that the majority of tracks in the alignment sample get rejected. This divergency is cured by using quasi-Newton steps.

Other alignment groups use at this point regularization procedures of the $\mathbf{C}'$ matrix, or decouple the degrees of freedom in independent alignments. The mentioned approaches have intrinsically possible pitfalls. The alignment degrees of freedom belonging to the same object, are strongly correlated, and the regularization procedures prevents a correct error propagation. The alignment parameter errors become only estimative in this case.

Millepede has also its own regularization method in place. However, I did not find this procedure useful for the cases that I studied, and thus I just removed it from the algorithm via a flag set. If in the future it might be proven useful in some case, it is no problem to enable it back by changing the flag value.

Regularization of the alignment matrix, whether a Millepede matrix or not, is bound to create later-on difficulties. The regularization scheme usually removes an eigen-mode which is associated with a low eigenvalue, from the alignment solution. This practically amounts to a redefinition of the alignment parameters. The obtained results in such procedures are linear combination of the initially defined alignment parameters. There is a caveat, to the last statemets, the proponents (advocates) of this approach argue that the alignment purpose is to find that solution which minimizes the measurement residuals, and not to really find the actual misalignments. Although I completely agree to this statement, I must recommend caution in using these regularization procedures. We are doing for IT a local alignment, hence it would be useful for the global alignment to know the exact meaning of the values which are provided by the local alignment. A full global alignment might be allowed to arbitrary regularize a complete alignment matrix of the LHCb detector. But at local level we need to be more careful.

To avoid potential problems, I do an a priori fixing of the least sensitive degrees of freedom. The global alignment can take the fixed degrees of freedom and align them with respect to other parts of the LHCb Tracking system, e.g., VELO. Millepede provides an easy way to do this, which unfortunately I had no time to implement. Assuming that the Before-Magnet Trackers are already aligned, it is possible

to propagate the track state from the Before-Magnet region, to the After-Magnet region. The track states, can be used to do an individual alignment of one sensor or one layer in all degrees of freedom. This is mathematically straightforward, since the dimension of alignment matrix is no more that 5, if all degrees of freedom are included. The absence or presence of the Field does not change this procedure, except maybe for an undefined magnetic field strength scale.

Mathematically the a priori fixing of detector's degrees of freedom is equivalent with the usually preferred regularization procedure of "Singular Value decomposition"[14] (see [33]).

I have used the full track sample of about 0.87 Million tracks to align the Inner Tracker in all 5 degrees of freedom. The figures 5.12(a) to 5.12(e) display the final alignment results after an 80-iterations run. This time the quasi-Newton runs which minimized the global χ^2 are the only allowed iterative steps.

The results of iterations that have not minimized the alignment-χ^2, were discarded, and only those steps which decreased alignment-χ^2 are allowed to update the track and alignment parameters. The method is consistent with the quasi-Newton steps of chapter 3. In addition to the usually a priori fixed degrees of freedom of the section 5.1, it was needed to fix additional degrees of freedom. Namely, two δz degrees of freedom for one U and one V layer, per each quadrant. These 8 constraints in total are visible in 5.12(e). The additional constraints were not mandatory, but allowed a better error propagation.

Some "weak" eigen modes are still present in the plots 5.12(a) and 5.12(e), as seen in the larger values of the third station's δz and $\delta\tau$.

To put the 5.12's numbers in a perspective, they are the result of: a simultaneous alignment of $48 \times 5 - 44 = 196$ alignment degrees of freedom, and tracking for 0.87×4 Million track parameters. The problem is equivalent with a fit of 3.5 Million parameters. The large number is consequence that we minimize in the common phase-space of track and alignment parameters.

Quasi-Newton Convergence

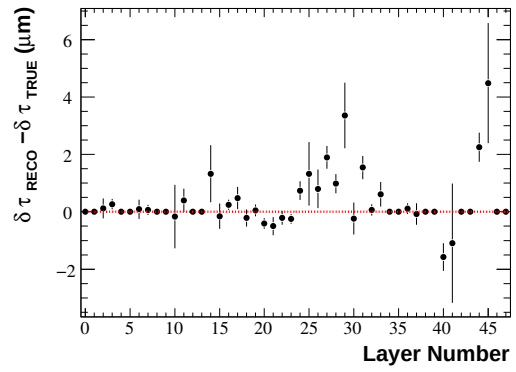
As expected for a method similar to a quasi-Newton method, the convergence is slower than the Newton-Raphson with full Newton steps. 80 iteration were required to achieve convergence even for the least sensitive degree of freedom. The total number of iterations that succeeded is smaller, though, and equal to about 20 quasi-Newton steps and 8 initial full Newton steps. The quasi-Newton steps are starting when the Newton-Raphson iteration approaches the minimum of the true global-χ^2.

During the quasi-Newton steps the track-χ^2 cut becomes harder, and it allows to reject the bad-quality trajectories. This is visible in the figure 5.13(b), where the evolution of the alignment-χ^2 is given together with the number of tracks used in alignment. The latter quantity changes as the track-χ^2 cut changes from iteration to iteration.

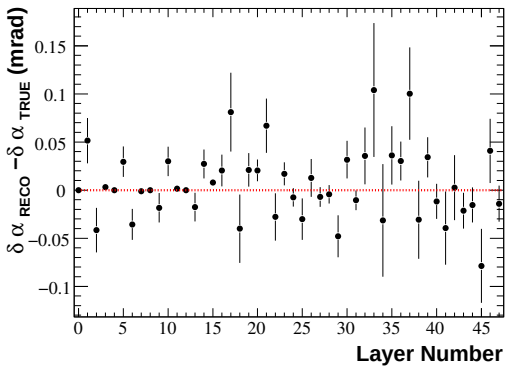
The figure 5.13(a) includes besides the successful quasi-Newton steps the steps that failed to decrease the global χ^2 and hence had the alignment correction discarded. I included this plot to outline the erratic behavior of the Newton-Raphson method without the global-χ^2 minimization criterion, and the Newton-Raphson method with successful quasi-Newton steps. From 5.13(a) and 5.13(b) it is obvious that left unchecked the full-Newton step leads to divergences. It is the robustness of the quasi-Newton method that provides the convergence to an alignment-χ^2 minimum.

To check the convergence of the last case we need to show that the alignment parameters converged too. Convergence of the alignment-χ^2 is a requirement, but is not sufficient. The changes in the alignment parameters are the "true" gauges of the convergence.

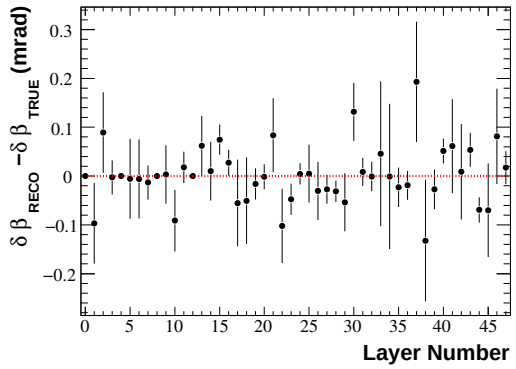
[14] Actually the regularization procedure eliminates the null space of the Singular Value Decomposition from the solution.



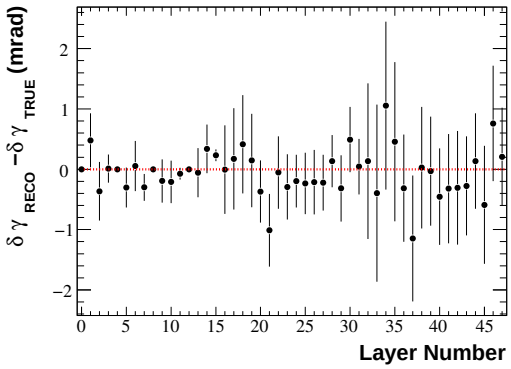
(a) $\delta\tau$ Degrees of freedom



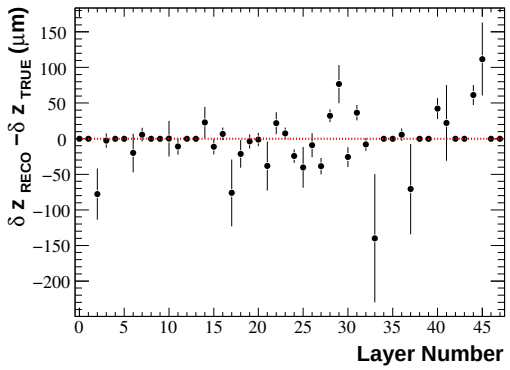
(b) $\delta\alpha$ Degrees of freedom



(c) $\delta\beta$ Degrees of freedom

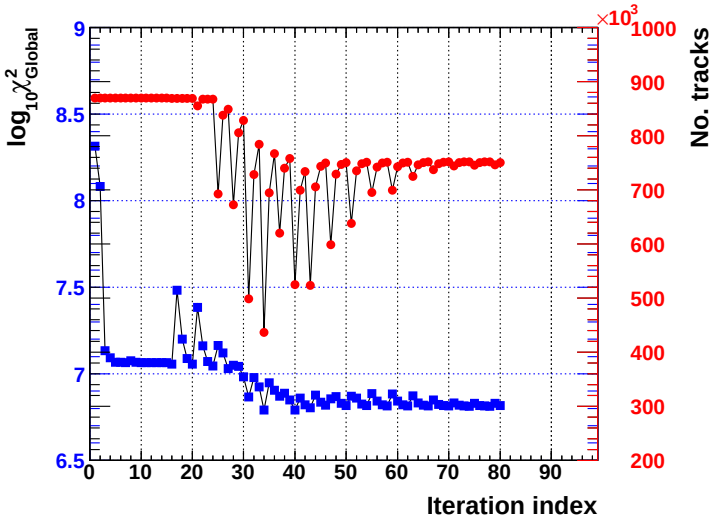


(d) $\delta\gamma$ Degrees of freedom

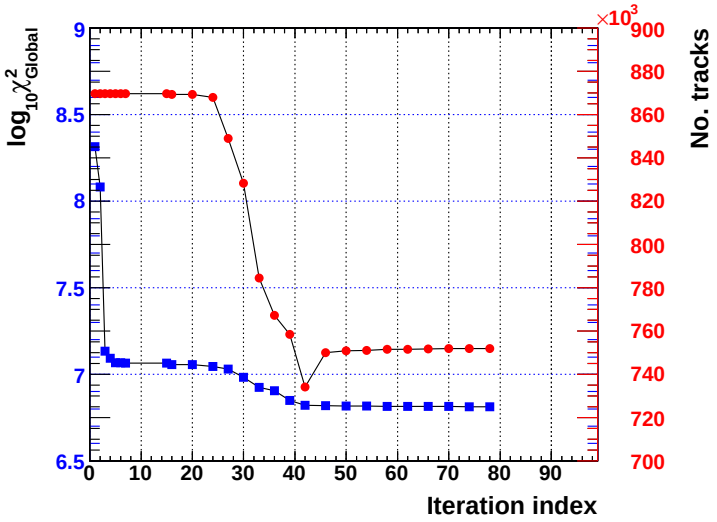


(e) δz Degrees of freedom

Figure 5.12: Resolution plots for 5 types of degrees of freedom, simultaneously aligned, with a quasi-Newton step procedure. Millepede error estimates, and alignment parameters after 77 iterations



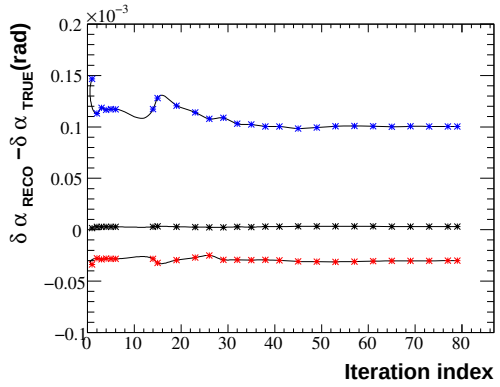
(a) Display of χ^2 evolution for all iterations



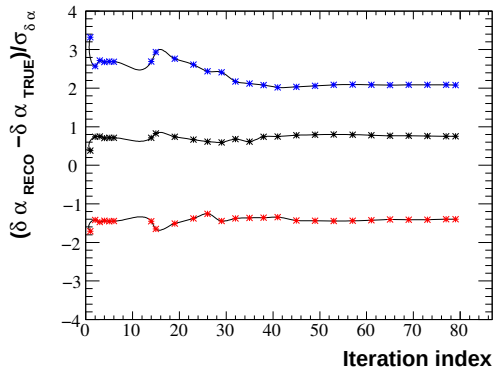
(b) Display of χ^2 evolution for successful quasi-Newton steps

Figure 5.13: Global χ^2 evolution for a Newton-Raphson iterative procedure with quasi-Newton steps

I plotted in 5.14(a) the corrections to the more sensitive degrees of freedom $\delta\alpha$. It is seen in the 5.14(b) that the error estimates were convergent, too. The plots show only the alignment corrections where the quasi-Newton step have minimized successfully the χ^2. Technically the quasi-Newton step has no error estimate from Millepede unless is a full Newton step. Yet the convergence of the error estimates allows to use an extra full-Newton step to extract the alignment precision with Millepede. This is possible as soon as the procedure converged in both alignment parameters and alignment/global-χ^2. The last full-Newton iteration computes the alignment parameters and by inverting the $\mathbf{C}'$ alignment matrix recovers the covariance matrix associated with the last alignment corrections.

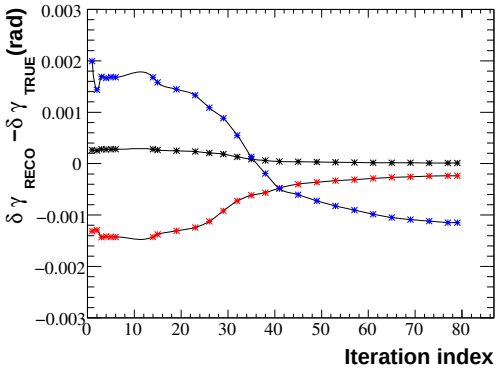


(a) $\delta\alpha$ convergence to real solution

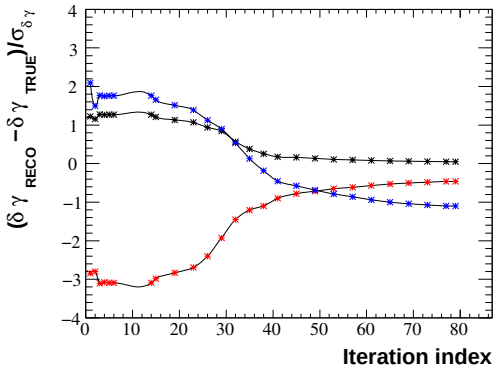


(b) $\delta\alpha$ convergence scaled by the Millepede error estimates

Figure 5.14: Convergence for 3 degrees of freedom of type $\delta\alpha$, for a Newton-Raphson iterative procedure with quasi-Newton steps. The 3 DOF were randomly chosen from the total of 48 DOF of this type



(a) $\delta\gamma$ convergence to the real solution



(b) $\delta\gamma$ convergence scaled by the Millepede error estimates

Figure 5.15: The slow convergence for 3 degrees of freedom of type $\delta\gamma$, for a Newton-Raphson iterative procedure with quasi-Newton steps.

The less sensitive degrees of freedom of type $\delta\gamma$, rotations around x axis, have a much slower convergence rate than the much more sensitive counterparts: $\delta\tau$, $\delta\alpha$, and $\delta\beta$, respectively. I highlighted the difference between convergence rates, in the contrast between figure 5.14 and figure 5.15. The behavior of the error estimates and of the alignment residuals rescaled with the Millepede errors, is the same. One extra feature of the plots is the scale difference in the error estimates of $\delta\alpha$ and the ones for $\delta\gamma$. As we achieve average resolution of 0.4 milli-rads for $\delta\gamma$, the average error for $\delta\alpha$ was found to be at about 33 micro-rads. The latter averages were taken over all layers of IT.

If desired the residual "weak" modes of the problem can be removed. This will add stability to the $\delta\tau$ error in the third station. However the alignment results are already very good. And if new tracks are added to the sample we can expect to reach the same levels of precision as the ones achieved when I was aligning for 3 Degrees of freedom only. Error propagation will suffer, though.

In the appendix, I have included plot 8.6 of the reduced global χ^2. The reduced χ^2 converges to a value close to 1.05. In the quasi-Newton steps it is actually this value that gets minimized. It is necessary

to do this, because, otherwise a large track rejection by the track-χ^2 cut would fake a successful quasi-Newton step.

In the last successful iteration, I got a reduced alignment-χ^2 equal to 1.0550, for about 6165000 degrees of freedom. This 97σ-equivalent difference from the expected value, is partially due to residual "weak" modes and to a small random scattering contribution present even for 100 GeV muons. The last disagreement does not mean the alignment was not successful, it just shows that the analytical alignment model has its limitations. In the presence of weak modes and a small random scattering it is not a surprise that the model is not perfectly fitting the data.

In conclusion, given the alignment precision obtained, I have just proved that for very high energy particle tracks, it is possible to align the detector in all the 5 determinable alignment degrees of freedom simultaneously. It is possible to produce an alignment with the same precision by taking 50-100 GeV pions in a real sample.

5.2.5 Minimum Bias 10 TeV Collisions with Open VELO, Nominal Geometry in Reconstruction

In the following subsection I have used a Minimum Bias event sample, mass-produced on the Grid at the request of the LHCb alignment group. The geometry used in producing this sample is the "Nominal" geometry, hence no misalignments, and no survey corrections[15]. Unless otherwise stated, the LHCB-COND/SIMCOND database used by the Alignment package contains non-null values as misalignments. After conversion to the LHCb common system of reference (see section 4.4.3), these values are the starting alignment parameters for the Newton-Raphson procedure. The alignment results, if convergent, should approach the null value of the Nominal Geometry.

Here, the collisions are taken at 5 TeV + 5 TeV, in the absence of magnetic field. The geometry has the VELO vertex detector in an open setup. The Open-VELO geometry, means that a large fraction of event's high energy component will be lost close to the beam. There are approximately 1 Million generated events, and I found about 3460000 tracks with at least 9 hits in IT. The tracks were retrieved from the Best Container. Best Container is the container that contains the "best" tracks, and the latter tag is set by the Tracking phase algorithms after a previous Pattern-Recognition phase. In setting this quality factor, the tracking take into account: track-χ^2, number of hits in each sub-detector, etc.

Of the upper total number only a minority is kept in the last iteration. The last effect is to be expected, given that the large majority of particles have relatively low momentum, and by consequence suffers from random scattering. In addition to the random scattering of hadronic component, it is natural to expect a leptonic component, electron and positrons. The low mass of this particles means that this component will suffer changes in track trajectory due to bremsstrahlung.

Starting from the Nominal Geometry

As first guesses of the alignment parameters I use the null values. This is "ideal" alignment already because I use the "correct" geometry from the start. The choice is justified because I want to outline in the following paragraphs the independence of the final result with respect to the starting values. To complete this case, I shall give later-on the more realistic case where the Alignment package uses as starting geometry the Misaligned Geometry, with randomly generated misalignments .

The results of alignment for two types of alignment parameters, $\delta\tau$ and $\delta\alpha$, are given in figures 5.16. There are additional plot in the appendix too, subsection 8.3.2. However, these are given to provide points of comparation with the results obtained in the next paragraphs where a misaligned geometry was used.

[15] An optical alignment survey of the LHCb Inner Tracker sensors, was done in the past, yet this information has been only recently made available in an useful form.

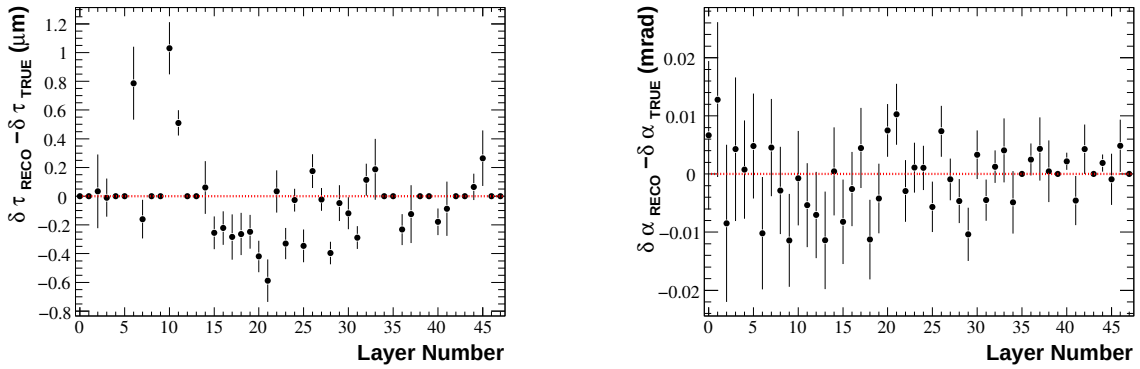


Figure 5.16: Alignment results after 21 iterations, shifts along measurement direction ($\delta\tau$) and rotation around beam axis $\delta\alpha$, error bars given by the alignment method. An initial "perfect" geometry assumed.

A Misaligned Geometry in the Alignment Phase

As before I align for two types of degrees of freedom, $\delta\alpha$ and $\delta\tau$. This time the starting value for the alignment parameters are null too, but in the Geometry Condition Database LHCBCOND/SIMCOND, misalignments were included, and hence the starting geometry is different from the Nominal. The values were randomly generated with Gaussian distributions. The measurement shifts $\delta\tau$ were produced with Gaussians of 1 mm RMS. The $\delta\alpha$'s are smeared consistently with a Gaussian of 1 mrad width.

To see the dependence on the initial value for the alignment results, I am aligning now starting form this misaligned geometry. The final results should not depend significantly on the starting alignment values, unless the Newton steps ended in a local minimum of the global-χ^2, fact disproved by the results in this case.

Comparing the plots 5.17 with 5.16, it is possible to conclude that except for some small differences observed in final results, the alignment parameters are reconstructed within the same range of values. This proves that we have not ended in some local minimum, and that the iterations converged to a set of values that are close to the true minimum of the alignment global-χ^2.

The similarities between the $\delta\tau$ plots confirm that the biases observed are not non-linear effects, but most probably are due to random scattering.

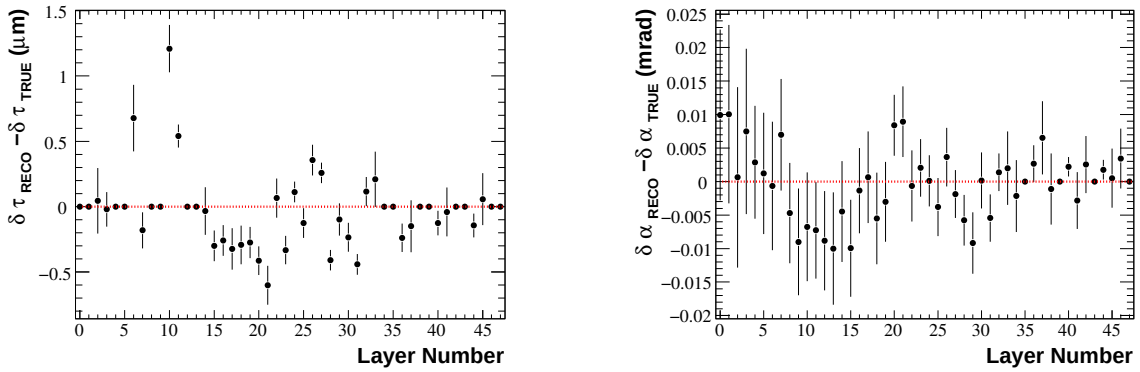


Figure 5.17: Alignment results after 21 iterations, shifts along measurement direction ($\delta\tau$) and rotation around beam axis ($\delta\alpha$), error bars given by the alignment method. Alignment starts with a misaligned geometry.

To decrease the bias of the measurement residuals, it is necessary to use the track-χ^2 quality cuts during the Newton-Raphson iterations. This reduces typically the number of tracks to about 1 million in the last iteration. These are the tracks that suffer the least from changes in particle trajectory. Although

this method allows to approach more realistic error estimates for alignment parameters, it is expected that biases will still be present, as I accept all tracks bellow a certain value in track-χ^2. The resolution plots 5.17 of this section and the quasi-pull plots in the appendix 8.11(a) to 8.11(d), show that error propagation for $\delta\alpha$ degree of freedom is correct, whereas the error propagation $\delta\tau$ is still slightly biased from the random scattering. The errors of $\delta\tau$s are small enough at 0.35 μm on the average, and each error is under 1.2 μm, see 8.11(b).

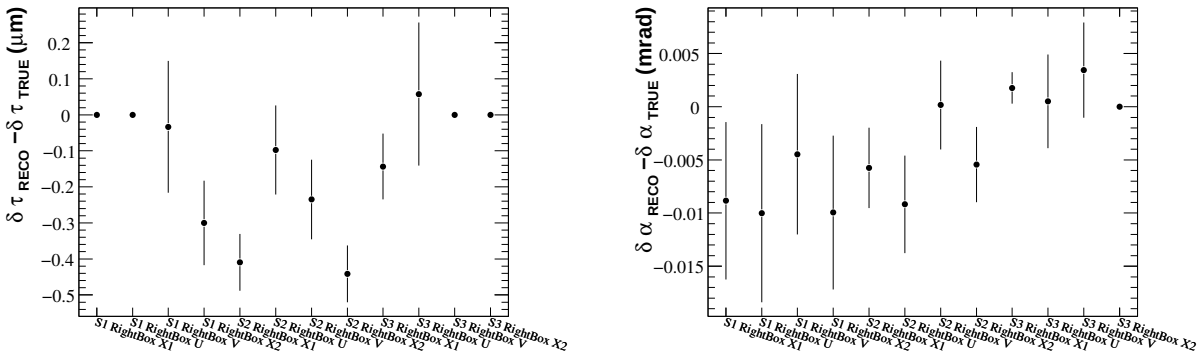


Figure 5.18: Alignment results after 21 iterations, with corresponding Station, Box, Layer index label

The 5.18 figures are resolution plots giving in addition the labels of each degree of freedom. The plots were done for the Right-Cryostat boxes, with a stack of 12 layers connected by tracks coming from a primary vertex. The difference between $\delta\tau$'s degrees of freedom and $\delta\alpha$'s is again evident. The latter have more consistent error estimates from Millepede. Not surprising if the $\delta\tau$ errors are produced by random scattering, which does not have the same impact on rotational degrees of freedom.

One consequence of random scattering at low energies, is to make the Newton-Raphson convergence more difficult. The Figures 5.19 compared with their Particle Gun counterparts 5.6, show that before the activation of the track-χ^2 cut at the 7th iteration, the alignment parameters fluctuate greatly. After I switch-on the track-χ^2 cut, the effects diminish, and as the cut becomes harder with each iteration the convergence of results becomes evident. The blue dots on the 5.19 graphs, correspond to an X-layer which is close to a layer that has the $\delta\alpha$ a priori fixed. This explains the lower fluctuation in the latter degree of freedom, as compared with the other two.

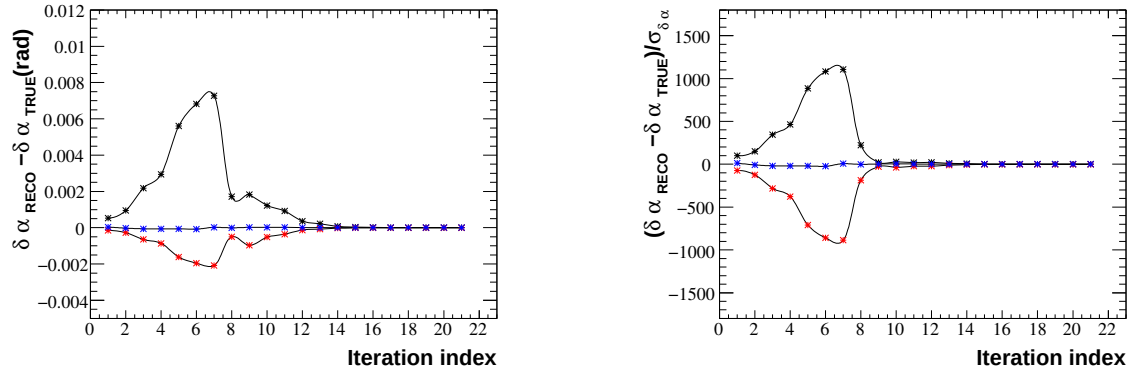


Figure 5.19: Alignment progression with iterations for rotation around beam axis, the effect of the χ^2 quality cut is seen after the 7th iteration when the bias due to random scattering is reduced, and convergence becomes evident. The plotted quantities correspond to: Station 1 Top Box Layer X2, Station 2 Left/A-Box Layer U, Station 3 Bottom Box Layer U. (rank 3, 25, 37)

To have a confirmation on the convergence it is possible to look how the global χ^2 of the fit evolves

with iteration. In the same time we have to monitor the total number of degrees of freedom. I found it useful to implement the convergence criterion with respect to the reduced alignment-global χ^2, χ^2/N where N is the total number of degrees of freedom for the alignment fit[16]. The last quantity is labeled total because it is the sum of all degrees of freedom for all tracks minus the number of alignment parameters not a priori fixed. The figure 5.20 displays the global-χ^2's evolution and the number of tracks used in alignment at each iteration. Since I do not re-run the Pattern-Recognition algorithm on the tracks, their hit composition remains the same during successive iterations. It is obvious that after the 14th iteration the results converged. Here the iterations were allowed to continue, even though the convergence criterion was fulfilled. The convergence plateau, is visible in both global χ^2 and the alignment parameters (see fig. 5.19).

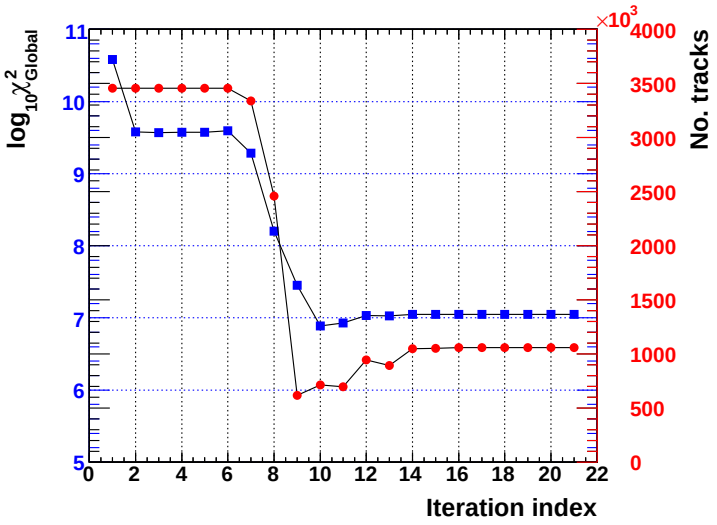


Figure 5.20: Evolution plots of the global χ^2 and number of tracks with iteration number

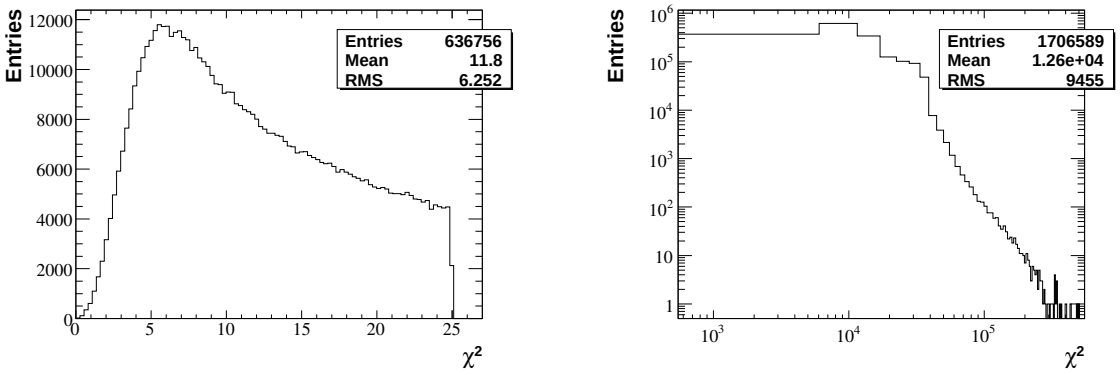
The final global-χ^2 is about 11.13 millions with a number of degrees of freedom equal to 8.38 millions. The reduced χ^2 is about 1.33, much too large compared to one. Again this is not unexpected, since a proper measurement error needs to include random scattering. These numbers are in contrast with the close agreement obtained for Particle Gun data. If the same grade of agreement is desired, then it is straightforward to cut on the Calorimeter energy deposit and select high energy pions. Electron on the other hand should be avoided, since their track quality suffers as a result of bremsstrahlung. I have already made alignment runs using cuts on the Calorimeter energy estimates. For now, the latter procedure was decreasing the track sample to much. Thus more Minimum Bias data would be required, together with a method to separate between energy deposits in HCAL and ECAL. The last criterion allows to separate the low mass leptonic component from the hadronic part. This is one avenue that I had no time to cover, unfortunately.

The effect of random scattering is to smear the track-χ^2 distributions. The large tail in the 5.21 graph is in contrast with the much smaller tail of the Particle Gun case (see figure 5.8(a)). More tracks get rejected as the iterations proceed, with only about one third of initial track sample still present in the final iteration.

One interesting point in Figure 5.21(a) is that although it is clearly afflicted by the random scattering smearing, it represents in the same time a clear improvement with respect to the plot 5.21(b). With track quality in mind we can look whether there is any improvement on the measurement residuals after alignment.

We know that Millepede as any least square method, minimizes the spread of the measurement-

[16]Care not to mix the fit statistical degrees of freedom with the alignment geometrical degrees of freedom



(a) track-χ^2 distribution after 21 Newton steps

(b) track-χ^2 distribution before Alignment

Figure 5.21: Estimated track χ^2 in the last iteration, here the 21st. Quality cut equivalent with a 3.15 σ probability rejection. The effect of random scattering still evident in the smearing of the data points. Only tracks with 8 degrees of freedom.

residual distributions. In the same time the track model used here is exactly equivalent to the default tracking model of the Pattern-Recognition algorithms and the Kalman-Filter track model. This is due to the fact that in absence of the magnetic field, the energies associated with a certain track are not obtained from tracking, and thus all algorithms use a line trajectory without random scattering estimation[17].

All the precedent facts lead someone to conclude that if it exists an improvement in measurement residual distributions after alignment, the effect will be universally true for any tracking or pattern-recognition algorithm. Of course, someone might argue that with the alignment precision much better on average than 1 μm this is a foregone conclusion. It is nevertheless nice to check this statement, using the the residuals computed by alignment's tracking. We already seen the effect of alignment in track-χ^2 graphs.

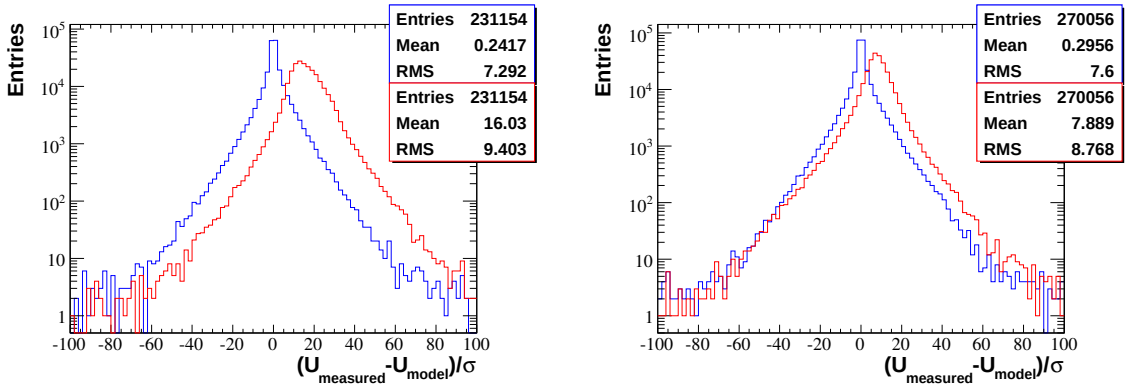


Figure 5.22: Plot of residuals for measurements in two sensors, left a sensor in 1st Station, right a sensor in 3rd Tracking Station. Red histograms are before alignment and blue after alignment. Tracks not used in alignment dominant in blue histogram. The sensors are the central-Top X2 and central-Bottom X1, respectively.

In Figures 5.22 and 5.23 I plot the measurement residuals:

[17] In tracking and alignment it is possible to estimate the effect of the random scattering, by giving the track an energy equal to the Calorimeter energy estimate. This in case there is an associated Calorimeter cluster. To my knowledge this approach was not implemented, yet.

1) for the central sensor of layer X2, of the Top-Box, of the first Station;
2) and for the X1 layer of the Bottom-Box, of the last Station.
The sensors just mentioned are those that receive on average a high occupancy per event, due to their proximity to the beam.

In the first plot of 5.22 we see the improvement after alignment for the entire sample of tracks with measurements in the selected sensors. The initial bias due to alignment is gone, and the only effect visible is the broadening of the distribution, due to random scattering. This deviation from Normal distribution was expected, as we have not included in measurement error estimates the random scattering.

The second plot, 5.23, on the other hand offers a confirmation of the alignment quality. The blue histogram has included only those tracks that were used in the last iteration of the alignment. This time the blue histogram is almost a **perfect Normal distribution**. It is a nice check, and it provides as a feed-back a confirmation that the alignment parameters and the track parameters are the true best linear unbiased estimator in the sense of chapter 3, even though the problem is non-linear[18].

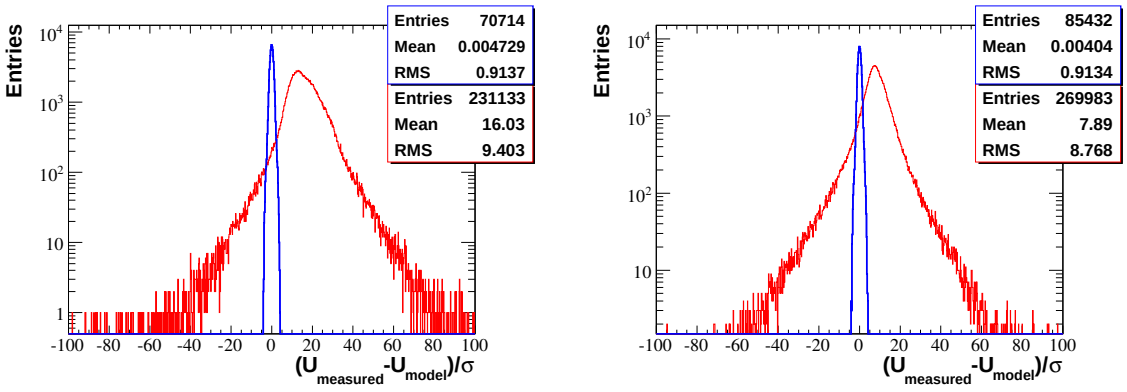


Figure 5.23: Plot of residuals for measurements in two sensors, left a sensor in the 1st station, right a sensor in the 3rd tracking station. Red histograms are before alignment and blue after alignment. All Minimum Bias events were used in alignment. The blue plot includes only the tracks used in the last iteration step of alignment. Sensors as in 5.22

Actually some of the sensors especially the ones far from the beam, acquire small biases and a slightly larger broadening in their residual distributions, and this explains some of the larger alignment values seen. Yet, the upper statement remains valid, given that the biases are smaller or equal to about 1 μm, and thus much less than the measurement error (of about 20-40 μm for Monte Carlo data). The small biases in the outer-lying sensors of Left/Right Boxes are to be expected, since the sample used in alignment has the lowest occupancy in the region of these sensors, the alignment result will depend in average less on these sensors than on the more inner part of IT.

If desired the alignment precision can be made better by increasing the number of alignment tracks. A parallel run on 8 CPU cores, for 10 million similar events would take about half a day. If needed, tuning of the track-χ^2 cut method can be done and the required number of iterations decreased.

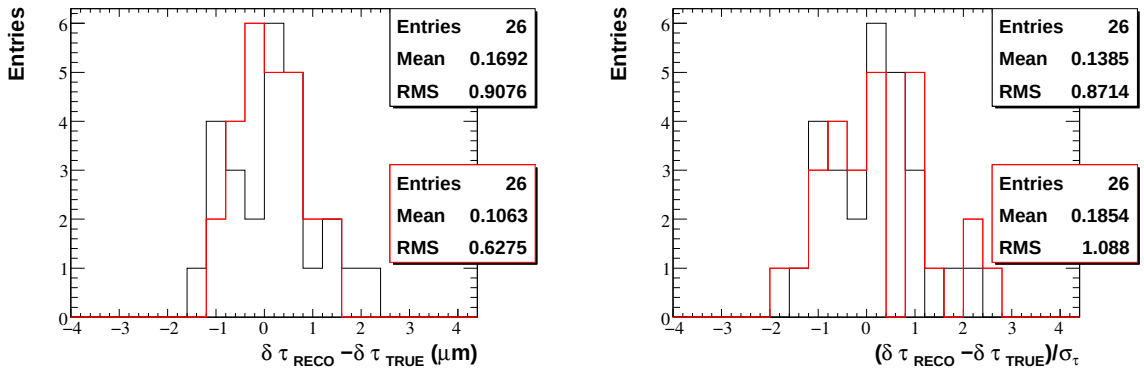
Pull Plots of 26 Runs

To provide true pull plots of the alignment parameters, I have split the previous track sample in 26 independent track samples, of roughly the same number of tracks. For each sample I made a separate alignment run, and this allows to extract the pull plots and resolution plots for a single alignment parameter. The alignment was done for both alignment parameters types $\delta\tau$ and $\delta\alpha$, simultaneously.

In the plot of figure 5.24(a), I give the errors obtained when aligning the Top-Box' 1st X layer and U layer in $\delta\tau$ shift. The resolution in the X layer is a little better at 0.6 μm than the $\delta\tau$ U layer's resolution

[18]The non-linearity was cured when the convergency was reached.

at 0.9 μm. The better resolution is explained by the choice of constraints. An a priori fixing of alignment parameters gives better resolution for the neighboring layers of same type. E.g., the Top Box has the last X layer and the V layer fixed in the $\delta\tau$ shift, which in turn make the alignment of the first X layer of the same box more exact. In "real life" this means that the inclusion of an external information about the positions of the last X and V layers, will make the position of the near-neighboring layers better known in an alignment with connecting tracks[19].

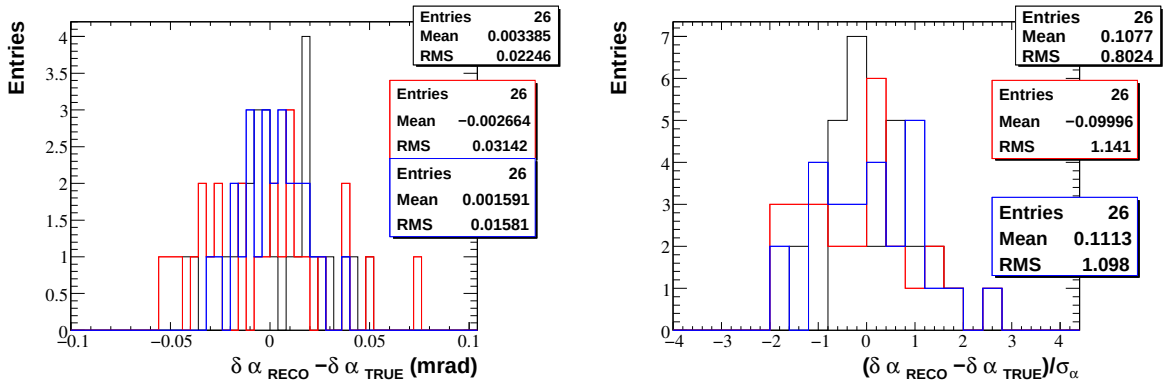


(a) $\delta\tau$ resolution plots

(b) $\delta\tau$ pull plots

Figure 5.24: Resolution and pull plots of 26 alignment runs for 2 DoF of type $\delta\tau$, Top-Box, 3rd station, 1st X and U layers

The pull plots of figure 5.24(b) have the RMS and Mean consistent with the Normal distribution, sign that the error propagation in the Newton-Raphson+Millepede procedure worked without significant bias. For $\delta\alpha$ the Top Box layers 1st X, U and V, I found the following non-biased[20] resolutions: 0.015 milli-rads, 0.032 milli-mrads, and 0.023 milli-rads, respectively.



(a) $\delta\alpha$ resolution plots

(b) $\delta\alpha$ pull-plots

Figure 5.25: Resolution and pull plots of 26 alignment runs for 3 DoF of type $\delta\alpha$. The corresponding layers are in Top-Box, 3rd Station, 1st X, U and V layers

[19]This is obvious, since larger the lever arm between the layers, the larger the position uncertainty will be, especially when aligning in magnetic field and with low momentum tracks

[20]The mean of the distribution was consistent with null value 5.25(a)

The histograms of figure 5.25(b) are consistent with the Normal distribution. Hence, the error propagation worked even for the non-linear $\delta\alpha$ alignment parameters[21].

5.2.6 Alignment for Minimum Bias Events at 7+7 TeV or 450+450 GeV Collisions, Open or Close VELO, B-off

21k Events at 14 TeV with Closed VELO

Applying the same recipes for a small sample of Minimum Bias events with a nominal beam and luminosity, gives similar results as in the precedent alignment. This last statement is true, provided that we take into account the track sample size difference. In this alignment, we have available close to 190k tracks, and as displayed in the Figure 5.27 we have close to 60k usable tracks in the last Newton-Raphson iteration. One other factor is that this time VELO is in closed position, and as a consequence we can access a higher energy component, which was lost otherwise. This extra feature, insures a slightly better error propagation and iterative stability. Due to the smaller track sample, the random scattering biases are smaller than the intrinsic uncertainty in the alignment parameters.

This sample of data was produced by one of our co-workers, and he was kind to make it publicly available [34]. The sample comes in two versions, one reconstructed with a nominal-perfect geometry, and one produced with a misaligned geometry. The misalignments used were not large, they were chosen less than 1 mm in shifts and correspondingly small in rotations. The other detectors were misaligned as well. The effect of reconstruction with such a geometry, was to decrease the number of tracks. However, the local IT alignment results were not changed except for a slightly larger uncertainty due to the decreased sample size.

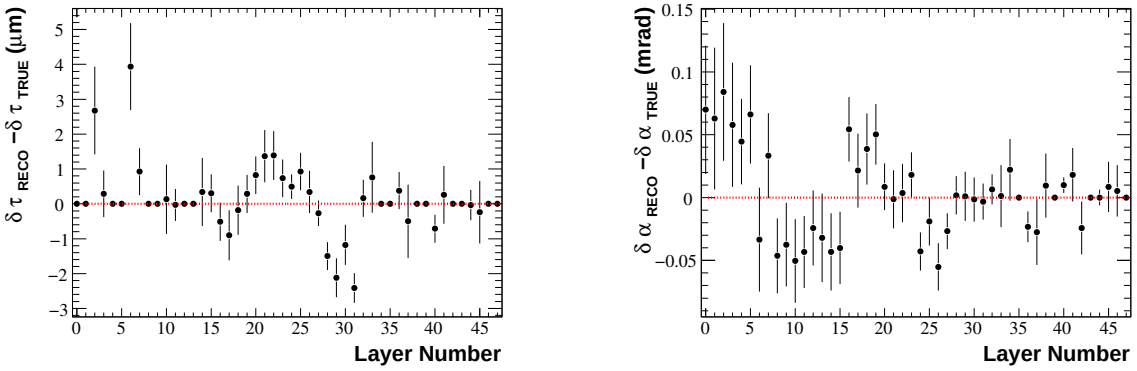


Figure 5.26: Alignment results after 27 iterations, shifts along measurement direction ($\delta\tau$) and rotation around beam axis $\delta\alpha$, error bars given by the alignment method. 7+7 TeV, Closed-VELO.

Figures 5.26 show the alignment results for the nominal-geometry sample. The reconstructed $\delta\alpha$ and $\delta\tau$ alignment parameters are given together with estimated errors from Millepede. In this case a misaligned geometry was used only in the Alignment phase. In the plots of 5.26 the misalignments were subtracted.

The convergency of the Newton steps, this time without quasi-Newton and χ^2 checks, is outlined in the track number and global-χ^2 evolution graph 5.27. The alignment parameters display the same convergency trend as the global χ^2, and it is the same for the error estimates of Millepede. In conclusion, even for this small sample, an average alignment precision within 1-2 μm was achieved for the $\delta\tau$'s geometry degrees of freedom. The alignment precision for the rotational degree of freedom was found smaller than 0.1 mrad, and the average over all layers is of about 0.035 mrad.

The $\delta\beta$ degree of freedom was reconstructed simultaneously with the other 2 types. Due to space constraints I include the results for the third alignment parameter in appendix' figure 8.16.

[21] $\delta\alpha$ appears in the alignment Model in a mix term of alignment and track parameters.

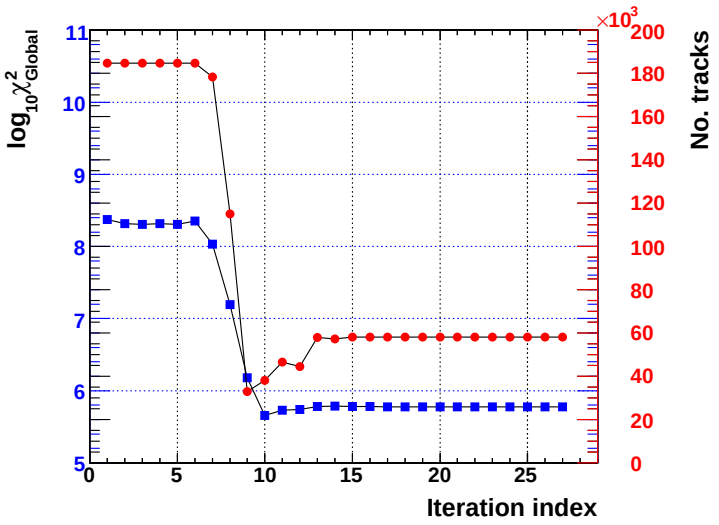


Figure 5.27: χ^2 convergence with number of tracks used in alignment, plateaus of convergence observable. 21k Minimum Bias events with 7+7 TeV and Closed-VELO.

Minimum Bias, 900 GeV Collisions, Open-VELO

Alignment set 3072 of simulated data for 450+450 GeV collisions, with an Open VELO, was made at the request of the alignment group. The sample is produced and reconstructed with the SIMCOND Nominal geometry. There are in total about 1 million Minimum Bias events in this sample. Due to the much lower collision energy, the number of reconstructed tracks trough IT is reduced by a factor of about 2 with respect to 3074 sample with 10 TeV center of mass collisions. The latter case was already discussed in the previous subsection. As then, I reconstructed only two types of alignment parameters, $\delta\tau$ and $\delta\alpha$, respectively.

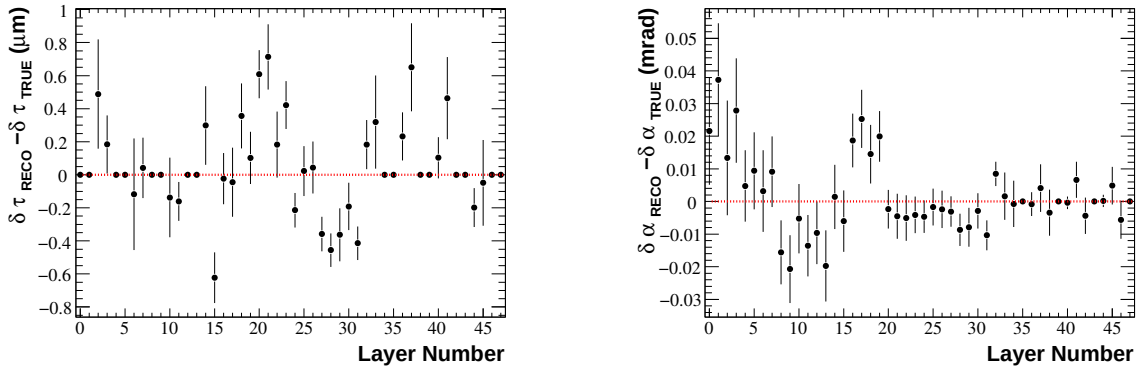


Figure 5.28: Alignment results after 21 iterations, shifts along measurement direction $\delta\tau$ and rotation around beam axis $\delta\alpha$, error bars given by the alignment method. 450+450 GeV, Open-VELO.

The usual constraints were used: 16 a priori fixed $\delta\tau$'s and 4 $\delta\alpha$'s. It makes the alignment parameter number shrink to 76, from an initial number of 96. When using the 3072nd sample, I got the alignment results given in Figures 5.28, where the free alignment parameters are plotted together with the fixed alignment parameters. At the end of the iteration procedure, the uncertainty on $\delta\tau$ is lower than 1 μm and on average 0.3 μm. The precision on $\delta\alpha$ rotation is on average 0.011 milli-radians, and any extreme observed deviations within 0.04 milli-radians.

A Standard Newton-Raphson method was applied[22]. For the studied geometry degrees of freedom, there is no need for robustness, the usual Newton steps will do. With this method the convergence is achieved quickly after 2 steps. Again, only the track-χ^2 cut and the subsequent track number reduction change significantly the global-χ^2 evolution towards convergence, as displayed in 5.29. Unlike alignment

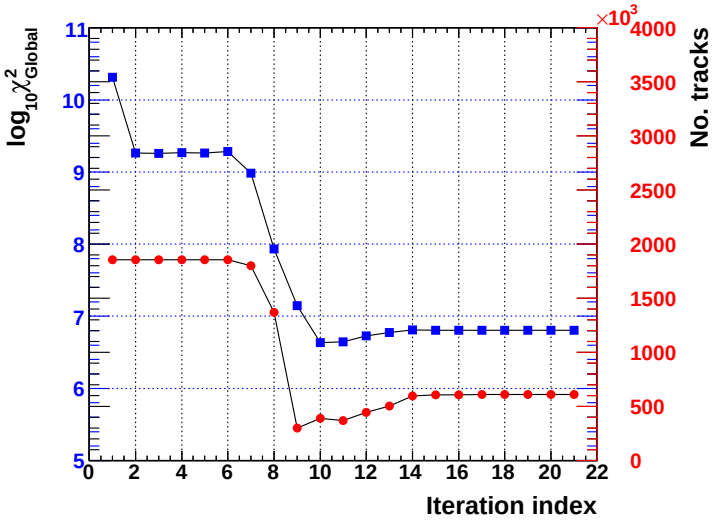


Figure 5.29: 21 iterations, alignment-χ^2 dependence on iterations, number of useful tracks, stability, 450+450 GeV, Open-VELO .

with all 5 degrees of freedom per layer, here, save for the track-χ^2 cut region, there is no significant non-linear fluctuation after 2 Newton steps, see Fig. 5.29. The alignment parameters follow the same trend as the one exhibited by the global-χ^2 evolution curve.

5.2.7 Layer Alignment, Conclusions

In conclusion of this section, it is worth mentioning that other samples were used in the past. E.g., Beam-Gas collision events produce reconstructed alignment parameter with lower precision than the high center-of-mass energy events. Only 3 degrees of freedom were universally obtainable for layers: $\delta\tau$, $\delta\alpha$ and $\delta\beta$. The two additional alignment parameters $\delta\gamma$ and δz are well reconstructed only for high energy track sample. Even there it was needed to use robust quasi-Newton steps. The problem is not in convergency of the algorithm, but it resides in the implementation of the track quality cut. In the presence of strong non-linear fluctuation amplified by the random scattering effects, the track-χ^2 rejection method might reject most of the tracks in one iteration.

I think, the problem can be cured by a regularization scheme of the Matrix, coupled with an optimization of the track-χ^2 cut method, along the lines of the Levenberg-Marquardt algorithm. Yet, I would say that for real data, a sample of a million tracks of high energy pions, is not an unrealistic expectation. The pions would be tagged by the LHCb Calorimeters, which would supply also an energy estimate for the reconstructed track.

[22]No quasi-Newton steps

5.3 Ladder Alignment

There are 336 ladders in IT, 168 of them single sensor ladders in Top and Bottom Boxes. In the figure 5.30, it is shown the IT ladder arrangement for the first 4 layers as seen form the primary vertex position. The length of the distance from the collision point to the IT surface makes the primary vertex track almost perpendicular to the ladder surface. It means that the ladders are directly connected by tracks, only if they are successively placed on the Z axis. There is naturally some connecting tracks between near-neighbors, e.g., via the small overlaps between neighboring ladders in the same layer. These overlaps are so small that as in the case of layers where the Inner Tracker was split in quasi-independent quadrants, we have 28 quasi-independent stacks of ladders. There is some degree of connectivity between stacks, but the observed effect in the absence of the magnetic field is negligible, with only less than 5% of Minimum Bias track sample connecting neighboring stacks.

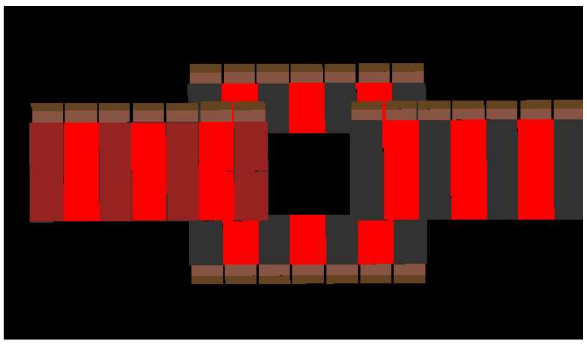


Figure 5.30: 28 ladders of the first 4 layers in IT

To be able to align at ladder level the IT, it is needed to separate the alignment problem in 28 individual stack alignments. As for layers quadrants, it is required to impose a priori constraints. The requirement amounts to fixing the ladders of 16 layers in $\delta\tau$, and the ladders of 4 layers in $\delta\alpha$. This will properly define our alignment problem, when aligning in $\delta\tau$ and $\delta\alpha$.

Ladder lever arm is much smaller that the lever arm of a layer with 7 individual ladders. It is expected as a consequence that the least constrained alignment degrees of freedom shall not be determinable. This is true for $\delta\gamma$'s, δz's, $\delta\beta$'s.

In the same time, for the remaining 2 types of alignment DoF, we are less sensitive to $\delta\alpha$ because the lever arm again. A rotation around LHCb's z axis for a ladder will affect the measurement-residual distributions less as when we were rotating all 7 ladders of a layer around layer's center. This in turn means that the $\delta\alpha$'s will get more susceptible to suffer from random scattering biases.

To decrease the effect of weak modes, an extra constraint was added per ladder stack: a $\delta\alpha$ alignment parameter if a priori fixed, for the ladders in the last 4 layers of the 3rd station (this in addition to the ladders of the first 4 layers in the first IT station). The constraint is not mandatory but it helps the iterative procedure and the error propagation.

To remove any possible remaining weak mode, an over-constrained system was considered, too. Initially, it was not clear that the chosen set of constraints will be sufficient to fix some x-y global scaling modes of the IT, but it turned out that the extra constrains were not required.

5.3.1 Particle Gun Data

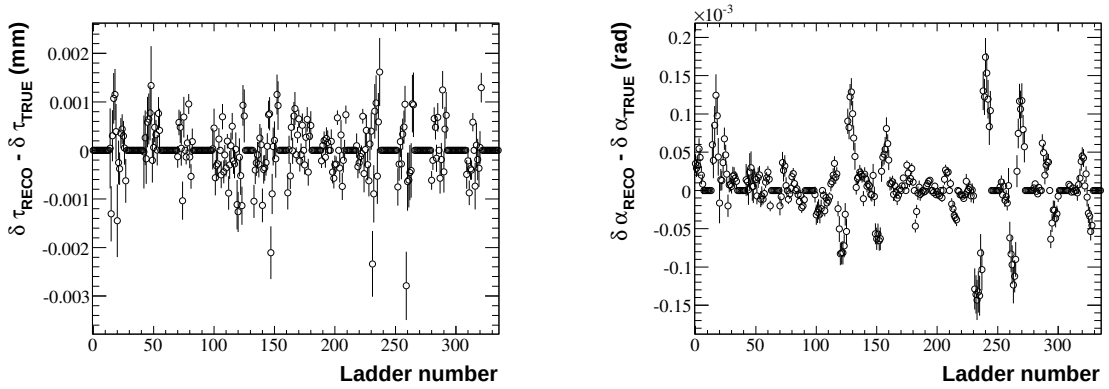
The particle gun data is the same as in the layer alignment. Though, this time it includes always the full track sample of about 9.2 million tracks, mostly muons at 100 GeV.

Over-constrained System

First it is considered the over-constrained setting. For a given IT ladder stack, I constrain the following measurement direction shifts or $\delta\tau$'s:
- the first ladders, which are in the layers X1 and U of the first station;
- the last ladders, which are in the layers V and X2 of the third station;
- the ladders in 4 middle layers.

For rotations $\delta\alpha$, as already stated, the constraints applied are for the ladders in the first 4 U-layers and the last 4 X2 layers, at the end-caps of IT.

The most outer ladders in the Left and Right Boxes were constrained in both $\delta\tau$ and $\delta\alpha$ degrees of freedom. Usually the latter ladders have the lowest occupancy level, hence it is for them that we get the largest uncertainties in alignment. An uniform track sample would have helped here.



(a) $\delta\tau$ alignment parameters

(b) $\delta\alpha$ alignment parameters

Figure 5.31: Resolution plots, ladder alignment results in α and τ for an alignment with Particle Gun data, over-constrained system

The alignment results are given in figures 5.31. The average error estimate is 0.6 μm, and as seen the reconstructed values are within 2 μm, excepting the ladders which are farther for the beam-pipe. These outer-lying ladders have too few measurements, and as a consequence have the lowest alignment precision.

For $\delta\alpha$ the alignment precision is on average 0.05 milli-rads, and the reconstructed values fluctuate in a range between -0.15 milli-rads and 0.2 milli-rads.

The convergence for this case is apparently slower as for layers, see 5.32. It is only apparently, because in the case at hand the method of quasi-Newton steps was used, which slows the convergence rate and in the same time smoothens the evolution graphs of track number and global-χ^2, see Fig. 5.32. In the absence of quasi-Newton steps and with the standard Newton-Raphson, the convergence rate is no different then the one in case of layer alignment.

The error propagation does not work as good as before. I see underestimation in the average resolution of $\delta\alpha$ equivalent to a 3 factor difference, and a 2 factor for $\delta\tau$. The quantitative effect described, is explained by some residual random scattering effect in case of $\delta\tau$, and by weak modes in $\delta\alpha$. The presence of the weak modes is due to the smaller model derivatives and the smaller acceptance range of a sensor in a ladder when compared with the layer case. The track parameter phase-space associated with a ladder or sensor is 7 times smaller as for a layer. It means that the tracks with measurements in an IT-ladder have a very restrictive range of variation for their track parameters. In the simple model function 4.1, the $\delta\alpha$ alignment parameter gets maximal coupling to the track slopes t_x and t_y (for these parameters the coupling constant is z equal to thousands mm). Since these slope parameters have a small

fluctuation range and are practically null for Top and Bottom boxes' ladders, the contribution from these correlation terms is too small. In this way some weak modes are created for the ladders of the Top and the Left boxes, and in a smaller amount for the ladders of the Left and Right boxes.

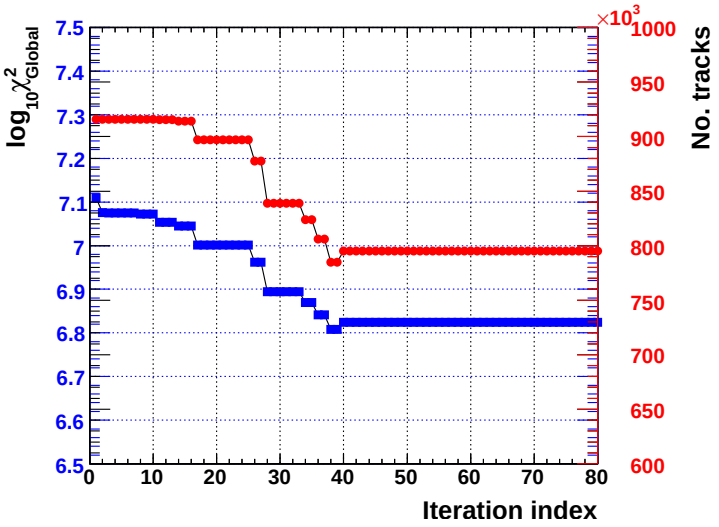
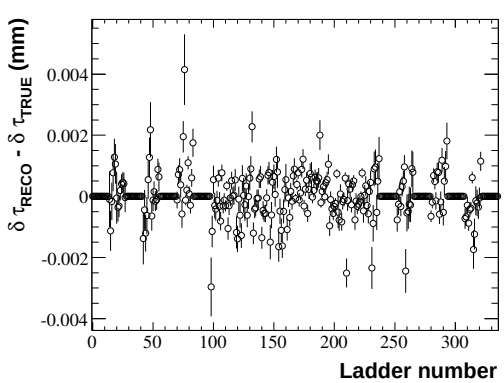


Figure 5.32: χ^2 and number of tracks versus iteration index for Particle Gun data, ladder alignment

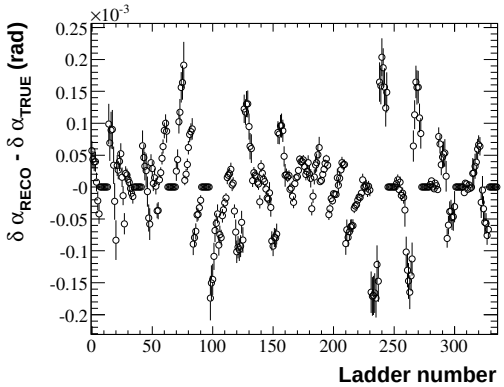
The error propagation is secondary in importance, though, and the reconstructed alignment precision is hopefully sufficient. If not, an increase in track sample size, and some uniformity in track parameter sample would fix the problem of these weak modes of $\delta\alpha$. It will be displayed later in the chapter, cases with alignment results that suffer more drastically from the effects of random scattering.

Minimally-constrained Setup

Simultaneously with the precedent alignment, an alignment with less constraints was performed using the same track sample. The found alignment precision in $\delta\tau$ is on average 0.8 μm. The slight difference with respect to the last alignment, is due in most part to inclusion in alignment of the most-outer-lying ladders. The average error estimate for $\delta\alpha$ is 0.07 mrad. Again the difference is due mostly to the outer ladders.



(a) $\delta\tau$ alignment parameters



(b) $\delta\alpha$ alignment parameters

Figure 5.33: Resolution plots, ladder alignment results in α and τ, for a minimally constrained setting

The underestimation of the alignment error is identical with the one found in the last case. The plots in 5.33, correspond to alignment results of all ladder degrees of freedom, except the a priori fixed ones plotted on the null axis.

The fixed $\delta\tau$ alignment parameters are: the ladders of layers X1, U of first station, and ladders of layers V, X2 of last station. In $\delta\alpha$ are fixed only: U-layer's ladders in 1st Station, and X2-layer's ladders in last Station.

The relatively slow convergence of the results as exemplified by figures 5.34, is due again to the use of the quasi-Newton-steps method[23] with slow convergency rate. The sequence of small plateaus in 5.34, is generated by the hardening of the track-χ^2 cut.

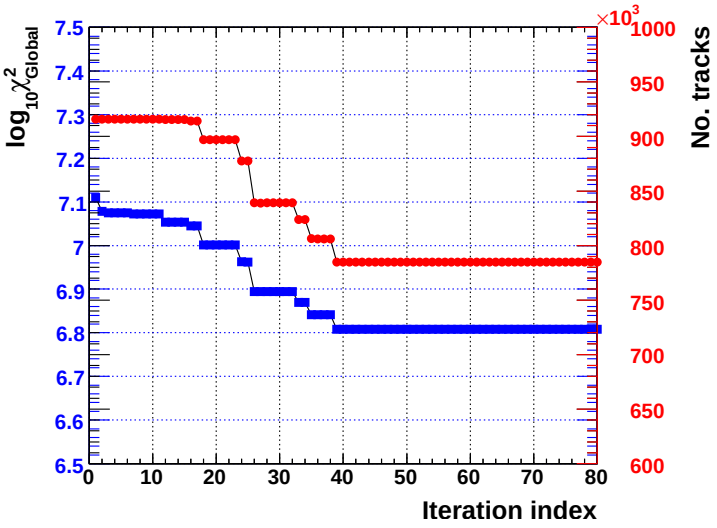


Figure 5.34: χ^2 and number of tracks versus iteration index for Particle Gun data, ladder alignment

The effect of removing the extra constraints was negligible, and identical results were obtained in both setups. As a note of caution, the last statement does not mean that the observed weak modes in the $\delta\alpha$ degree of freedom, are unimportant. It is exactly these modes which in combination with the small random scattering[24] perturb the error propagation in Millepede.

5.3.2 14 TeV, Closed-VELO, Data Sample with 21k Minimum Bias Events

In this subsection there is no major high energy component in the track sample used, hence we could expect larger effects coming from random scattering, even with the track-χ^2 quality check in place.

A minimal set of constraints was used: 56 $\delta\alpha$'s a priori fixed together with 112 $\delta\tau$'s, as in the previous alignment.

Figures 5.35 reflect the error estimates and reconstructed alignment parameters of all aligned ladders, i. e. the usual resolution plots.

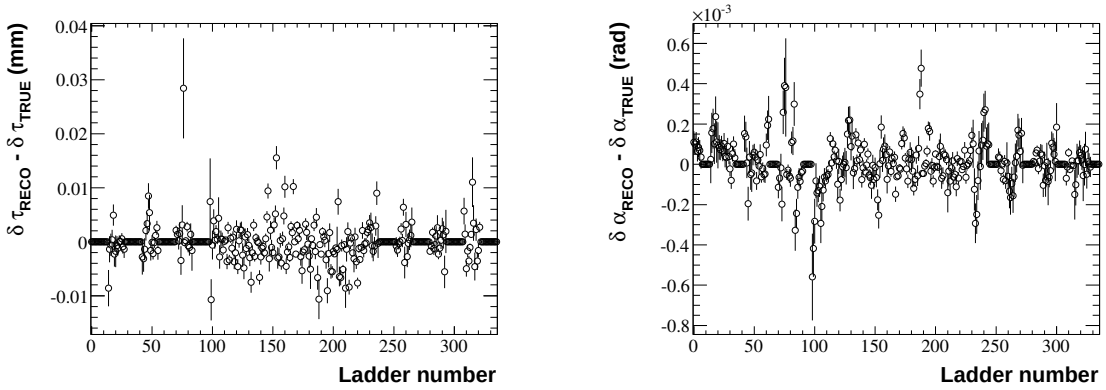
In this case the alignment precission in $\delta\tau$ is in average 4 μm, explainable since I have used a smaller sample this time. The rotation around z axis of each ladder is determined up to an average uncertainty of 0.1 mrad. The rotation pivot is in the center of the ladder. Several ladders have much larger uncertainty in rotation, but these ladders are the outer ladders with less occupancy per event.

Convergency is slower again for the same reason, as I am using quasi-Newton steps. The quality track-χ^2 cut creates a stepper descent in the evolution of the global-alignment χ^2, see Fig. 5.36. The reduction in the number of low momentum particles, allows removal of most random scattering effects.

[23] The quasi-Newton-step method of 3 is only similar and not identical with any of the Quasi-Newton Methods

[24] After achieving convergence, the track-χ^2 cut has eliminate most of random scattering effects.

The Particle Gun data gave alignment resolutions of 0.07 mrad and 0.8 μm. The last alignment 4 μm and 0.1 mrad. Taking into account the track sample size difference equal to a multiplicative factor[25] of 11, then these numbers are not very different.



(a) $\delta\tau$ alignment parameters

(b) $\delta\alpha$ alignment parameters

Figure 5.35: Resolution plots, ladder alignment results in α and τ for nominal LHC beam and B-off

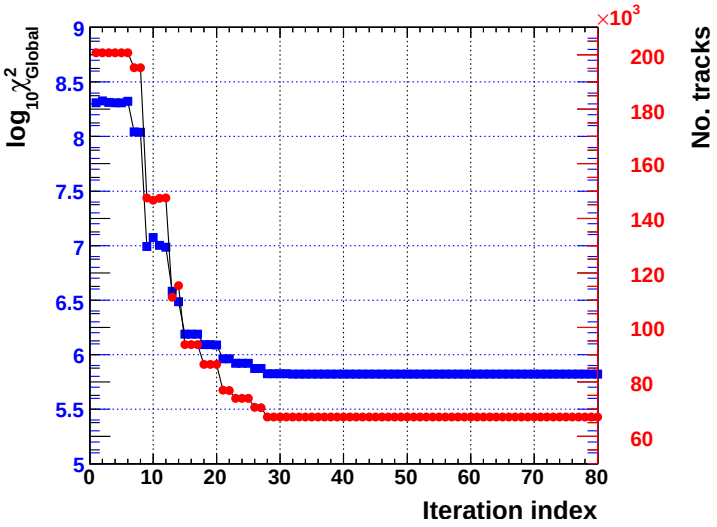


Figure 5.36: χ^2 and track number versus iteration index for 14 TeV data sample, ladder alignment

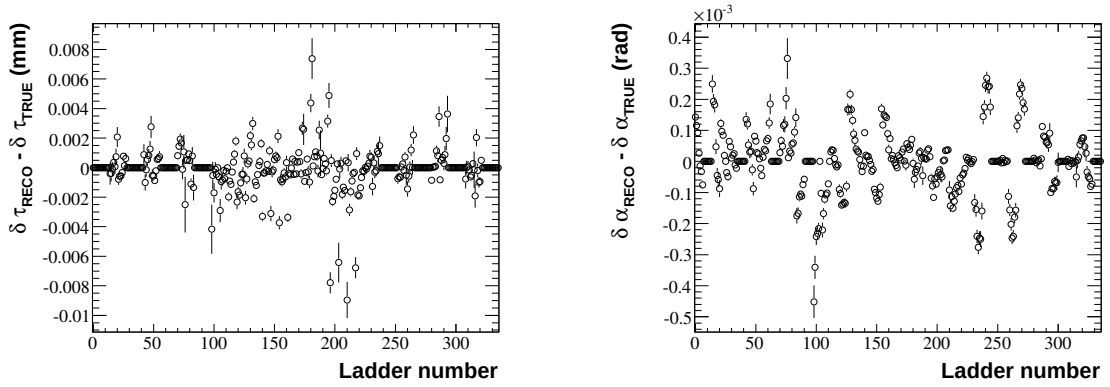
The error propagation experience the same difficulties as for Particle Gun data, yet the effect is less in $\delta\alpha$ case. Here the intrinsic larger uncertainty for a small sample, makes the effect of $\delta\alpha$ weak modes not as relevant as before. The Millepede error estimates are much larger for this alignment sample than the error estimates obtained in Gun Particle case.

5.3.3 10 TeV, Open-VELO, Data Sample 3074

When using a 1 million Minimum Bias event sample, the about 4 million IT-tracks of this sample experienced a lot of random scattering. Leptonic contribution is expected to play a role, too. The number of tracks used in the final iteration is of about 1250k (see figure 5.38).

[25]The number of tracks in the last iteration matters, not the size of the total sample.

Using the same set of constraints as before, Figures 5.37 reproduce the results of this alignment. The average precision of $\delta\tau$ alignment parameter is 1.7 micro-meters. The offending reconstructed values in the 5.37 graphs are due to the most outer-lying ladders. When VELO is open, It seems that these rim-placed sensors have quite some low occupancy for a 10 TeV collision. Hope it is not some "feature" of the Reconstruction latest release. The average reconstructed precision in $\delta\alpha$ is again 0.1 mrad.



(a) $\delta\tau$ alignment parameters

(b) $\delta\alpha$ alignment parameters

Figure 5.37: Resolution plots, ladder alignment results in α and τ for 3074th data sample

The 3074 data sample, proved largely unsuited for ladder alignment. The error estimates in 5.37 are clearly unrealistic. I have noticed multiplicative factors of 6 in the quasi-pull-plots of $\delta\alpha$'s when compared with the Normal distribution. The effect is less dramatic for $\delta\tau$, where the quasi-pull-plots are broader with a factor of 3 compared with the Normal distribution.

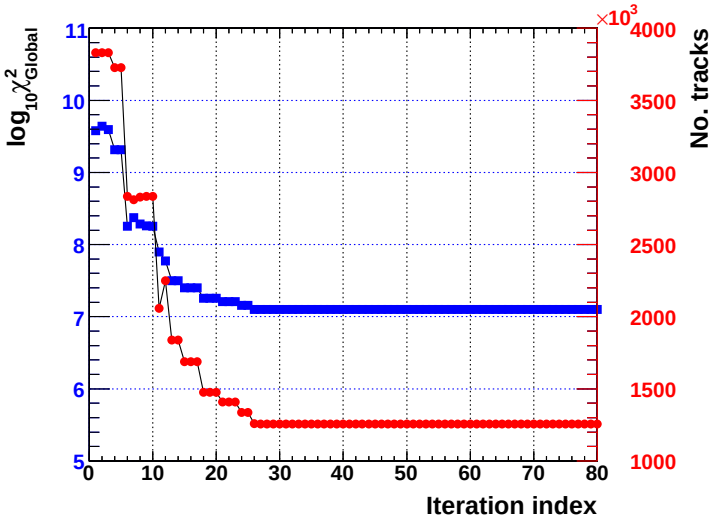


Figure 5.38: χ^2 and number of tracks versus iteration index for 10 TeV data sample, ladder alignment

Clearly, to properly align at ladder level, it is required to have a sample with high energy pions. Muons would be even better, but for Minimum Bias data those particle are in short supply. With these low energy samples, the error estimate distribution is broader than the Millepede-supplied error estimates. Yet, even for these kind of track samples, alignment resolution better than 1 μm in shift and 0.1 mrad in rotations is achievable if sufficient tracks pass through the most outer-lying ladders. One other possible

improvement is to fix the weak-modes in $\delta\alpha$ which are largely responsible for the error propagation difficulties[26].

5.4 Alignment Matrix, Global Modes, Weak Modes, and Distortions

The fields of Matrix Regularization, Singular Value Decomposition, and Matrix Diagonalization are wonderful mathematic "landscapes". Their diversity is extreme, and I am truly sorry I had no time during my research to study them in more detail. For now I shall just be an user of these tools, and as any user I shall be trying to include and implement them in my alignment algorithm frame.

In the first subsection I give an introduction to the alignment's mathematical regularization. Most of the times, the main task in alignment is redefining the problem to avoid the "ill-posed problems". In alignment, it is usual to start with a problem which can not be solved – e.g., when the alignment matrix is singular – and after several trials to achieve a redefinition of the alignment problem that makes the problem solvable.

To reach the latter point, it is required to a priori fix all global modes together with all those weak modes that may disturb irremediably the alignment. By irremediably, I understand the weak modes that would prevent a successful Newton-Raphson convergence, or would simply give an alignment precision which is too bad to be useful. Here, Newton-Raphson method is implicitly including the quasi-Newton steps and the track-χ^2 quality cuts, and even though with quasi-Newton steps the convergence is always reached for a given sample of tracks, it might occur that during the hardening of the track-χ^2 cut most tracks would be rejected. The latter effect is caused by big non-linear fluctuations, which themselves are a byproduct of the weak-modes.

Global and Weak mode tagging, must be the first procedure to be applied in any serious alignment undertaking. Defining the alignment problem turns then into selecting a suitable threshold from which all the $\mathbf{C}'$'s eigenvalues or singular values are too small to contribute to the alignment solution. The corresponding singular-modes make what is named in the mathematical regularization field, the "null space" of the matrix.

In the case of the symmetric alignment matrix, $\mathbf{C}'$, a diagonalization procedure is perfectly equivalent with the singular-value decomposition up to a sign change. I have used the two methods most of times in parallel, and arrived to indistinguishable results. Due to space limitation, I restrict myself to the diagonalization, and the corresponding spectra of the alignment matrices.

5.4.1 Layer's $\delta\tau$ Alignment and the Global Degrees of Freedom

Shearings and global translation of the IT layers are easily tagged by looking at the alignment matrix eigenvalues. To avoid effects unrelated directly with the alignment, I use the Particle Gun data for aligning the detector in $\delta\tau$. Since these geometrical degrees of freedom are linear in all model functions, this makes the problem almost linear. In model function 4.10, the terms depending on the track parameters are non-linear, however the last two fractions in 4.10 have denominators which are close to constants, and numerators close to linear functions ($\xi_z \approx 0$ as well as other z-vector components).

Having the problem linear allows immediate solving of weak modes and global modes, since for a linear model function, the global χ^2 function which gets minimized is not an approximation of the real χ^2, but is the true χ^2, itself.

By leaving out any a priori constraints, and letting the system move freely in all global translations and shearings, we can compute using the alignment algorithm, the alignment $\mathbf{C}'$ matrix as in chapter 3. The matrix will be singular, and hence no alignment results will be retrieved this time.

$\mathbf{C}'$'s spectrum gives us the alignment weak modes, and maybe even more important gives us the global degrees of freedom for the alignment. In plot 5.39, it is possible to tag 4 global degrees of

[26]The residual weak modes in $\delta\alpha$ are largely removed if in the last station the V-layer's ladders are fixed in place of X2 ladders

freedom. Added on top of these modes are several weak modes that are due to the split of IT detection-layer system in quadrants. The latter modes are fixed by the overlap tracks, otherwise they would be true global modes of the system. Having used the Particle Gun data of 100 GeV muons, the extra biases coming from the low energy component of an event are avoided.

After diagonalization, the eigenvectors associated with the lowest 4 eigenvalues have been retrieved. A graphical representation is plotted in the Figures 5.40(a) to 5.41(b), in order from the lowest to the highest eigenvalue.

On the graphs x-axis, the Layer index spans the usual ordering of the IT layers, identical with one in the LHCb database:
1) First layer is the 1st X layer of the Top Box of the first Station;
2) 2nd layer is the U layer of the same box;
3) 3rd layer is the V layer of the same box;
4) 4th layer is the 2nd X layer of the same box;
5-8) same order as before for the Bottom Box of the First Station;
9-12) same order for Left (A) Box of the first Station;
13-16) same order for Right (C) Box of the first Station;
17-32) same for the second Station layers;
33-48) same for the third Station layers.

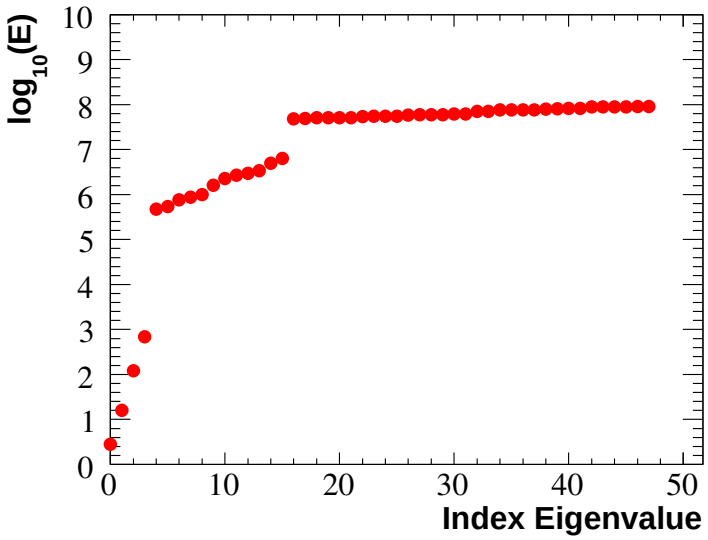
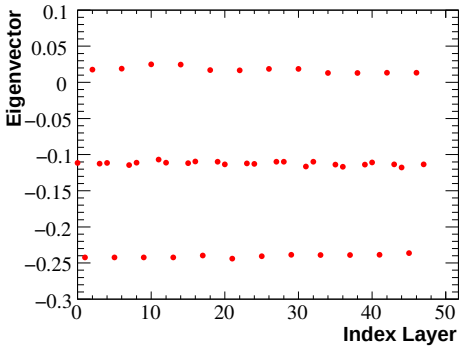


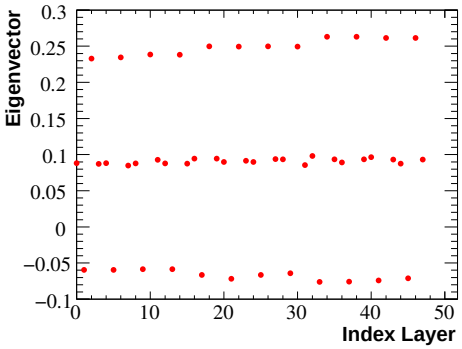
Figure 5.39: Spectrum of an alignment matrix when aligning only in $\delta\tau$

Figure 5.40(a) shows an eigenmode which corresponds to a mixing between a translation in x and one in y. In the quoted graph the x layer are shifted to -0.1 (in arbitrary units), and the U, V layers decouple to -0.25 shift in measurement direction for all U layers, and a 0.02 shift in measurement direction for V layers. The latter 2 shifts correspond to a single coherent y-shift for all IT layers. Of course the y shift is undetectable by the X-layers and the U and V layers perceive opposite shifts.

Figure 5.40(b) displays the complementary mode of the 5.40(a)'s mode. The polarity of the x and y shifts have reversed sign this time. Together, these modes make the 2 global x and y shifts of the IT layers.

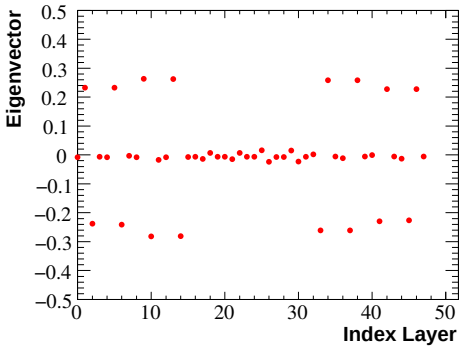


(a) Lest constrained eigenmode

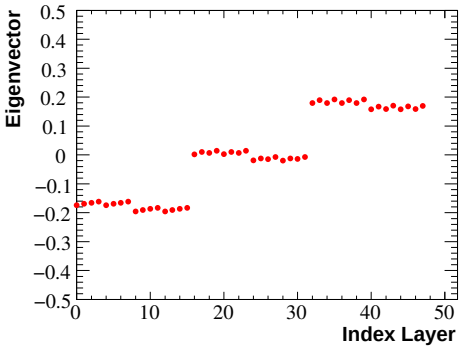


(b) Second eigenmode

Figure 5.40: Global translations as eigenmodes of the alignment matrix



(a) Third eigenmode



(b) The fourth global mode

Figure 5.41: Global shearings as eigenmodes of the alignment matrix

In the figure 5.41(a), the represented mode is a shearing in y. In the first station, layers' indices 0 to 15, the layers U and V "detect" a global shift in y. The second station is stationary, whereas the third station's U and V layers "see" a global translation in y this time reversed with respect to the first station. The mode is clearly a shearing with a fixed 2nd station.

In 5.40(b) the 4th global mode is a shearing in x, the layers in the first station, shift in x opposite direction with respect to the 3rd station's layers.

The plots of 5.40 and 5.41 are a nice proof for the usefulness of the algebra's matrix decomposition methods in alignment. As a general rule the matrix decomposition will provide the full global mode and weak mode description, yet, in most cases the labeling of the found mode will not be as trivial as in the described example. The weak modes and the global modes are most of the time mixed in decomposition, and a proper identification with the associated geometrical twist or displacement is difficult. In the next subsections, examples of the geometrical complexities of the weak-modes will be provided.

5.4.2 Weak Modes in Layer Alignment and the Minimally Constrained Setting

In this subsection I am using the Particle Gun data to align the IT layers when most of the constraints are removed. The split of the IT in quadrants makes the alignment a four-fold problem, as each side of IT is independently aligned. The partition, was caused by the relatively small number of tracks going

through the overlap areas between boxes. In total there are 12 overlap regions between boxes of the same station. The tracks passing through the common acceptance areas between boxes represent about 1 % of the number of total tracks detected by IT. For a sample equal to about 600 k muon tracks, the fraction is of about 6000 tracks. On the average, the connecting number of track between adiacent boxes is of about 1200. The measurement residuals of these tracks contain the only available information to align what are essentialy quasi-independent IT boxes. E.g., there are about 1000 tracks passing through the overlap region between the Top Box and the Left Box of the first station. A typical overlap region is shown in 5.42, with the sensors which cover the overlap acceptance highlighted in bright red.

When aligning 2 types of geometrical degrees of freedom, $\delta\alpha$ and $\delta\tau$, the entire IT has 5 global degree of freedom. There are 2 translation and 2 shearings in x and y, which affect the $\delta\tau$ alignment parameters, and 1 global rotation for $\delta\alpha$. I have chosen to fix the 5 globals by a priori constraining 5 alignment parameters of the Left-Boxes, namely:
1) the $\delta\tau$'s of the 1st X layer and of the U layer in the first station, and those for the V layer and the 2nd X layer in the third station;
2) the $\delta\alpha$ of the 2nd X layer in the 3rd station - it is the most down-stream layer in the Left Boxes.

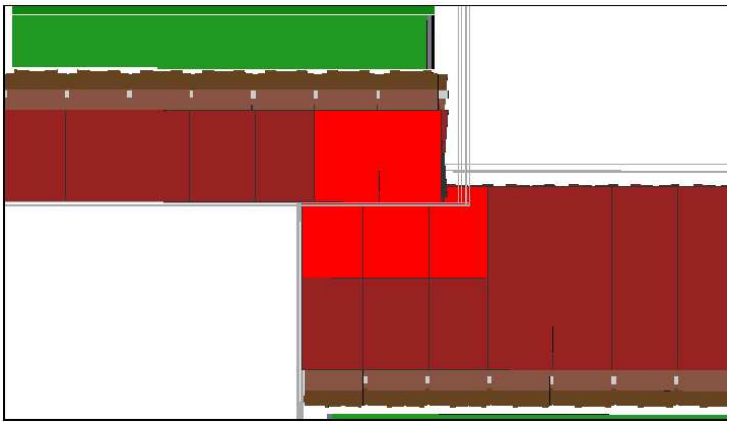


Figure 5.42: Overlapping acceptance area between Top-Box and Right-Box in the first station

The chosen set of constraints fix all the global degree of freedom for IT. The remaining modes that prevent the alignment of IT, are all weak modes, and generated by the insufficient information carried by the overlap tracks. The alignment is usually best when the track parameters are allowed to fluctuate between values as large as possible, e. g. a maximum x-slope t_x allows a better determination of the alignment $\delta\alpha$ parameter[27]. Due to the very restrictive acceptance, the overlap track have almost fixed slopes. Taking into account that the track parameters x_0 and y_0 are essentially null – as most tracks originate close to the primary vertex – then the overlap track parameters are almost fixed. This restriction of the track parameters to a very restrictive range, is highly damaging to the alignment of one quadrant of IT with respect to an other quadrant.

Two Scales in the Alignment Matrix

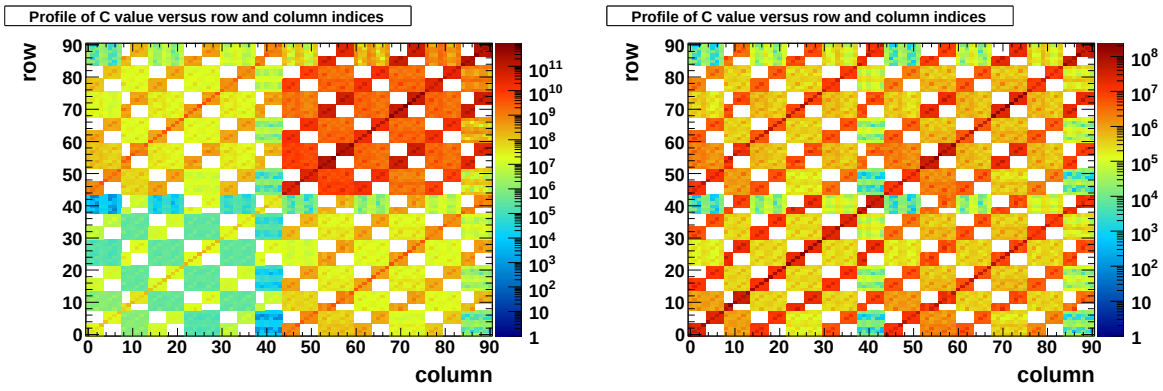
The two types of alignment parameters $\delta\tau$ and $\delta\alpha$ have different dimensions. One is simply a shift in space, whereas the other is a rotation and defined in terms of radians. Both parameters are to be determined simultaneously, and as consequence the alignment matrix is split into blocks with different dimensions.

In Figure 5.43(a) the alignment $\mathbf{C}'$ matrix is split in 2 block diagonal sub-matrices and 2 off-diagonal blocks. Of these, the first sub-matrix on the diagonal is non-dimensional, the off-diagonal blocks have dimensions of $\mathrm{m} \times \mathrm{rad}^{-1}$, and the last diagonal term has a dimension of $\mathrm{m}^2 \times \mathrm{rad}^{-2}$. In equation 5.11

[27]This aspect is evident if we look in the simple model function 4.1

the dimension of each block is displayed. The equation is the dimensional representation of matrix equation 3.17, which inverted gives the alignment parameters $\mathbf{a} = (\mathbf{C}')^{-1} \times \mathbf{b}'$.

$$\begin{pmatrix} \begin{pmatrix} 1 \end{pmatrix} & \begin{pmatrix} m \times rad^{-1} \end{pmatrix} \\ \begin{pmatrix} m \times rad^{-1} \end{pmatrix} & \begin{pmatrix} m^2 \times rad^{-2} \end{pmatrix} \end{pmatrix} \times \begin{pmatrix} \begin{pmatrix} m \end{pmatrix} \\ \begin{pmatrix} rad \end{pmatrix} \end{pmatrix} = \begin{pmatrix} \begin{pmatrix} m \end{pmatrix} \\ \begin{pmatrix} m^2 rad^{-1} \end{pmatrix} \end{pmatrix} \tag{5.11}$$



(a) Unnormalized alignment matrix

(b) Normalized alignment matrix

Figure 5.43: Normalized and unnormalized alignment matrices for 91 non-fixed alignment parameters, 47 $\delta\alpha$, and 44 $\delta\tau$, matrix "axis y" is reversed

It can be immediately noticed in figure 5.43(a) the difference in values between the upper and lower diagonal blocks. The fact that in average the measurement model derivative with respect to the $\delta\alpha$ parameters is 60 times larger than the derivative with respect to $\delta\tau$'s, is the cause of the scale difference between the upper and the lower terms[28].

If an eigenvalue decomposition is done and the eigenvectors of the weak modes are retrieved, it would be difficult to ascertain which are the alignment parameters that give the major contribution to the mode. The $\delta\tau$ components would be on the average 60 times larger than the $\delta\alpha$ components.

As a first step, to insure that the $\delta\alpha$ parameters contribution to the weak modes is not underestimated, I multiply the radian values with a lever arm of 60 mm. This amounts to: divide each element in the matrix 5.43(a) by 60 if the element is in the off-diagonal blocks, and divide each element of the second diagonal block (the large values one) by 3600. The resulting normalized matrix is shown in the graphical display 5.43(b). In this graph is clear that the split in 4 blocks, each with a different scale, has almost been avoided.

Initially I was applying this method of normalizing, only to make the radian dimension disappear and have every values in mm or derivatives of this dimension. The outlined procedure was not the only one used, in some cases the average factor of 60 was replaced by the mean value of the model derivative with respect to an alignment parameter. This is equivalent to multiplying the alignment matrix with two diagonal matrices, each diagonal matrix with the inverse of the derivative mean values on the diagonal. The equations behind this algorithm are in 5.12 and 5.13.

[28] In reality the multiplying factor of 60 refers only to the layers with long ladders and double-sensors, whereas for layers with short ladders and single sensors the multiplying factor is 30.

$$\mathbf{C}' \times \mathbf{a} = \mathbf{b}' \Longleftrightarrow \mathbf{D}\mathbf{C}'\mathbf{D} \times \mathbf{D}^{-1}\mathbf{a} = \mathbf{D}\mathbf{b}' \tag{5.12}$$

with

$$\mathbf{D} = \begin{pmatrix} \left\langle \frac{\partial U_{model}}{\partial a_1} \right\rangle^{-1} & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \left\langle \frac{\partial U_{model}}{\partial a_2} \right\rangle^{-1} & 0 & \cdots & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & \cdots & \cdots & 0 & \left\langle \frac{\partial U_{model}}{\partial a_n} \right\rangle^{-1} \end{pmatrix} \tag{5.13}$$

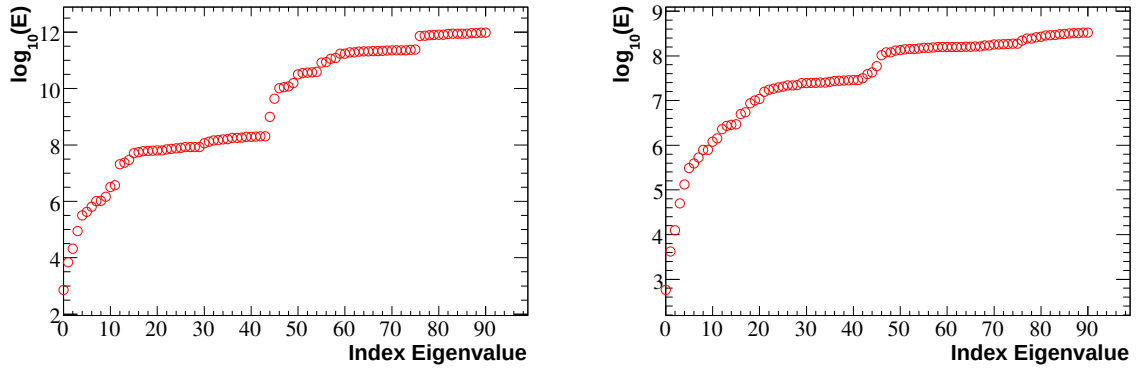
I am solving for $\mathbf{D}^{-1}\mathbf{a}$, but this is not a problem since I have defined the diagonal matrix to be consistent with a change in dimension or scale of the alignment parameters.

Later, I found out that there is an additional mathematical fundament for this approach, see [35]. The method can be viewed as a preconditioning method, which makes the matrix eigenvalues more meaningful, such that there is no dimensional difference. Why I am allowed to apply this normalizing procedure in the first place, is explained nicely in reference [36]. I shall try to sketch the reasoning given there.

If $\mathbf{v}_i$ are eigenvectors of the new alignment matrix $\mathbf{D}\mathbf{C}'\mathbf{D}$ with eigenvalues e_i :

$$\left(\mathbf{D}\mathbf{C}'\mathbf{D}\right)\mathbf{v}_i = e_i\mathbf{v}_i \Longleftrightarrow \mathbf{C}'\left(\mathbf{D}\mathbf{v}_i\right) = e_i\left(\mathbf{D}^{-1}\mathbf{v}_i\right) \tag{5.14}$$

If $\mathbf{D}$ diagonal matrix is orthogonal $\mathbf{D}^{-1} = \mathbf{D} = \mathbb{1}$, than the eigenvectors of $(\mathbf{D}\mathbf{C}'\mathbf{D})$, are eigenvectors of the $\mathbf{C}'$. Of course, this is not true in general, and not true in the case at hand. However for eigenvalues close to null, as it is the case of global and weak modes, for every eigenvector $\mathbf{v}_i$ of $\mathbf{D}\mathbf{C}'\mathbf{D}$ with an almost null eigenvalue, there is an eigenvector $\mathbf{D}^{-1}\mathbf{v}_i$ of the alignment matrix $\mathbf{C}'$ with a eigenvalue close to null. Hence the transformation 5.12 does not change the behavior of the eigenvalue spectrum close to null values. This is exemplified in the spectra of figures 5.44(a) and 5.44(b).



(a) Spectrum of the unnormalized alignment matrix

(b) Spectrum of the normalized alignment matrix

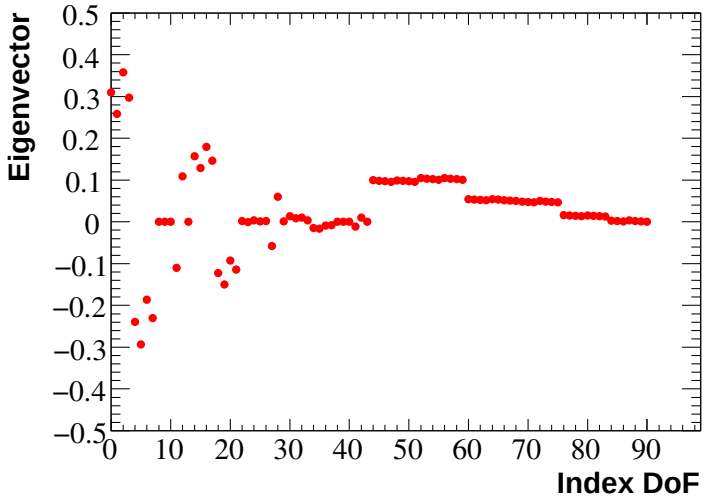
Figure 5.44: Alignment matrix spectra for 91 non-fixed alignment parameters, 47 $\delta\alpha$, and 44 $\delta\tau$

Going back to the matrices 5.43(a) and 5.43(b), the eigenvalue spectra are given in the figures 5.44(a), and 5.44(b), respectively. The number of eigenvalues with the lowest value in both matrices is identical, only the other-end of the spectrum has changed from one picture to the other.

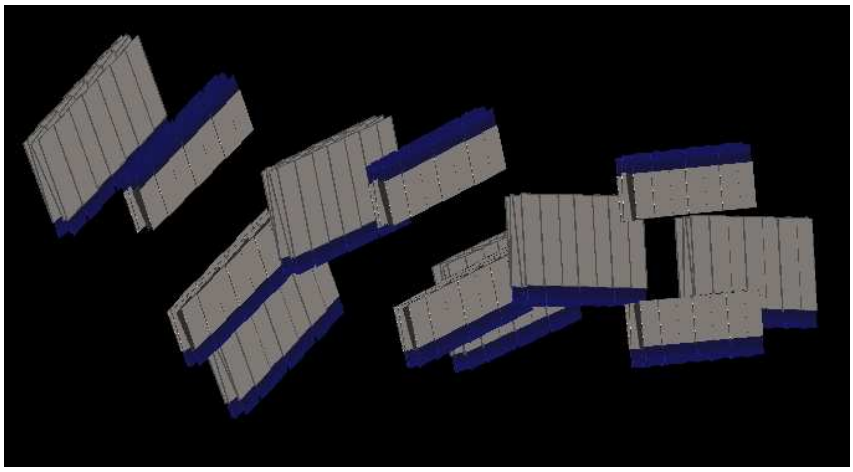
On top of the already constrained global modes, there are at least 4 weak modes that are damaging the alignment results. As explained already, these weak modes are the result of the insufficient information carried by the overlap tracks, which prevents aligning between quadrants of IT.

4 Weak Modes

Extracting the eigenvectors by diagonalizing the matrix 5.43(b), we can make some visual representations of the found weak modes 5.45(a), 8.17(a) , 8.18(a), and 8.19(a). The latter 3 weak modes are provided in Appendix.



(a) 1st Weak Mode, eigenvector



(b) 1st Weak Mode, Panoramix display

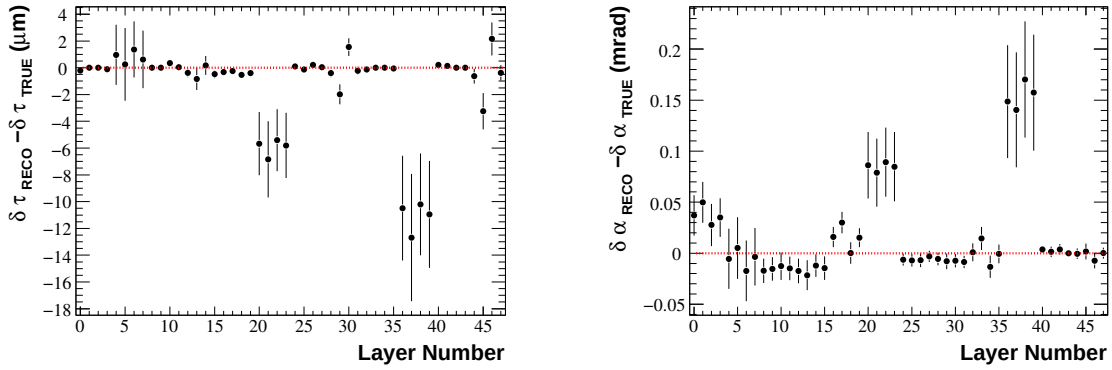
Figure 5.45: Alignment matrix' first eigenvector with 91 non-fixed alignment parameters, 44 $\delta\tau$ and 47 $\delta\alpha$ ($\delta\alpha$'s DoF after the $\delta\tau$'s), it corresponds to the lowest eigenvalue

The nice Panoramix display of the weak-modes is given in 5.45(b), 8.17(b), 8.18(b), 8.19(b), with the scale of the misalignments arbitrarily enhanced by multiplying with a scale factor larger than 100. These weak-modes are clearly an effect of twists in the box-quadrant structure of IT. Even though the alignment was done at layer level, the layers of the same box remain stationary one with respect to other.

Alignment with 87 Alignment Parameters

To provide an alignment when constraining 9 global and weak modes, it was found useful to fix in addition to the previous 5 alignment parameters, other 4 parameters. The a priori fixed alignment parameters are for $\delta\tau$: 1st X layer of the Left Box and first station, U layer of the Left Box and first station, V layer of the Left Box and third station, 2nd X layer of the Left Box and third station, U and V layers for Top-Box and first station, U and V layers for Top-Box and third station. The a priori fixed $\delta\alpha$ corresponds to the z-rotation of the 2nd X layer in the Left Box of third station. In total, there are 9 a priori fixed geometrical degrees of freedom or alignment parameters.

The results of alignment after 30 Newton-Raphson steps are outlined in plots 5.46(a) and 5.46(b). It is apparent that for the Bottom-Box one more constraint would be helpful, yet, since the Millepede error estimates seem to be consistent with the large values observed, I think that this residual weak-mode can be removed by adding more overlap-tracks to the sample. For real data is already under-consideration within the LHCb Alignment group, to submit a proposal regarding the storage of these type of tracks in dedicated data containers, hence easily accessible for alignment. If a large overlap-track sample will not be available, an additional parameter has to be constrained, in order to get the alignment precision to within 1 μm between Bottom-Boxes quadrant and the other quadrants.



(a) $\delta\tau$ resolution plot

(b) $\delta\alpha$ resolution plot

Figure 5.46: Alignment results for 87 non-fixed alignment parameters, 47 $\delta\alpha$'s , and 40 $\delta\tau$'s

The iterations are stable, sign that the remaining weak mode is not damaging the alignment, but instead is just making the resolution of Bottom-Boxes a little worse than expected.

Just as an oddity, it is worth mentioning that in this alignment, the reduced χ^2 in the final iteration is close to 0.999 for approximately 4 Million Degrees of freedom. This would make this alignment the most precise alignment I ever performed, with a caveat that the track-χ^2 cut removes the tails of the measurement residuals, hence a value of 1 is not exactly what we should expect...

The iterative evolution plots of 4 alignment parameters are given in 5.48(a) and 5.48(b). Looking carefully, there is some extremely small non-linear fluctuation, as an effect of the residual weak mode.

In conclusion I have observed alignment convergence achieved for only 9 a priori fixed alignment parameters. Except the Bottom Boxes, all other layers were fixed within micrometer precision. An increase of 10 times in the number of overlap-tracks would get down the Bottom-Box alignment resolution from about 5 μm to one micron.

If some of the 9 constraints are relaxed, I have observed that the Newton-Raphson procedure without quasi-Newton steps generates results which are divergent. It signals that the 4 weak modes must be constrained. It remains the task of the global alignment to find the positions corresponding to the constrained layers in the frame of an external detector, e.g., VELO plus TT.

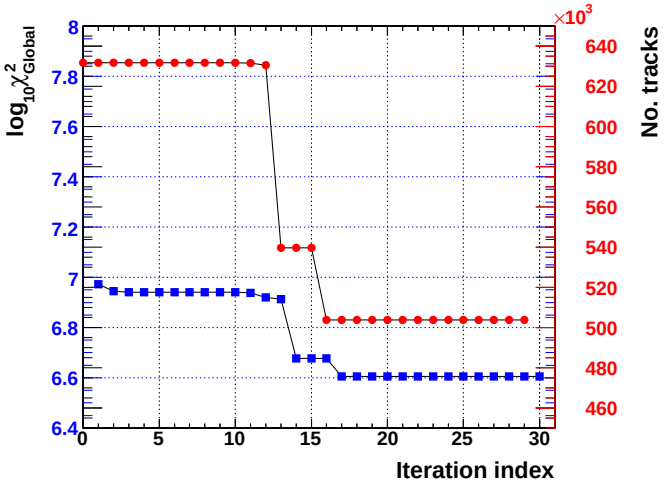
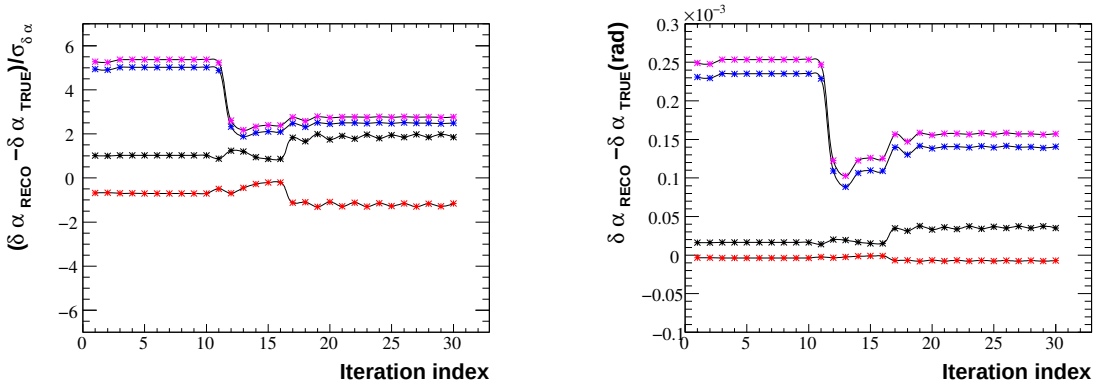


Figure 5.47: Alignment χ^2 and number of tracks as they evolve with each iteration



(a) $\delta\alpha$ iterative pulls

(b) $\delta\alpha$ iterative evolution

Figure 5.48: Alignment parameters' variation with iteration, display of convergency

5.4.3 Weak Modes and Alignment Matrix for an Alignment with 5 Degrees of Freedom per Layer

Coming back to the alignment in all 5 degrees of freedom of the subsection 5.2.4, it is possible to look whether there are residual weak-mode which could explain the slightly larger alignment deviation than the one expected according to Millepede's error propagation.

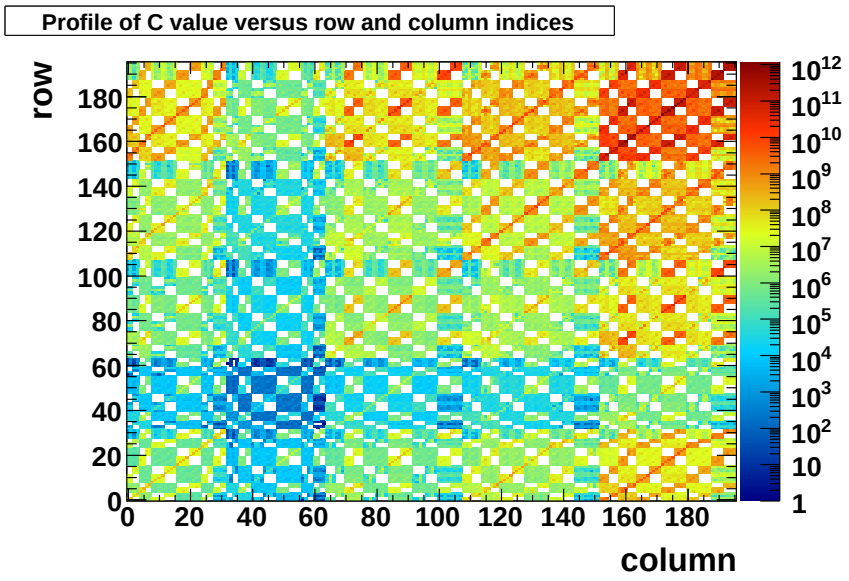
The Millepede unnormalized matrix, $\mathbf{C}'$, is given in the 2D-profile histogram of figure 5.49(a). In this subsection I apply in place of my usual set of normalization choices the normalization described in the previously mentioned reference [36]. In this case the $\mathbf{D}$ matrix of the equation 5.12 is a diagonal matrix with:

$$D_{ii} = \sqrt{\frac{N_i}{C'_{ii}}} \tag{5.15}$$

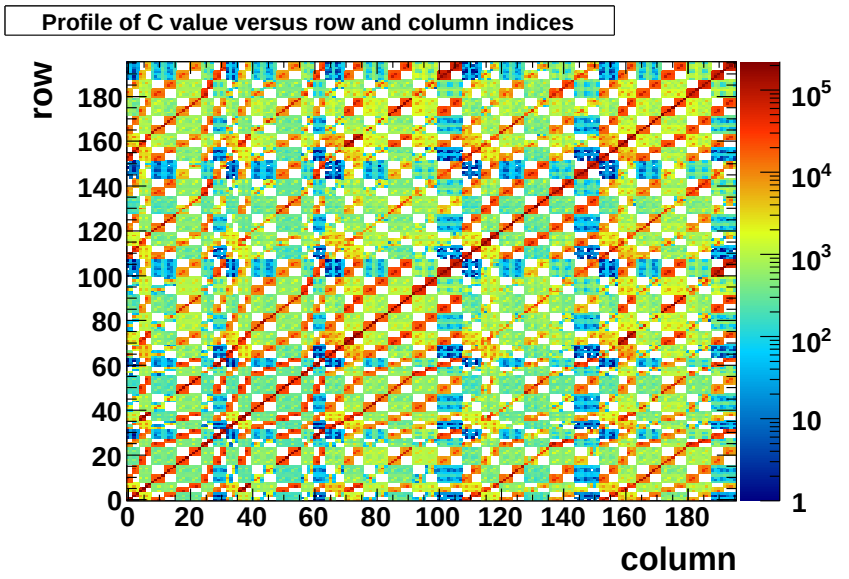
The number N_i is the number of measurements that contribute to the value of the diagonal term C'_{ii} of alignment matrix matrix $\mathbf{C}'$. As it can be inferred from equation 3.17, each element of the alignment

matrix is equal to a sum over track and measurement indices, hence all measurements that add to this sum a non-null value, are in number equal to N_i.

Having retrieved form the data in addition to the usual alignment information, the vector of measurement numbers $(N_0, N_1, \cdots, N_n)^T$, it is possible to apply the normalization scheme. The result is given in 5.49(b). Initially, blocks corresponding to distinct types of alignment parameter have very different values one with respect to the other. The 5 blocks are easily distinguished in 5.49(a) on the diagonal (carefully as I am plotting the row numbers in reversed).



(a) Unnormalized Alignment Matrix



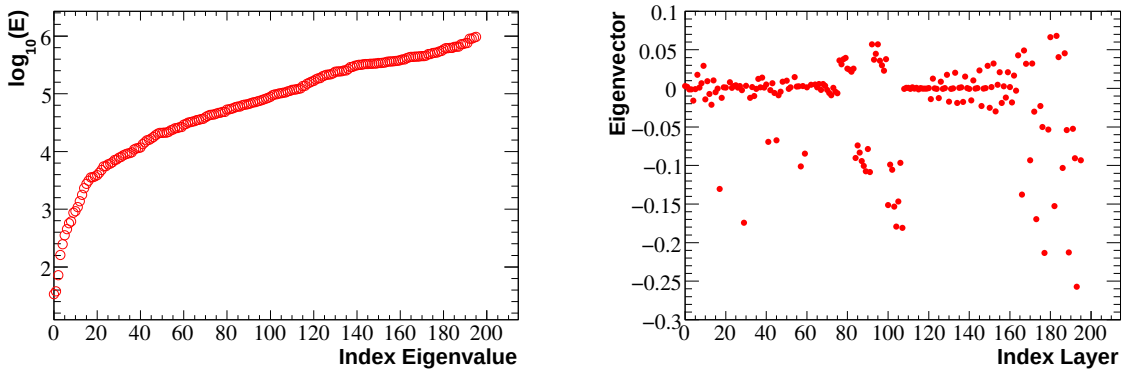
(b) Normalized Alignment Matrix

Figure 5.49: Alignment matrix for 5 degrees of freedom per Layer, 32 $\delta\tau$'s , 32 δz, 44 $\delta\gamma$'s , 44 $\delta\beta$'s, 44 $\delta\alpha$'s, matrix "axis y" is reversed

The blocks ordering is equivalent to the order of alignment parameters: $\delta\tau$'s, δz's, $\delta\gamma$'s, $\delta\beta$'s, $\delta\alpha$'s . The higher values are observed in the block of the $\delta\alpha$ and the lowest values in the δz block which is all dark-blue.

After normalization, the matrix 5.49(b) is much more homogeneous. The normalization here does more then just assigning to the rotational degrees of freedom a lever arm. This time the eigenvalue j can be reinterpreted as the "effective number of hits contributing to the alignment eigenmode j", $\forall$ j smaller than number of alignment parameters which are not fixed.

The eigenmode decomposition of the normalized matrix provides the usual insightful information on the alignment weak-modes. In the matrix spectrum 5.50(a) the lowest lying eigenvalue is associated with the potential weak-mode eigenvector displayed in 5.50(b) .



(a) Alignment spectrum when solving for 196 alignment parameters

(b) Least constrained eigenmode

Figure 5.50: Spectrum and the least constrained eigenmode of the normalized alignment matrix.

In figure 5.50(a), the lowest eigenvalue indicates that at least a residual weak mode appeared as result of a mixing between mostly the $\delta\alpha$ degrees of freedom and the $\delta\gamma$'s, thus rotations around z axis, and x axis, respectively. The spread in the latter two alignment parameters is visible in 5.50(b), where the $\delta\gamma$'s have an index between 64 and 107, and the $\delta\alpha$'s between 152 and 195. The successive 2 eigenmodes show similar trends, all pointing to additional constraints required for 3 or 4 of the $\delta\gamma$ alignment parameters[29].

The scale of $\delta\gamma$ in figure 5.50(b) is at least 40 times larger than $\delta\alpha$ if the normalization procedure would not have been applied. Hence, in spite of 5.50(b) values, the $\delta\gamma$ component is clearly the dominant component for all 4 residual weak modes.

[29] 4 a priori fixed $\delta\gamma$ are needed, because of the quadrant structure of the IT.

Chapter 6

Alignment Results in the Presence of Magnetic Field

6.1 Alignment with Full Samples

As it will be shown later in this chapter, the initial track sample used suffers from a large anisotropy in charge distribution. The usual track sample for IT can be partitioned in left-side sample and right-side sample, with almost all the positive tracks passing through the left-side and almost all negative charges passing through the right-side.

With the track-cubic model precise down to the level of measurement resolution of about 50 $\mu\mathrm{m}$, the model biases are small only if we use for a given sensor the same amount of positive-charge tracks as negative-charge tracks.

In the first part of this chapter there are given the alignment results for the full track samples. In the second part, I remove the excess of positive-charged tracks from the Left-Side of the IT, and the excess of negative-charge tracks from the Right-Side.

$$x = x_0 + t_x * (z - z_{ref}) + Q\left[(z - z_{ref})^2 + d * (z - z_{ref})^3\right] \tag{6.1}$$

The model biases are explainable by mostly two facts:
1) the model neglects the exact magnetic field map, and sums the magnetic field effects on the average terms Q and d, see the track model function in equation 6.1;
2) the model is essentially an asymptotic double extrapolation from the magnet region and form the IT's field free third station.
A Runge-Kutta calculus over the trajectory's path of the particle would remove any inconsistency, but unfortunately the track extrapolator based on the Fifth order Runge-Kutta method is much too slow to allow a fast alignment.

Naturally, it is safe to assume that the usual biases generated by random scattering and bremsstrahlung are still present in magnetic field.

I made the choice to use this model because, as I shall show, it is possible by a careful selection of tracks to make the measurement-residual distributions unbiased. In this way the effects of random scattering and model biases all but disappear, and the alignment parameters are correctly computed. Up to micrometer precision, the error propagation was found to be consistent, too.

6.1.1 A priori Constrains, Globals

From equation 6.1 it is easily to infer the existence of additional global degrees of freedom for the system. The 5th extra track-parameter, couples with the alignment parameters in the model function in a similar way as the slope t_x did in the simple alignment model of eqution 4.1. If for simplicity, the stereo angle

is neglected, and as a first approximation all layers are taken as X layers, than the alignment model has the following component, with z origin shifted in the middle of the second station:

$$U_{model} = x_0^i + t_x^i z + Q_i z^2 + \delta\tau + O(z^3) \tag{6.2}$$

Assume for now that the cubic term is not as important as the parabolic term[1]. It is straightforward to see that the following generic transformation leaves the alignment-χ^2 unchanged.

$$\left\{ \begin{array}{lcll} \delta\tau & \longrightarrow & \delta\tau + \delta\tau_0 \times z^2 & \\ Q_i & \longrightarrow & Q_i - \delta\tau_0 & \forall\, i \end{array} \right. \implies \text{invariant} \quad \chi^2 \tag{6.3}$$

This is, what I usually call the "parabolic shearing"[2], and it is a true global degree of freedom for the IT system. In the following section the effects of this global mode will be seen in the form of a residual weak mode. In this chapter, if not explicitly stated otherwise, I use the same set of constrains as for the case of no magnetic field. The parabolic shearing mode gets fixed by the tracks connecting IT-boxes of different quadrants. E.g. if a track passes through the overlap region between Left-Box and Top-Box and these quadrants are independently constrained, then the track passes through 8 fixed layers (if only $\delta\tau$ is aligned). The 8 fixed layers are more than sufficient to remove the "parabolic shearing" mode. However, since the tracks through the overlap region are only a percent of the total, care must be exercised, as the global mode might not be simply removed but merely undergoing a transformation from a true global degree of freedom to a "weak" mode that will only perturb the alignment. The effect of the transformation will be apparent in the next sections.

A second possible global mode might appear due to the mixing between the alignment parameter $\delta\alpha$ and the track parameter Q. If we take again only x-layers the simplest model function is:

$$U_{model} = \left(t_y^i z + y_0^i\right) \times \delta\alpha + Q_i z^2 + t_x^i z + x_0^i \tag{6.4}$$

the following mode keeps the global alignment χ^2 invariant:

$$\left\{ \begin{array}{lcll} Q_i & \longrightarrow & Q_i - t_y^i\, \delta\alpha_0 & \forall\, i \\ t_x^i & \longrightarrow & t_x^i - y_0^i\, \delta\alpha_0 & \forall\, i \\ \delta\alpha & \longrightarrow & z\, \delta\alpha_0 + \delta\alpha & \end{array} \right. \implies \text{invariant} \quad \chi^2 \tag{6.5}$$

The latter mode is partially removed, if the stereo angle sequence is considered. It appears that this type of geometrical twist, has undergone transformation to a weak mode, too. In the case of layer alignment the mode is not very visible in the reconstructed values. It is in the case of ladder alignment, where some of the weak modes that I observed belong this type. Severe perturbations in the $\delta\alpha$ reconstructed value were seen in those cases, and the effect disappeared when constraints were added for some of the non-a priori fixed $\delta\alpha$.

6.1.2 Layer Alignment for the Most Sensitive Alignment Parameters

As I explained in the chapter 5, the alignment is best performed for the two types of alignment parameters $\delta\tau$ and $\delta\alpha$ - i.e. the shift in the measurement direction and the rotation around the LHCb z axis, with pivot in center of the layer (or ladder if ladders are aligned).

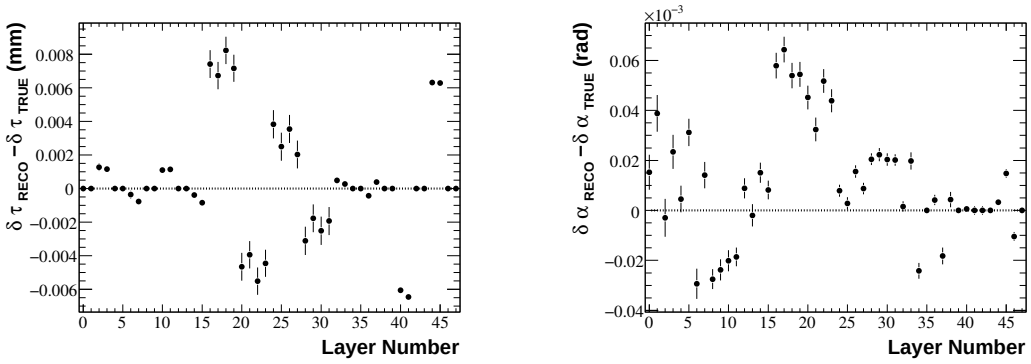
Here I am using 1 Million Minimum Bias simulated events. The data is part of the default data used by the LHCb physics group to predict the LHCb physics for Minimum Bias events.

The constraints used are:
- in $\delta\tau$ the first and the last two layers in each quadrant;
- in $\delta\alpha$ the last layer of each quadrant.

[1] In the last station this is not quite true, but if the cubic term is included it will not change the final qualitative result that I want to reach.

[2] I use this name because the mode resembles a shearing with a variable scaling factor linear in z

The weak mode and the model bias effects are seen in both resolution plots 6.1(a) and 6.1(b). The Millepede error estimates are gross underestimation of the observed error.



(a) $\delta\tau$ degrees of freedom

(b) $\delta\alpha$ degrees of freedom

Figure 6.1: Alignment resolutions with Millepede errors estimates, for 2 types of alignment parameters

Yet, the result are not too bad. The found average alignment error is: for $\delta\tau$ the average over all layers for this run is 4 μm, and for $\delta\alpha$ the average over all layers is 27 micro-radians.

The Millepede error estimates for $\delta\tau$ are in average 22 times smaller than the observed errors in 6.1(a). For $\delta\alpha$ the factor is about 5. The error propagation fails because of two interconnected reasons. The track-χ^2 cut is less efficient in dealing with random scattering because the alignment results are perturbed by the model biases. The model bias on the other hand pulls apart the left and the right side of IT, because of the charge asymmetry. The latter effect is easily seen in 6.1(a) where 4 of the last 8 dots on the right are pulled apart. These correspond to the non-fixed alignment parameters of the Left and Right Boxes in the third station.

On top of these interconnected effects and causes, there are the parabolic weak modes, the parabolic shearings and the rotational twists, see subsection 6.1.1.

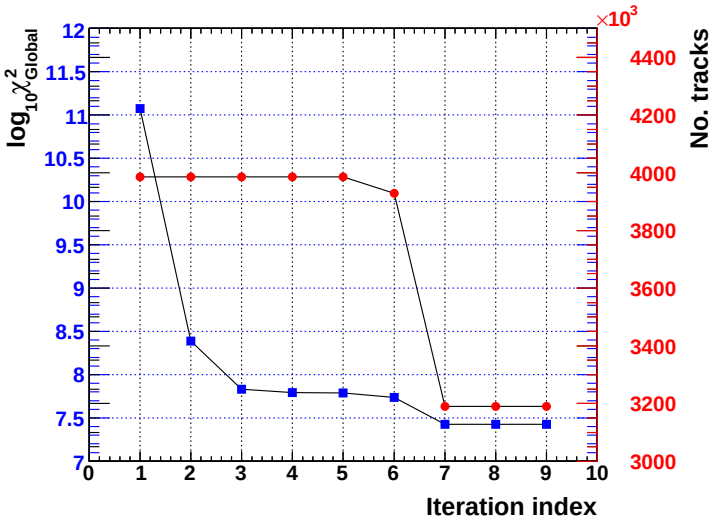


Figure 6.2: Convergence, and evolution plot for the global-χ^2 and the number of tracks used in alignment

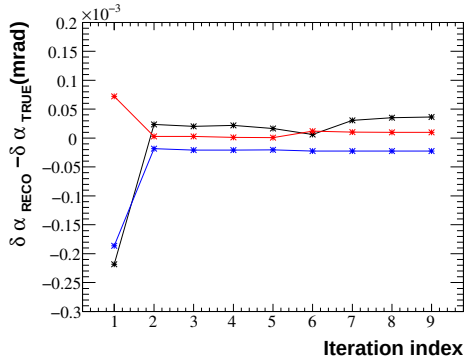
In these conditions, it is remarkable that the normal-standard Newton-Raphson procedure without

quasi-Newton steps converges to good values. Except for the left/right boxes in the third station where the model bias dominates, the layers alignment inside boxes is precise within micrometer level, and the boxes themselves are aligned with a precision of about 5 microns.

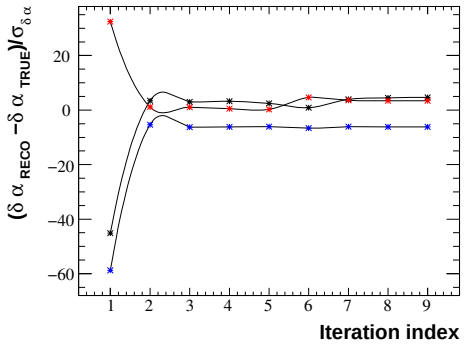
Alignment in $\delta\alpha$ are not bad too. The precision of 30 μrad is equivalent to fluctuations in the position of layer's left/right edges of about 7 microns. I assumed a quarter meter lever arm.

In the last iteration the reduced global χ^2 of the alignment is 1.190, with 22.33 Millions Degrees of freedom for the fit. The very large difference with respect to 1, is due to the model biases, random scattering and residual weak modes, which bias and broaden the measurement residual distributions for each sensor.

Convergence is achieved extremely fast, as the quasi-Newton steps are absent for this alignment. Only 2 steps are necessary for reaching the convergence plateaus in both alignment-χ^2 (fig. 6.2) and alignment parameters (fig. 6.3(a) and 6.3(b)). The track-χ^2 cut is active after the 5th iteration, where the swap from one convergence plateau to the other takes place.



(a) iterative evolution of the error



(b) iterative evolution of the error scaled with the Millepede error estimates

Figure 6.3: Iterative evolution of the error for 3 alignment parameters, all rotations around z axis.

Unfortunately, as the result of large model biases, the track-χ^2 cut losses its usual potency, and the results are essentially unaffected.

6.1.3 Alignment of all 5 Alignment Parameters for each Layer

In this subsection it will be shown that the effect of large fluctuations in the two least sensitive alignment parameters $\delta\gamma$ and δz has little impact on the alignment of the other three geometrical degrees of freedom.

To exemplify, I choose to align in all 5 degrees of freedom the IT layers. The IT split in quadrants is preserved and the constraints applied as usual for $\delta\tau$, $\delta\alpha$ (see previous subsection). $\delta\gamma$ and $\delta\beta$ are fixed for the same layer as $\delta\alpha$, the last in each quadrant.

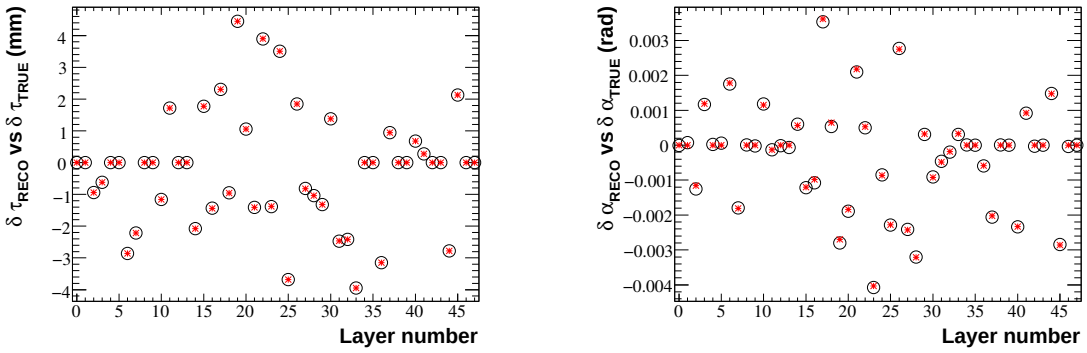
To fix the arbitrary global scale in z and the global translation in z, the first and the last layers in each quadrant are a priori fixed. In addition for the same degree of freedom δz, a potential weak mode is constrained by a priori fixing the second layer in each quadrant. The z alignment parameter and the Q track parameter are highly correlated. In a similar way as for the "parabolic shearing", the z alignment parameter has a weak mode in each quadrant.

The track sample used now is a Minimum Bias sample from the same source as the sample used in the previous alignment. This time there are less events in the sample: 340000. The magnetic field is set to the nominal non-null value.

The alignment geometry used is a misaligned geometry, with the values generated randomly with gaussian. The misalignment values are fed to reconstruction Geometry Database and more exactly to

the Condition Data Base LHCBCOND. The alignment algorithm assumes the real geometry is the one generated by this method. If everything is consistent, the Nominal geometry should be retrieved back.

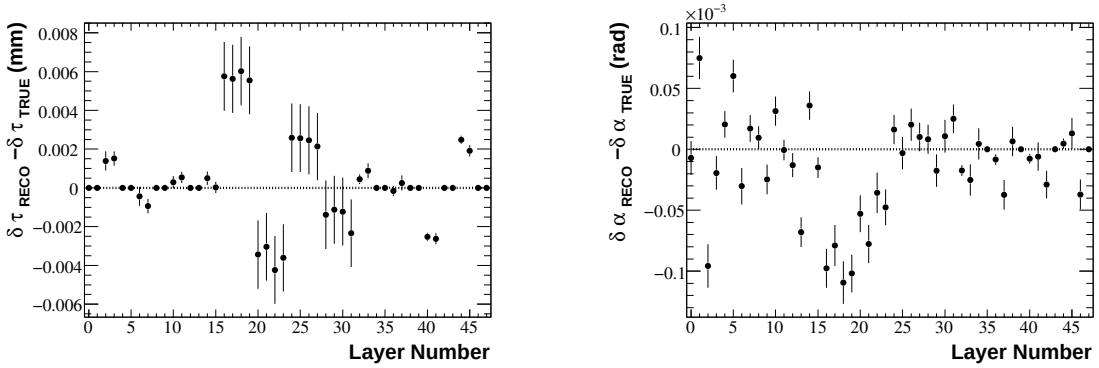
The input misalignment could be seen in figures 6.4(a), 6.4(b), and in the appendix figures 8.20(a), 8.20(b) and 8.22(b). The values span a range of millimeters for shifts and milli-radians for rotations.



(a) $\delta\tau$ degrees of freedom

(b) $\delta\alpha$ degrees of freedom

Figure 6.4: $\delta\alpha$ and $\delta\tau$ input misalignments versus the reconstructed alignment parameters, Millepede errors estimates are too small on this plot. Alignment is done for all 5 types of alignment parameters



(a) $\delta\tau$ degrees of freedom

(b) $\delta\alpha$ degrees of freedom

Figure 6.5: Alignment resolutions with Millepede errors estimates, for 2 types of alignment parameters given, a simultaneous alignment of all 5 DoF.

The alignment results for $\delta\alpha$ and $\delta\tau$ are reliable as seen in the differences between true and the reconstructed values. The usual weak-parabolic shearing is seen in 6.5(a), and the usual model biases in the left/right boxes of the third station (last layer indices, for the non-fixed parameters).

The non-linear parameter $\delta\alpha$ is perturbed by weak modes and model biases, too. In 6.5(b) the central indices planes – corresponding to the second station layers – have the tendency to fluctuate around the fixed third station. Model bias effects are observed in the first station (lower indices). The alignment precision, when averaging over all layers, is given in table 6.1, the error is defined by the difference between the reconstructed and the true values.

DoF	average error
$\delta\tau$	5 μm
$\delta\alpha$	41 μrad
$\delta\beta$	210 μrad
$\delta\gamma$	2.76 mrad
δz	0.57 mm

Table 6.1: Alignment precision when aligning in all 5 geometrical DoF

In this alignment run the track-χ^2 quality cut was not used. It was observed that in the first iterations this cut removes almost no track, and as the iterative steps proceed the cut removes all tracks in just few iterations. This is due to large non-linear fluctuation, amplified by the model biases, weak modes, and random scattering effects. As discussed in chapter 5, the addition of quasi-Newton steps does not offer stability for the track-χ^2 cut procedure. The quasi-Newton steps as defined in chapter 3 subsection 3.3.2, provide only convergent results for a given fixed sample. As the track-sample is changed from one iteration to the other, the method is not necessarily convergent. To obtain relevant results for $\delta\gamma$ and δz, it is required to eliminate most model biases and most random scattering effects. Additionally, the global parabolic shearing must be removed together with any potentially damaging weak-mode, by a careful choice of constraints.

The standard Newton-Raphson converges slower than previously, but fast enough. 4 iteration are needed to reach the convergence plateau in figure 6.6.

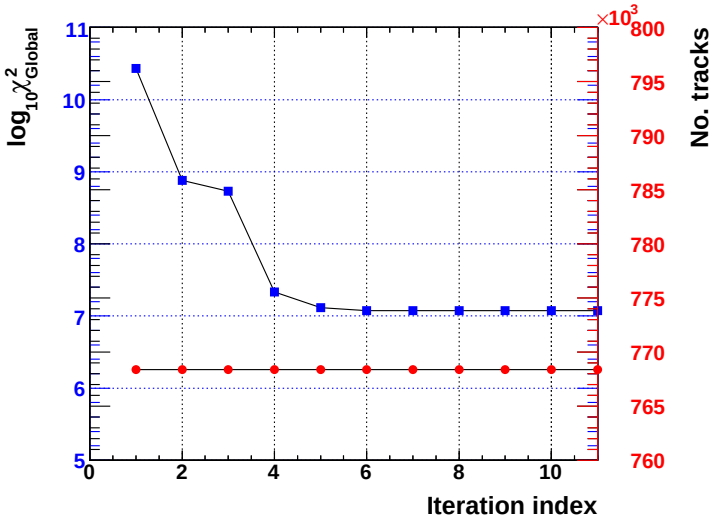


Figure 6.6: Alignment global-χ^2 evolution and the constant number of tracks in the absence of the track-χ^2 quality cut.

Without the track-χ^2 cut to clean the alignment track sample, the final reduced global-χ^2 of the alignment was about 2.2 for 5.4 Millions fit-degrees of freedom. Even though the alignment fails if we consider the $\delta\gamma$ values, it is remarkable that the most sensitive alignment parameters are found close to the values in the last subsection[3]. If the track number difference between the used two samples is considered, than a scale factor in the error estimates of 2.0 is found.

The results of the subsection, tell us that large uncertainties in rotation around x axis and in shifts along LHCb-z direction, will not give large effects in measurement residual distributions. Hence, it

[3]It is need to rescale the problem form 0.76 Million tracks to 3.19 Million tracks

should be decided by the Physics and Alignment groups if the two geometrical degrees of freedom are to be reconstructed or not, and if yes to what precision.

6.1.4 Ladder Alignment for the Most Sensitive Geometrical Degrees of Freedom

It is almost certain that only the most sensitive geometrical degrees of freedom of ladders will be reconstructible with reasonable precision. I am using again a sample with about 0.77 million tracks in nominal field from a standard LHCb-Physics Minimum Bias data.

The ladders in the first two layers of each IT quadrant are a priori fixed in their nominal positions, and the same holds true for the last two layers in each quadrant. These are the usual constraints used by the LHCb alignment group[4].

The results are given in figures 6.7 and 6.8 for each type of parameter. In $\delta\tau$'s plot the effect of parabolic shearing is immediately apparent, the central part of IT, at central indices of graph 6.7, tends to inflate as a balloon fish. Each 12-ladder stack is approximately independent in this weak mode, however the pattern suggest that some information gets passed between the neighboring stacks via the overlap tracks[5].

The Figure 6.7 is even more suggestive in the sense that in this representation, it is possible see the actual connection between the ladders of Left(Right) box which are practically "glued" to the Top/Bottom box ladders (the latter boxes have indices in a range centered on 150). The connection is realized by track passing through the overlap area between Top(Bottom) and Left(Right) quadrants of the second Station. This is a remarkable example of **the usefulness and power of the overlap-tracks in alignment**.

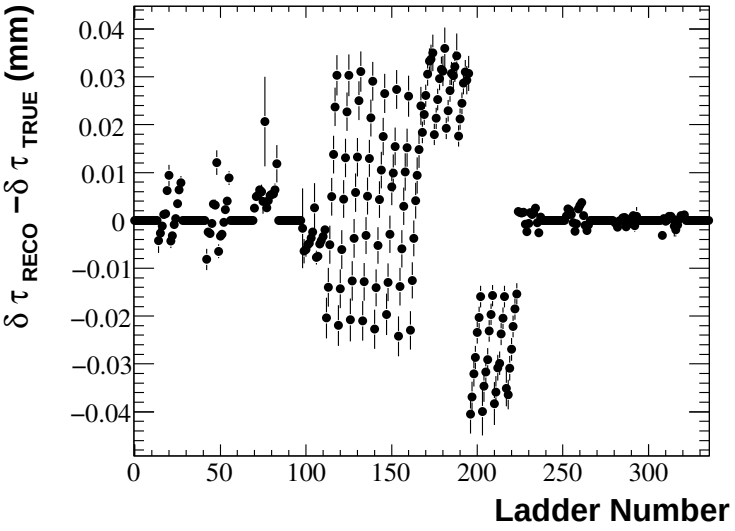


Figure 6.7: $\delta\tau$ alignment resolutions with Millepede errors estimates, alignment performed for 2 types of alignment parameters at ladder level.

In 6.8 the $\delta\alpha$ ladder alignment results show a severe deviation from the expected value, in the non-fixed ladders of the third station – this station is given from ladder index 223 to higher. The cause of this severe deviation is due to $\delta\alpha$ weak mode or modes, and I shall show in the next sections how it disappears when adding constraints. In addition to this cause, there might be a contribution from model biases and random scattering, but I am not sure quantitatively what role these play. Anyway, it is certain that the main contribution is from weak modes. A first indication of this is the fact that the affected ladders are

[4]Some of my co-workers use the Lagrange multiplier method, too.

[5]Overlap regions are now between neighboring ladders or sensors, in addition to the overlaps between ladders in the layer-quadrant overlap-region

U-V ladders, whereas the 1st X layer ladders are reconstructed precisely. The 2nd X layer in the third station has all ladders a priori fixed.

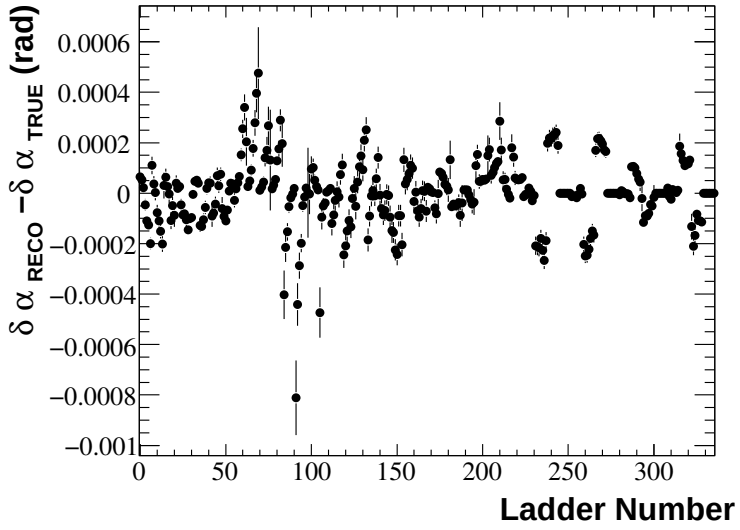


Figure 6.8: $\delta\alpha$ alignment resolutions with Millepede errors estimates, alignment performed for 2 types of alignment parameters at ladder level.

Effects of the $\delta\alpha$ weak modes are seen in the other stations too, but less severe, The tails of the reconstructed distribution are given by the most-outer-lying ladders, in Left and Right Boxes. These ladders are too far from the beam, hence their occupancy is relatively low compared with the central ladders by a few orders of magnitude.

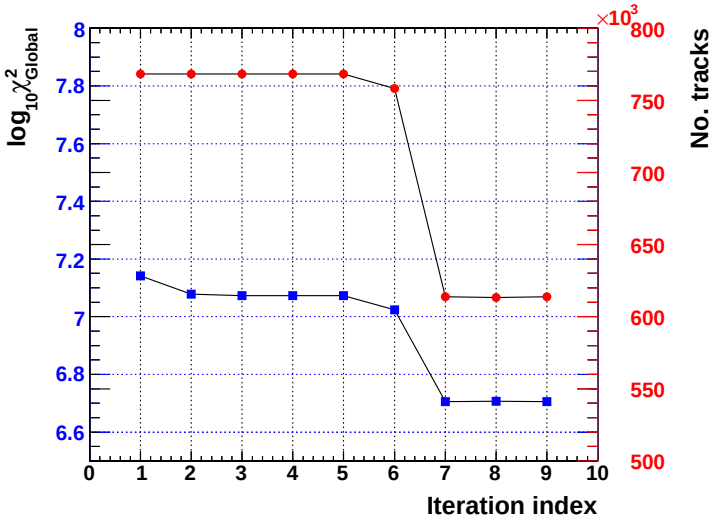


Figure 6.9: Alignment global-χ^2 evolution and the number of tracks used at each iteration, with track-χ^2 quality cut enabled after the 5th iteration.

The standard Newton-Raphson method plus track-χ^2 cuts is very fast. The convergence is immediate, and as the chosen misalignments were too small in this example their impact on the global χ^2 is negligible. Except for the track-χ^2 cut activation between 5th an 7th iteration, there is no major fluctuation in the global-χ^2.

In the end of the chapter I shall give an example where the alignment error propagation worked and all the weak modes were a priori constrained. Yet, given the few tracks through the overlap regions, the reconstructed values of this alignment were not bad, I even got some insightful knowledge on the weak modes, which I had not before. More tracks will be needed by far, and in the next example of ladder alignment this will be provided.

6.2 Anisotropy of the Charge Distribution Relative to the Sensor Position

About 2 Million tracks with 20 GeV momentum or higher, were used to map the charge anisotropy through the IT. The results are plotted in the 6.10 color-surface plots, where the relative occupancy in the phase-space of track parameters x_0 and y_0 is displayed.

Some track fits fail in 6.10, as the consequence that they have too few hits in U,V sensors[6]. The fit is done with only one or 2 U, V measurements, hence the track parameters in y get unreasonable values, outside IT acceptance. These kind of tracks are passing through the overlap regions between ladders or layers, in one Station, and then leave the IT fiducial volume. They are very few in one million Minimum Bias events, and are all rejected by the quality cuts of the alignment algorithm.

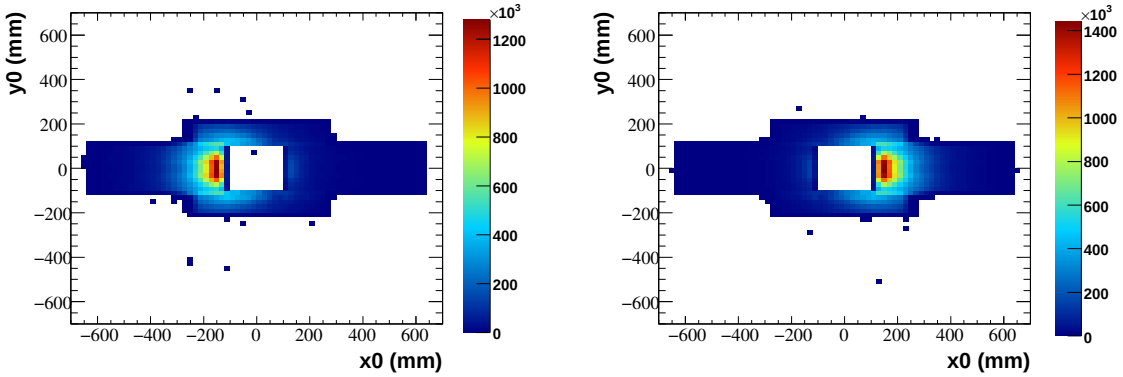


Figure 6.10: Positive charged tracks to the left of IT, negative charged tracks to the right of IT, color-surface plots in (x_0,y_0) track parameters.

Above a threshold of 20 GeV, about 92 % of the tracks passing through the left boxes are positive-charge tracks. The same is true for negative tracks through the right boxes. If the momentum cut is lowered at 5 GeV per particle, the fraction approaches 98 %. This huge charge anisotropy is responsable for most model biases observed in alignment results with magnetic field.

To avoid the charge anisotropy in the alignment track samples, an inverse map of 6.10 plots is done. I partition the phase space of x_0 and y_0 in rectangular unit cells of 2 cm length, and in each of these cells I remove the charge in excess. As a result each unit cell has on average a null charge current flow. This unfolding of the excess charge allows to keep in the alignment sample only those tracks that give an isotropic charge distribution in IT.

With the new sample the measurement-residual distributions and their means should not suffer from model biases, because the number of track of one charge being equal to the the number of tracks with the opposite charge, in the net result the model biases for opposite charges average-out. The only effect from model biases should be a broadening of the measurement residual distributions, and hence no impact on the reconstructed alignment parameters.

[6]Only two or one hits in a U or a V sensor, makes the fit impossible, because y track parameters t_y and y_0 are not constrained.

6.3 Parallel Computing under Python, Caching the Data, Optimization

In an alignment with a 10 Million Minimum Bias Magnet-on sample, there are at each iteration 22 Million individual track fits[7]. The main time is spent however in extracting the data from the track containers on the disk, only about one quarter of the total amount of time is spent on local fit of all tracks. The global fit or the alignment matrix inversion takes a negligible part of job's total time. With several iterations to perform the required amount of time expands to a unreasonable sum of more than 2 weeks.

Historically, at this point, confronted with large amount of data used in alignment and implicitly the large time intervals spent on one alignment, I took the decision to spend a few month optimizing and parallelizing the alignment algorithm.

Conceptually the alignment can be split in several sub-processes, with each fitting a different track sub-sample. A master process can collect the alignment matrices $\mathbf{C}'$ and the measurements vectors $\mathbf{b}'$ form each sub-process and sum them. The simple summation is sufficient to arrive at the alignment matrix of the entire sample.

$$\mathbf{C}' = \sum_i \mathbf{C}'_i, \qquad \mathbf{b}' = \sum_i \mathbf{b}'_i, \qquad \mathbf{a} = \left(\mathbf{C}'\right)^{-1} \mathbf{b} \tag{6.6}$$

Though the outlined parallelization algorithm is straightforward, its implementation within the LHCb software is not trivial (to say the least).

The alignment iterations are initialized and incremented by a python script that takes control over a large collection of Gaudi software algorithms, tools and services[8]. The Escher alignment package uses an implemented Gaudi Alignment Algorithm to communicate with several sets of Gaudi Tools, e. g. the tools that compute the measurements, configure the sub-detectors, or determine the derivatives. Each algorithm is executed once per each event.

The python alignment job script calls for each event the Alignment Gaudi-Algorithm to compute the measurements and to refit the tracks in the event containers. The obtained track parameters are stored in a cashed area, together with an updated $\mathbf{C}'$ matrix to which are added terms depending on global derivatives and local derivatives. Iterating the procedure over all data sample, when the last event is reached the Global Alignment takes over and finishes the construction of global alignment matrix $\mathbf{C}'$. The latter procedure subsequently inverts the alignment matrix and computes the alignment parameter vector $\mathbf{a}$. The script restarts a new alignment Newton-Raphson iteration after rewinding the data sample.

At this point I have modified the alignment python script, in such a way that the track parameter corrections and the alignment parameter corrections are consistently propagated between iterations. The measurements at each iteration are recomputed by subtracting the model function with the last iteration's alignment and track parameters. The derivatives, global or local, are computed by using the last iteration's values too. To avoid reading the same IT-cluster information at each iteration – in the same time asking the cached LHCb geometry database where is in space an IT-Cluster and which is its trajectory – I have cached all alignment relevant information on the IT-Clusters to a gzip-ed binary file. This made all the iterations after the starting iteration, 4 times faster.

To parallelize the job, a job pool is started. The pool uses multi-threading to spawn multiple jobs on multiple-cores machines. 8 jobs can be sent simultaneously to be executed, each job having its own set of dst (Data Summary Tape) files.

A master process monitors the status of each job in the pool, and after all have finished running, a new process is started , that sums all the alignment matrices and all $\mathbf{b}'$ vectors. The latter process inverts the alignment matrix equation and finds the alignment parameters or their corrections.

The Newton-Raphson method was implemented in this parallelized approach, but the newer quasi-

[7] I am using a momentum threshold of 20 GeV.

[8] On a basic level Gaudi is a customized set of predefined c++ class skeletons, and all the LHCb software algorithms or tools inherit form these archetypes

Newton step procedure was not yet ported from the old non-parallel alignment version.

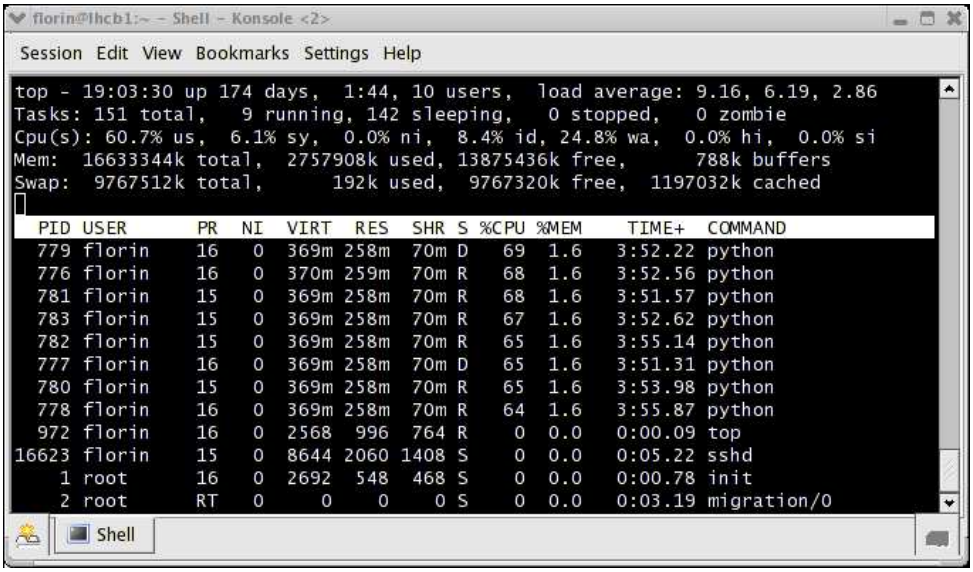


Figure 6.11: 8 alignment processes running simultaneously

6.3.1 Example of a Parallelized Job

In a typical example with 8 dst input files - each of 60k Minimum Bias 7+7 TeV events - was executed on a 8 core node from the MPI LHCb cluster. To compare, two jobs one with the old "Alignment" package and one with 8 parallel subprocesses were sent to separate nodes. The core usage and the allocated resources are displayed in the figure 6.11 for the 8-processes alignment. The two jobs were made to perform 9 Newton-Raphson iterations over the same data.

The completion time were:
1. for the old alignment 32h 55 min;
2. for the new alignment 4h 30 min:

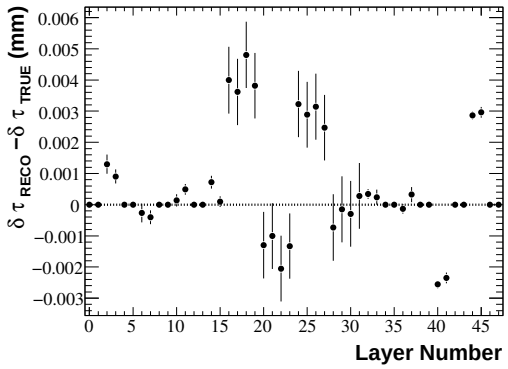
From this example the code was optimized further, and now a typical 10 Million Minimum Bias Magnet-on alignment takes between 1 day and 1 day and a half.

6.4 Monte Carlo Alignment Results for an Isotropic Charge Distribution in the Alignment Sample

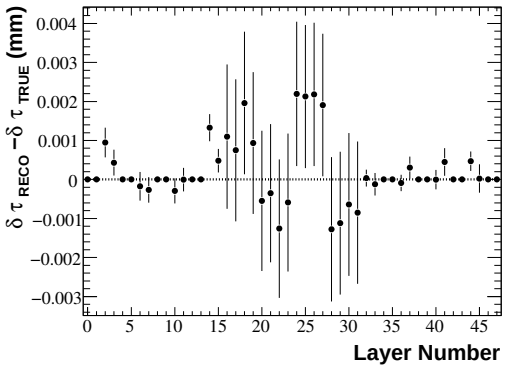
6.4.1 Layer Alignment, Two Degrees of Freedom $\delta\tau$ and $\delta\alpha$, Difference between Samples

To prove the utility of having an isotropic charge distribution sample, I made a double alignment, once with a full track sample and second time with a reduced sample obtained by selecting only those tracks in the sample that make the charge distribution in IT isotropic.

The alignment sample chosen for this run is composed of one million Minimum Bias simulated events from the standard LHCb mass-generated samples. The total sample has about 2 million tracks with momentum larger than 20 GeV. Of these tracks only about 1 Million tracks are included in a second alignment. The tracks chosen for the second alignment were selected with the requirement that in Left or the Right box there should be no positive charge excess or negative charge excess respectively. In this subsection I have not yet implemented the full unfolding of the charge anisotropy. In spite of this, it is clear when comparing the plots 6.12(a) and 6.12(b) that the alignment results improved despite the reduction of the tracks sample.



(a) $\delta\tau$ degrees of freedom, full sample results

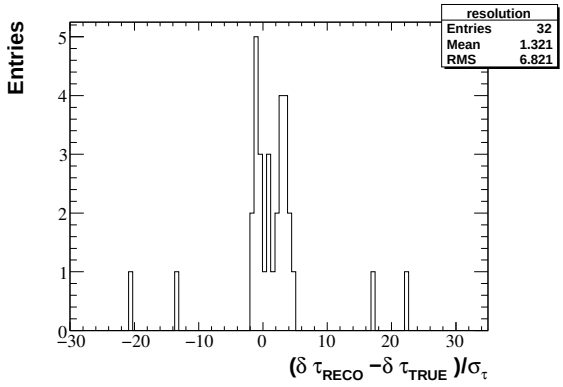


(b) $\delta\tau$ degrees of freedom, reduced sample

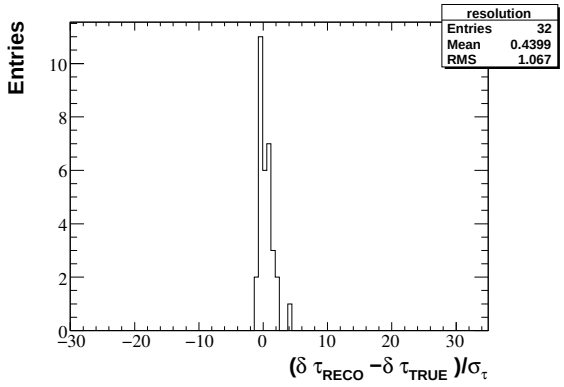
Figure 6.12: Alignment results: for full sample, and for the reduced sample with an isotropic charge distribution through IT. Alignment resolutions with Millepede errors estimates, $\delta\tau$ measurement shift. Alignment done at layer level.

Comparing Figures 6.12(a) and 6.12(b), it is obvious the disappearance in the latter figure of the large biases in the third station. The Left-Right Box layers of the third station (indices 40 to 48) have non-fixed 1st X and U layers, and the reconstructed $\delta\tau$ values for these layers are clearly much larger than expected in 6.12(a). The model biases pushed the mostly positive Left Box away from the mostly negative Right Box. As the model biases became negligible in 6.12(b), the split between Left and Right of the third Station became null.

The effect is better visible in the quasi-pulls of the alignment results 6.13(a) and 6.13(b). The quasi-pull for the latter is consistent with the Normal distribution, and the full sample histogram is dominated by the tails given by the Left/Right Box layers of the third IT station. The central distribution of the 6.13(b) histogram is also narrower than the central distribution of 6.13(a) histogram, fact seen in the resolution plot, too. The last effect is generated by the parabolic shearing weak mode in each quadrant in conjunction with the model biases. The weak mode is amplifying the effects of model biases, in the full samples results.



(a) $\delta\tau$ degrees of freedom, full sample



(b) $\delta\tau$ degrees of freedom, reduced sample

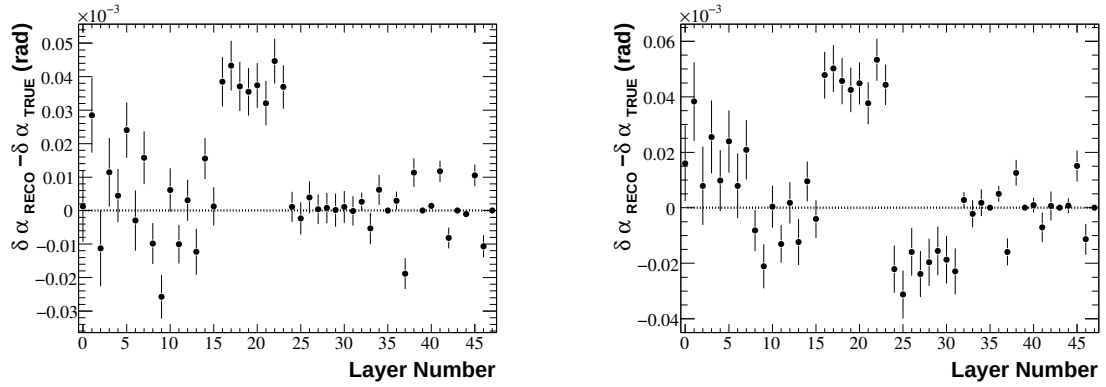
Figure 6.13: Alignment quasi-pulls: for full sample, and for the reduced sample with an isotropic charge distribution .

The parabolic shearing is seen easily in 6.12(a) in the values corresponding to the central indices.

The global mode is constrained by the overlap tracks, but the residual parabolic shearing weak mode in each quadrant is still persistent. In 6.12(b) the effect of the parabolic shearing is of the same magnitude as the Millepede error estimates, hence the effect is leveled-out.

The average error over all $\delta\tau$ layer alignment parameters is:
1) for the full sample with 1.75 million tracks in the final iteration, $2.0\mu\mathrm{m}$;
2) for the reduced sample with 0.85 million tracks in the final iteration, $1.0\ \mu\mathrm{m}$.

The results for the second type of alignment parameter $\delta\alpha$ are outlined by the figures 6.14(a) and 6.14(b). With small exceptions, the results are identical.



(a) $\delta\alpha$ degrees of freedom, full sample results

(b) $\delta\alpha$ degrees of freedom, reduced sample

Figure 6.14: Alignment results: for the full sample, and for the reduced sample with an isotropic charge distribution. Alignment resolutions with Millepede errors estimates, $\delta\alpha$ DoF. Alignment done at layer level.

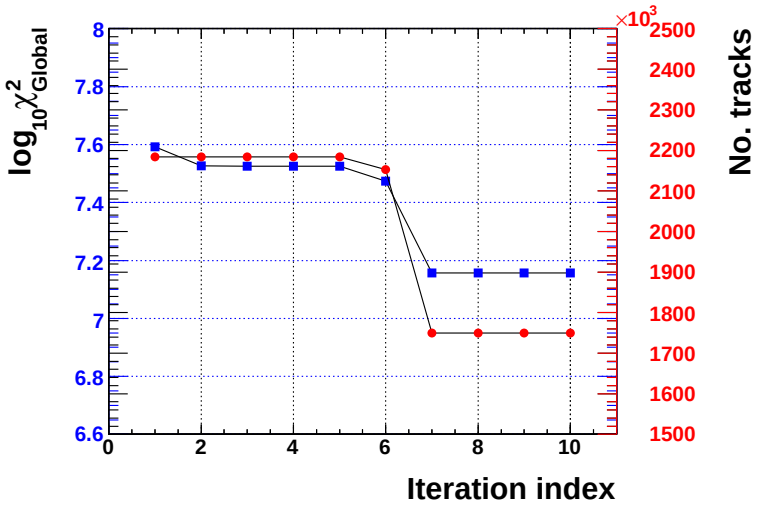
The average precision values for these degrees of freedom are:
1) for the full sample with 1.75 million tracks in the final iteration, 17 micro-radians;
2) for the reduced sample with 0.85 million tracks in the final iteration, 23 micro-radians.

The Millepede $\delta\alpha$ error propagation is underestimating the true error values by an average factor of 3. It is not surprising, since as I already explained I have constrained only one global rotation by fixing the last layer in each quadrant. There are in addition at least 2 $\delta\alpha$ weak modes per quadrant which perturb the error propagation procedure. The weak modes are the result of the correlation between track curvature parameter Q and the alignment parameters.

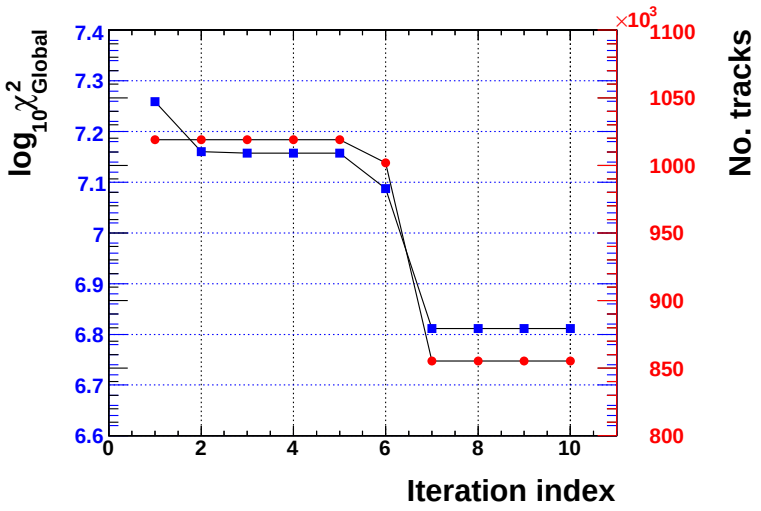
The Newton-Raphson iterations converge in one step, and after the track-χ^2 quality cut the procedure converges in 3 iteration. The convergency plateaus are displayed in 6.15(a) and 6.15(b) for the full-sample alignment, and for the alignment with a reduced sample respectively.

The reduced alignment χ^2 for the full sample is 1.17 with 12.25 million fit degrees of freedom. The reduced alignment χ^2 for the reduced sample is 1.08 with 5.99 million degrees of freedom. The difference with respect to one is due to: weak modes, model biases, some random scattering, and the potential $e^{\pm}$ bremsstrahlung.

No quasi-Newton steps were needed in the alignment of the most sensitive alignment parameters $\delta\tau$ and $\delta\alpha$.



(a) full sample's global χ^2



(b) global-χ^2 of the reduced sample

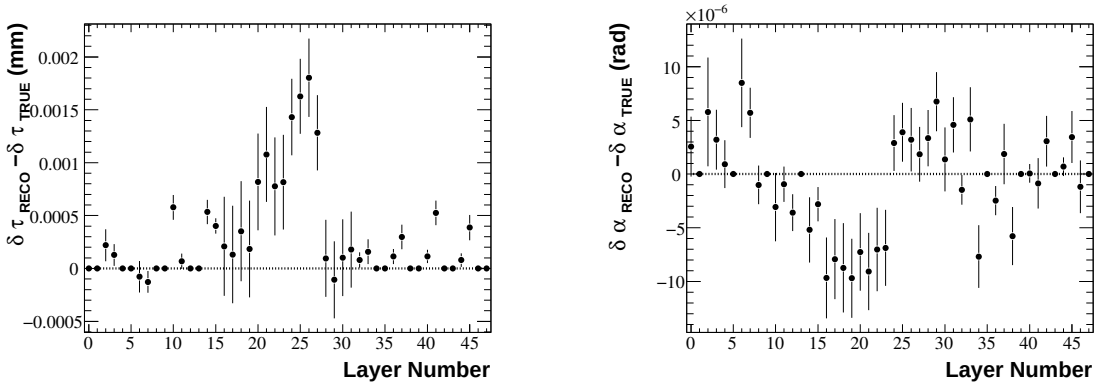
Figure 6.15: Alignment-χ^2 evolution for full and reduced samples, convergence and track selection at each iteration step

6.4.2 Layer Alignment, Two Degrees of Freedom, $\delta\tau$ and $\delta\alpha$, Isotropic Charge Distribution in IT

Using the parallelized version of the alignment software, I have aligned the IT at layer level using 10 million Minimum Bias events. The initial sample has about 22 Million tracks, and after removing the charge excess from each 2 cm length unit-cell in the phase-space of x_0 and y_0, I have about 4.7 million tracks left.

Since the parallelized version of the alignment software has less ROOT-storage capabilities, I have not provided the evolution plots of the alignment χ^2.

I have used here the usual constraints per IT quadrant: 2 X layers shifts, 1 U-layer shift, 1 V-layer shift, 1 U-layer angle, 1 X-layer angle. The extra 1 U-layer angle was used to constrain one weak mode in $\delta\alpha$. The global mode of parabolic shearing is partially fixed by the overlap-region tracks.



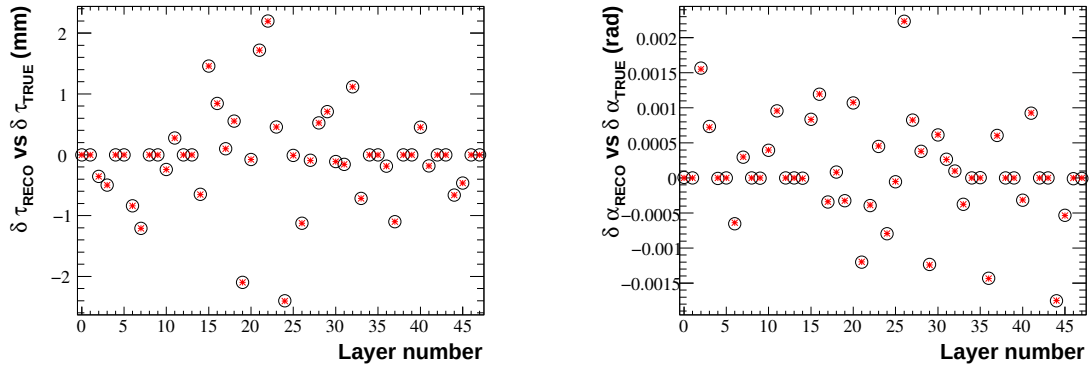
(a) $\delta\tau$ degrees of freedom

(b) $\delta\alpha$ degrees of freedom

Figure 6.16: Alignment resolutions with Millepede error estimates, alignment performed for 2 types of alignment parameters at layer level. Alignment used a charge symmetric distribution sample

Alignment resolutions are displayed in figures 6.16(a) and 6.16(b) for $\delta\tau$ and $\delta\alpha$ respectively. The average error over all layers is:
1) for $\delta\tau$, 0.50 μm;
2) for $\delta\alpha$, 5 micro-radians.

Most contributions to the $\delta\tau$ uncertainty are dominated by the parabolic shearing weak mode of the layers in the second station, i.e. see the central indices values in 6.16(a).



(a) reconstructed $\delta\tau$ degrees of freedom

(b) reconstructed $\delta\alpha$ degrees of freedom

Figure 6.17: Alignment results - black circles, and input misalignments - red asterisk, the alignment estimated errors too small to be seen. Alignment used a charge symmetric distribution sample

The error propagation slightly[9] underestimates the $\delta\tau$ error by a factor of 1.7, and there is a systematic shift in the reconstructed τ distributions due to the parabolic shearing in each quadrant. The error

[9]At 1 micrometer level and when starting with unrealistic measurement error estimates, any underestimation of alignment errors by only a factor lower than 2 is very small. Most alignment groups do not even bother with making error propagation, because of large initial biases in measurement residuals.

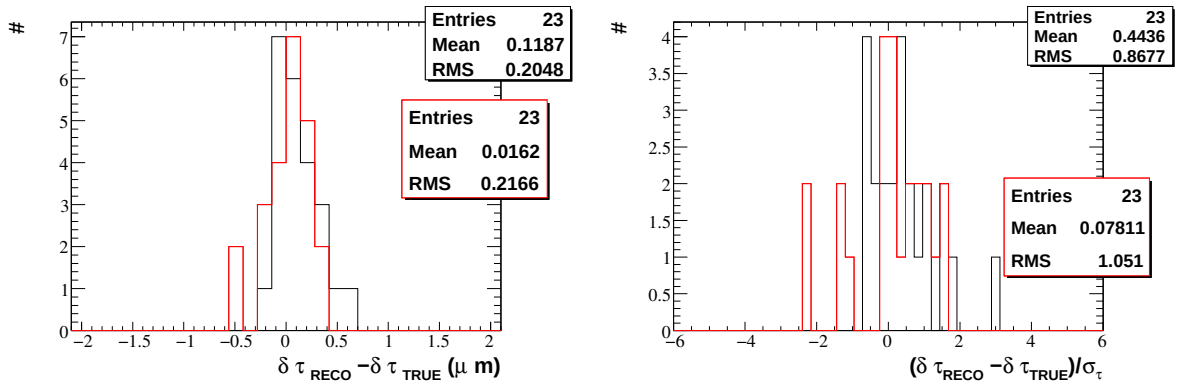
propagation of $\delta\alpha$ is underestimating the real error by a factor of only 1.6 and with no systematic shifts. The addition of an extra constraint in $\delta\alpha$ removed most systematic effects from the reconstructed $\delta\alpha$ z-rotations.

The convergence was reached with a standard Newton-Raphson method. The track-χ^2 cut was used but it had changed the alignment sample only by a fractional amount, as we would have expected the random scattering is not very important at 20 GeV.

The input misalignments versus reconstructed alignment parameters are shown in 6.17 plots.

6.4.3 Resolution and Pull Plots of the Alignment Results for 23 Independent Alignment Sample

23 alignment runs were performed on 23 independent and distinct alignment samples, each of 1 million LHCb-standard Minimum Bias simulated events, with a nominal non-null B-field. About 26 million Minimum Bias events in 4 TB reconstructed dst files were copied to MPI LHCb cluster (see [12]). Under the parallelized alignment software version, the processing of these data by the LHCb cluster took about 1-2 days. Separate misaligned LHCb Geometry Data Bases were used for each alignment run. An alignment run used on average between 0.4-0.5 Million tracks.



(a) $\delta\tau$ resolution plots

(b) $\delta\tau$ pull plots

Figure 6.18: $\delta\tau$ resolution and pull plots of 23 alignment runs for 2 DoF, 1st station Top-Box 2nd X layer, and 3rd station Top-Box 1st X layer

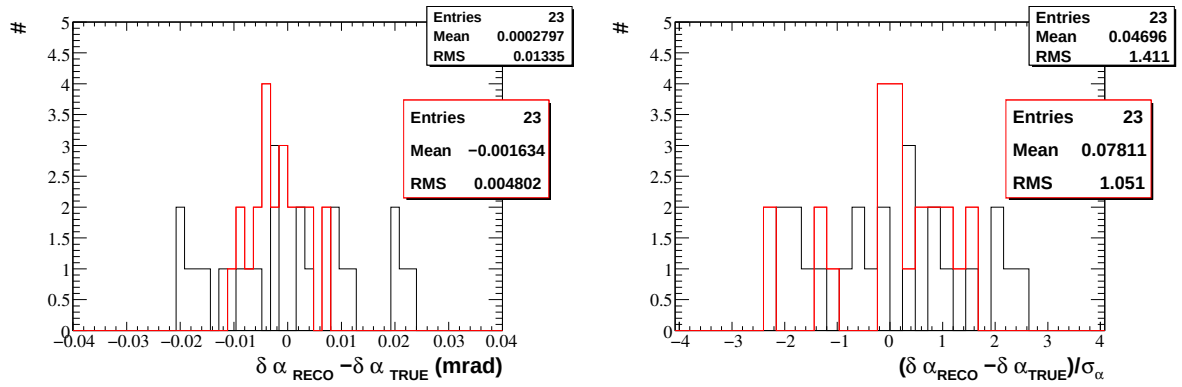
The alignment was done with the following constraints: for each IT quadrant I fix a priori the first two and the last two layers in $\delta\tau$, and for $\delta\alpha$ I fix the first U layer and the last X layer of each IT-quadrant.

The alignment results for two random layers, are shown in 6.18(a) and 6.19(a). The RMS of the observed distribution 6.18(a) is about 0.2 μm as expected for X layers. Pull plots of the alignment errors scaled by Millepede's alignment error estimates are displayed in 6.18(b), where the histograms have mean and RMS consistent with 0 and 1 respectively.

The RMS of $\delta\alpha$ distributions is some what different: 5 micro-radians and 13 micro-radians. The larger value corresponds to the X layer that is farther with respect to the a priori fixed 2nd X layer in the third station. The pull-plots for $\delta\alpha$ degrees of freedom are again within expected norms, see figure 6.19(b).

Convergence was achieved with the standard Newton-Raphson method. The track-χ^2 cut was used, but as before the tracks rejected were just a small fraction of the total.

For other layers systematic shifts were observed, but never larger than: 0.5 μm in $\delta\tau$, and 20 μrad in $\delta\alpha$.



(a) $\delta\alpha$ resolution plots

(b) $\delta\alpha$ pull plots

Figure 6.19: $\delta\alpha$ resolution and pull plots of 23 alignment runs for 2 DoF, 1st station Top-Box 2nd X layer, and 3rd station Top-Box 1st X layer

6.4.4 Minimal Number and Type of Constraints, Alignment Matrix, Weak-Modes

In this subsection I use again about 10 million Minimum Bias events, with about 22 Million tracks at larger than 20 GeV momentum. The final Newton-Raphson iteration uses about 4 million tracks.

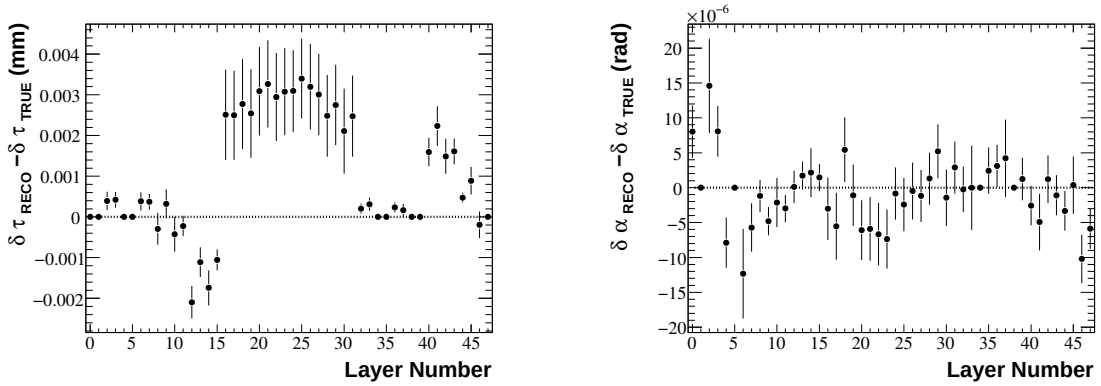
I use here the same type of algebra tools as the one used in section 5.4.

Minimally Constrained IT at Layer Level

After an extensive search for that configuration with the least number of constraints, I found that the following sequence of a priori fixed alignment parameters is equivalent to the minimum required set of constraints for IT at layer level:
1) In the IT-quadrant of the Top-Boxes I fix the first two and the last two layers in $\delta\tau$;
2) In the same quadrant I fix in $\delta\alpha$ the second layer which is an U layer, and the last V layer, too;
3) In the Bottom-Box I fix the same sequence of layers as for Top-Boxes in both $\delta\tau$ and $\delta\alpha$;
4) In the Right-Box of the third station I fix the last layer X in $\delta\tau$.
In total there are 13 a priori fixed alignment parameters.

The upper result was obtain after a step by step approach, with each step consisting in checking for the least constrained eigen-modes of the Alignment matrix followed by a full alignment to check the stability of solution in the Newton-Raphson iterations. Between 12 and 18 constraints, the step by step procedure avoided over-constraining the system by rejecting eigen-modes that are result of a mixture between a true geometrical weak-mode and a solution eigen-mode.

Figures 6.20(a) and 6.20(b) show the alignment results when 13 constraints are applied. The mean $\delta\tau$-precision over all non-fixed layers is 1.5 micrometers, and the $\delta\alpha$ mean is 5 micro-radians. With the simultaneous a priori fixing of an U and a V layer in $\delta\alpha$, there are no more weak modes apparent in the alignment results for this degree of freedom. The dominant contribution to the uncertainty of $\delta\tau$ parameters comes form a parabolic-shearing-like weak mode. As seen in the 6.20(a) graph, the central indices part which corresponds to the second station layers is coherently shifted with respect to the first and the third station layers.



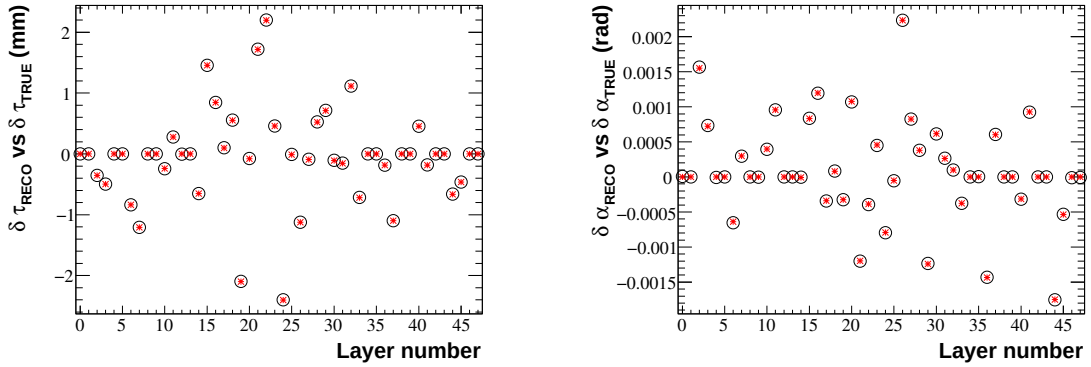
(a) $\delta\tau$ degrees of freedom

(b) $\delta\alpha$ degrees of freedom

Figure 6.20: Alignment resolution plots for 2 degrees of freedom. Alignment used a charge symmetric distribution sample, and 13 constraints.

The input misalignment are displayed in the plots 6.21(a) and 6.21(b), with the black circles being the reconstructed values. At this scale the Millepede error estimates just disappear under the size of the circles (even if the circles are 3 times smaller the error are still under the disk surface). Relatively large values of misalignments were considered, and the station-box-layer hierarchy was not explicitly considered in the misalignment generation.

The alignment results obtained with much smaller or null misalignments, were found to be equivalent to the ones just given.



(a) reconstructed $\delta\tau$ degrees of freedom

(b) reconstructed $\delta\alpha$ degrees of freedom

Figure 6.21: Alignment results - black circles, and input misalignments - red asterisk. Alignment used a charge symmetric distribution sample.

The alignment was done via the parallel version of the alignment algorithm. The algorithm in this case does not store in a ROOT format the relevant information for each Newton-Raphson step.

The convergency was reached after 11 iterations, and there was no significant decrease in the number of tracks when the track-χ^2 cut was applied. The standard Newton-Raphson procedure was applied.

For fewer constrains applied, the Newton-Raphson procedure is no longer convergent and the quasi-Newton step results are dominated by the weak-modes.

Residual Weak Mode in the Minimally Constrained IT at Layer Level

To continue the alignment discussion of the last paragraphs, I have retrieved the alignment matrix $\mathbf{C}'$ from the last example and diagonalized it. The eigenvalue spectrum and the least constrained eigenvectors were obtained. The matrix is given in the appendix, figure 8.23, in a normalized version. As I have explained in section 5.4, I have used only the simple 60 mm lever arm rule to normalize the matrix. The results with the other two normalization schemes were equivalent with the next results.

The spectrum of the alignment matrix is given in figure 6.22. As seen in the plot, there is one clear residual weak mode at the lower edge of the spectrum.

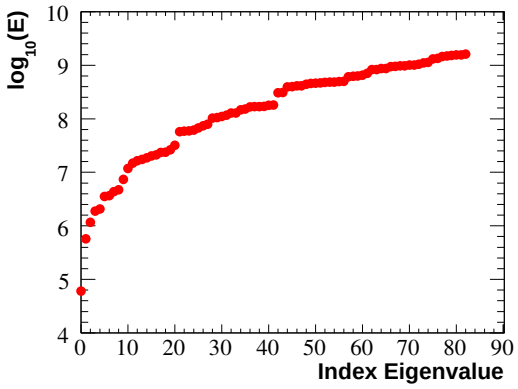


Figure 6.22: Normalized alignment matrix spectrum for a minimally constrained IT-layers system.

The eigenvalue decomposition of $\mathbf{C}'$ matrix provides the eigenmode corresponding to the lowest eigenvalue. The representation of this eigenvector is shown in 6.23, where the second station layers are clearly dominant, and all $\delta\tau$ parameters of these layers are coherently shifted. The eigenvector has a clear match in the alignment resolution plot 6.20(a).

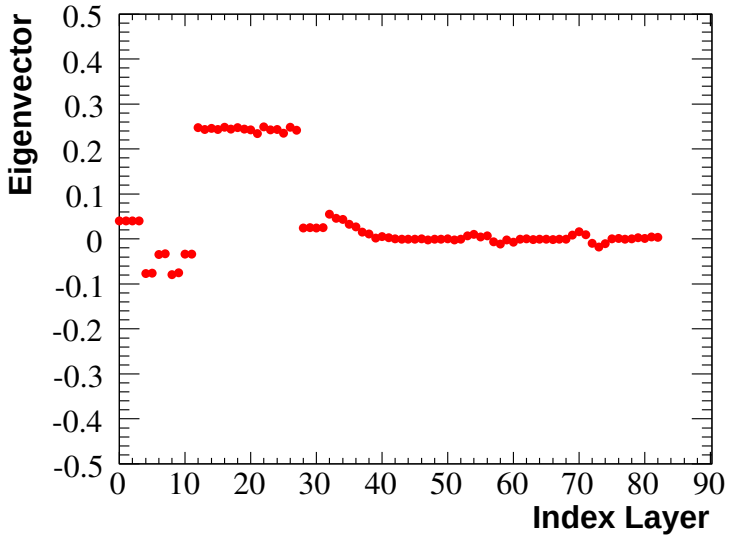


Figure 6.23: Least constrained eigenmode

Whether the last weak mode should be constrained or not, remains at this level an open question. With alignment resolution between layers of the same station much smaller than 1 micron, and the alignment resolution between IT-stations within 2-3 microns, there may not be any need to additionally constrain the residual weak-mode.

After constraining 14 weak modes and global modes, I obtained a close to perfect error propagation from Millepede.

6.4.5 Ladder Alignment Results

For ladders the main problem is similar to the one when the magnetic field was null. If only tracks originating in the interaction point are included in the alignment sample, there are then 28 quasi-independent ladder stacks. The tracks originating from secondary vertices or which have large curvatures do not fulfill the minimum quality selection rules of alignment. In absence of the latter tracks, the remaining tracks are connecting only very weakly adjacent stacks of ladders. The tracks passing through the overlap regions between the neighboring ladders of the same IT-layer, are in short supply again. The overlap-tracks are about 1 % from total. With this small fraction split over 168 overlap regions, and if we require to have about 1000 tracks through each, we esentialy need to have a staggering large alignment sample at our disposal. The situation is made even more problematic for the most outer-lying ladders in the Left and Right IT-Boxes. These ladders have the smallest average occupancy. The occupancy as function of distance to the beam is almost an inverse square power law.

In my approach the problem is made worse by the requirement to have isotropic track distribution.

There is already a proposal from the alignment group, to have a dedicated set of overlap tracks stored on-line.

For now I have considered the 28 ladders stacks independent. Even in this apparently simple system, there are major complication from weak modes due to the extra track parameter and its correlation to the alignment parameters. As it was already shown at the level of layers, these correlation generate weak modes and even globals.

From full alignment runs and matrix diagonalizations I noticed that for ladders it is imperative to have the parabolic-shearings a priori constrained. It means that in addition of the 4 a priori constraints in $\delta\tau$ there is needed a fifth constraint per each ladder stack. Before, for layers the parabolic shearing was acceptably constrained by the overlap tracks connecting boxes of same stations, it seems that for ladders there is not such luck as the overlap tracks connecting ladders of the same layer are not enough in number and quality[10].

In conclusion for each stack of ladders, I have a priori fixed the first 2 ladders, the last two ladder and a extra ladder in the second station.

For $\delta\alpha$ the situation is even more complicated, by the fact that there are no straightforward geometrical equivalent for the observed weak modes. Initially I tried an alignment with the first U ladder and the last X ladders constrained. The results could be seen in 6.24 for both $\delta\tau$ and $\delta\alpha$ alignment parameters. The effect of the weak mode is immediately visible. The error propagation in $\delta\alpha$ plot is grossly underestimating the actual errors, and in addition there are clear patterns emerging in the third station at large index value.

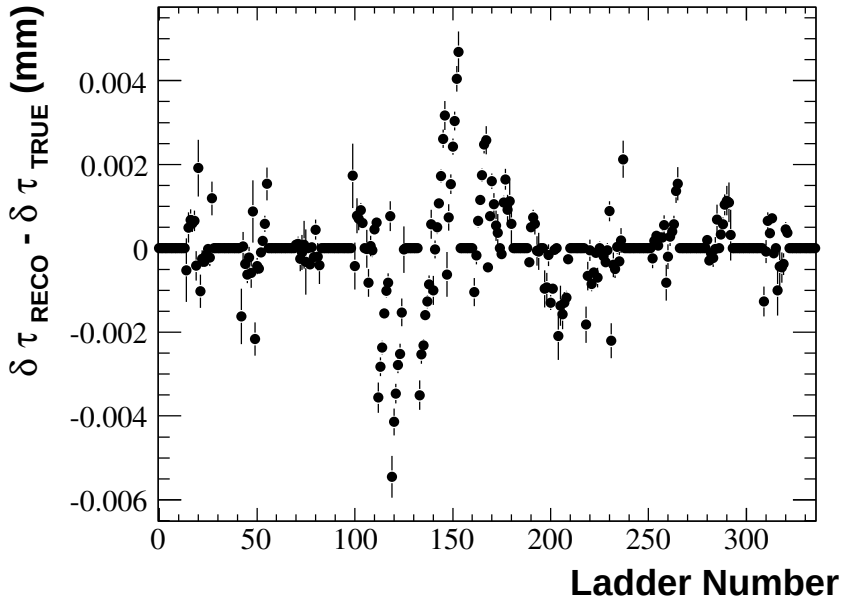
The results in 6.24 have for $\delta\tau$ an average of 1.3 micrometers, and in $\delta\alpha$ an average of 130 micro-radians. For a ladder of 20 cm length, it means the positions of the edges fluctuate with about 13 micrometers.

To constrain the observed weak-modes I applied in $\delta\alpha$ the same set of constraints as for $\delta\tau$. The resolution plot of all 336 ladders is given in 6.25.

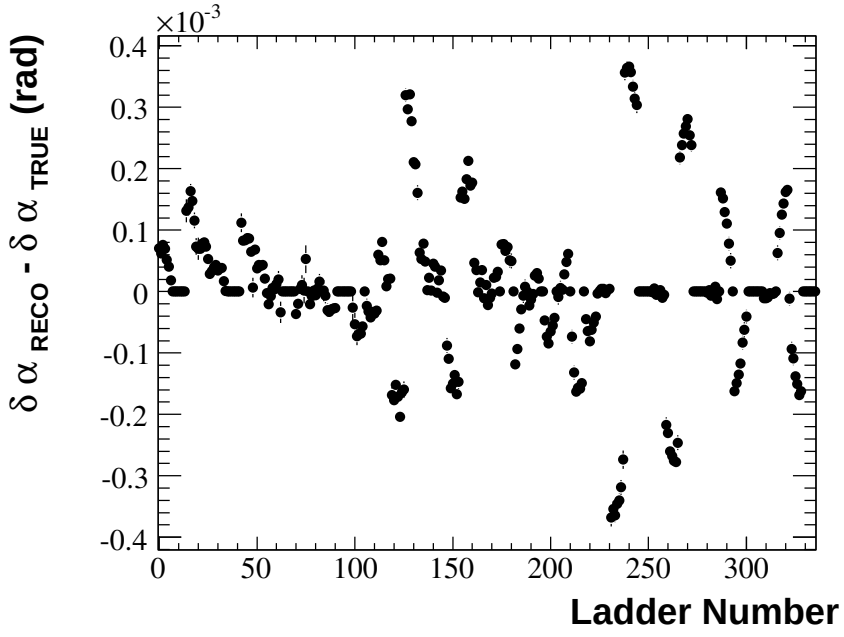
In this case the average error over 182 non-fixed $\delta\tau$ parameters is 0.8 micrometers, and the $\delta\alpha$ average over 182 non-fixed parameters is 11 micro-radians. The edge position uncertainty is now of about 1.1 micrometers for long ladders. The error propagation in this case is 10 times more exact for $\delta\alpha$ reconstructed parameters. Even though the error are underestimated – in $\delta\tau$ at a level of a multiplicative factor of 3, and in $\delta\alpha$ at level of a multiplicative factor of 1.8 – the effect is probably due exclusively to the random scattering effects and model biases. Here, there is no weak mode evident.

[10]Quality means in this context that the track parameters span a large part of the phase-space, whereas for ladder-overlap tracks the track parameters are essentially fixed, e. g. especially in x direction the t_x slope is fixed

It might be some in-between setup that has the minimal set of constraints. Due to time limitation this is still an open questions. In absence of a selection tool which allows to get a estimate on the coupling strengths between the neighboring ladders stacks, I had to deduce the effect of the overlap tracks directly from the alignment matrix. The typical non-normalized alignment matrix is given in appendix figure 8.24. The sparse nature of the alignment matrix is a direct consequence of the lack of connecting tracks between ladders of the same layer.

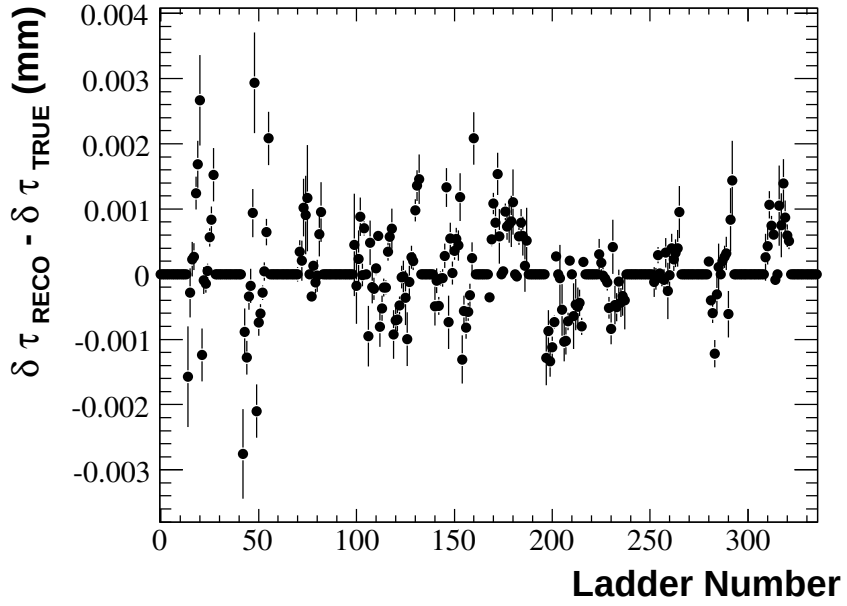


(a) $\delta\tau$ resolution plot

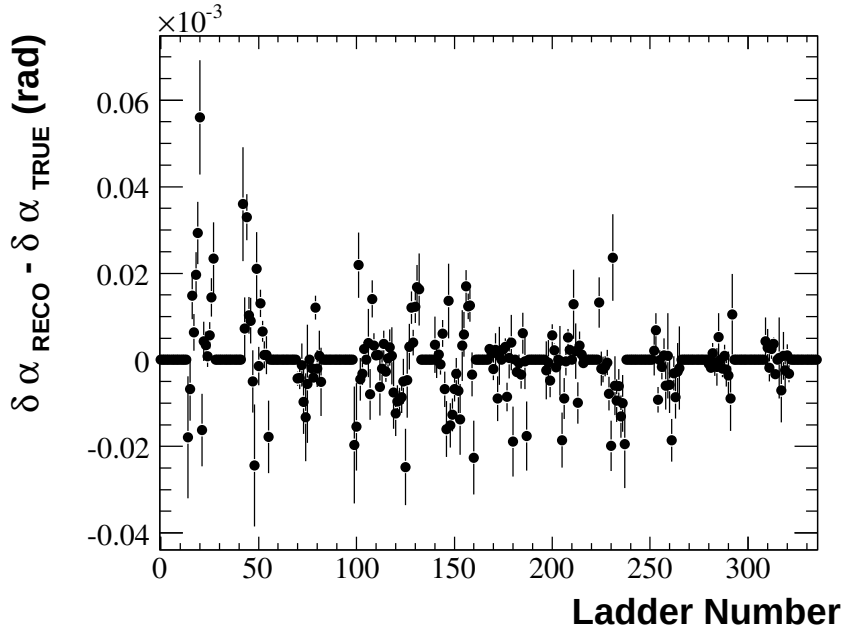


(b) $\delta\alpha$ resolution plot

Figure 6.24: Alignment of ladders with the same set of constraints as for layers



(a) $\delta\tau$ resolution plot



(b) $\delta\alpha$ resolution plot

Figure 6.25: Alignment of ladders with more constraints than the usual set of constraints for layers

Chapter 7

Summary and Conclusions

Three successful alignment models were implemented in the frame of the LHCb Monte Carlo and Reconstruction. Of these 3, two were described in this report together with the path taken in their derivation. As the models are fraction of polynomials, and hence non-linear functions of alignment and track parameters, a Newton-Raphson scheme was coded, with implicit linearization of the model function. The new linear problem is solved by Millepede tool and the results used in subsequent iterations. Sometimes when Newton-Raphson method leads to divergent results, a more robust approach is used by adding quasi-Newton steps. The latter are required to minimize the global χ^2.

On the other side, an independent tracking procedure was implemented, together with the measurement calculus. To extract a value corresponding to a track hit I used the position and trajectory direction of an IT-cluster. The measurement provided by this algorithm has already encrypted the misalignment information. With the alignment algorithm as the decrypting tool of the measurement residual distributions, alignment parameters and track parameters are obtained.

The misalignments were generated according to some properly scaled gaussians, and the generated number inputed in the LHCb Monte Carlo Geometry Databases. In general the LHCb's Pattern-Recognition algorithms, running in the Reconstruction phase were using the same Geometry Database as the Simulation Database. Thus, the hit selection for a track does not depend on the misalignments. The few Reconstructions performed using a misaligned geometry, showed problems arising only for very large misalignment values when the track finding efficiency decreases severely. As this happens for values much larger than the IT sensor pitch of about 200 micrometers, the problem is more in the domain of pre-alignment. The pre-alignment method being a simple and very robust alignment procedure that uses directly the means of measurement residual distributions to compute the alignment parameters. Since the purpose of the present research was to obtain alignment precisions of about 1 micrometer, I have not explicitly discuss the results related with alignment dependence on track finding efficiency of the Pattern-Recognition, this not being the problem which I wanted to solve.

With a multitude of misaligned geometry databases, alignment runs were performed on several samples of distinct event-types. Simultaneously the samples were partitioned in independent sub-samples for separate alignments. Probably the alignment code – early released versions as well as later versions – was used intensively in hundreds alignment runs. Depending on the number of tracks in the sample and on the set of a priori constraints, the precision in the reconstructed alignment parameters fluctuates from one case to the other.

In case there is a magnetic field, the charge anisotropy through IT is usually removed during the later Newton-Raphson iterations. This allows consistent error propagations, without model biases.

Layer Alignment

Even for a track sample which contains lots of low momentum particles, a quality cut on the track-χ^2 allows to compute alignment parameters and alignment error with high precision. In the absence of the magnetic field the typical alignment precision for a shift in the direction of measurement is of 1 μm and

lower, when about 40000 tracks are used in the final Newton-Raphson iteration. For the same type of events, $\delta\alpha$ z-rotation is found with a precision 20 micro-radians or better, depending on the number of tracks used. The error propagation worked up to a multiplicative factor of 1.3 to 1.7, depending on how many tracks were used.

For an alignment using a Minimum Bias sample with nominal non-null field and nominal luminosity, the alignment parameters $\delta\tau$ were found within 0.2 microns, when about 0.4 Million tracks were used in the final iteration. The $\delta\alpha$ degrees of freedom were reconstructed with a precision of 5 to 20 micro-radians. The error propagation is relevant up to a multiplicative factor close to 1.8. The last number is lower if "only" half a million tracks are used.

3 other degrees of freedom $\delta\beta$, $\delta\gamma$ and δz are obtainable with good precision relative to the error estimates, if very high energy tracks are used. The 5 degrees of freedom alignment needs quasi-Newton steps to prevent divergence and to limit large non-linear fluctuations. The measurement residuals are largely insensitive to δz and $\delta\gamma$ fluctuations, and as a result these alignment parameters are reconstructed with bad precision. Yet, this is not worrisome, as the track quality is not changed even for large deviations in these alignment parameters.

Ladder Alignment

For ladders the alignment precision decreases, but as there are 7 times less hits in a ladder, this is hardly surprising. Without Field, and in the case of a 100 GeV track sample, the alignment precision is about 1 micrometer for $\delta\tau$, and of 70 micro-radians for $\delta\alpha$. The error propagation is underestimating the real error, but as the biases coming form random scattering effects are much more likely to perturb the position of ladder than the one of a layer this is not unexpected.

When the field is non-null, and the beam is nominal, the ladder alignment is precise within micrometer level for $\delta\tau$. The $\delta\alpha$ alignment parameters are reconstructed correctly with a precision of about 11 micro-radians to 130 micro-radians, depending whether all the weak modes are constrained or not.

Minimally constrained setup, Layer level

For an alignment in absence of field a minimal set of 9 constraints allows an alignment with good precision and consistent error propagation. In magnetic field 13 constraints are required to obtain the same degree of fidelity in the alignment results. Bellow these values, the standard Newton-Raphson procedure is divergent or leading to results which are dominated by the weak-mode contribution. The quasi-Newton step is not much more helpful since it provides results dominated by the weak mode contribution too. When including $\delta\beta$ the situation is not changed, except that I had to fix one more alignment parameter to fix the global y rotation. As for the other degrees of freedom $\delta\gamma$ and δz, these act as weak modes themselves, and there is no clear way to select one minimal set of constraints for this case.

The a priori fixed alignment parameters must be found by a global alignment of the Tracking stations, where the position of IT is computed with respect to the VELO and TT. Our alignment algorithm could do this provided we make available the extrapolated track states computed using the hits in the VELO and TT.

For Ladder level and when the magnet-off samples are used, the tracks through the overlap regions are too few and of too low quality to allow a consistent eigen-mode decomposition. Especially for the most outer-lying ladders the occupancy is simply to low, and a proper study of the ladder weak-modes needs large amounts of overlap tracks with more uniformly distributed tracks sample. Even the newer samples, with 1 Million Minimum Bias events, have insufficient number of quality tracks - probably due to Open-VELO setting used. For the magnet-on case there was too little time available to do a systematic study of the weak modes at ladder level.

Open tasks

There are few tasks which could have been performed if time allowed.
1. I would have liked to try making a global alignment of IT using the VELO-TT extrapolated track states.
2. In magnet-off case, the HCAL and ECAL clusters can supply a momentum estimate and a particle ID tag, rejecting electrons and positrons. Selecting by this method a sample with very high energy pions, we could perform an alignment of all 5 degrees of freedom of a layer with high precision.
3. Finding the weak modes at ladder level in the presence or the absence of field is still needed. The same is true for the weak modes of $\delta\gamma$ and δz especially in the non-null field case.

Chapter 8

Appendix

8.1 Toy Monte Carlo Results

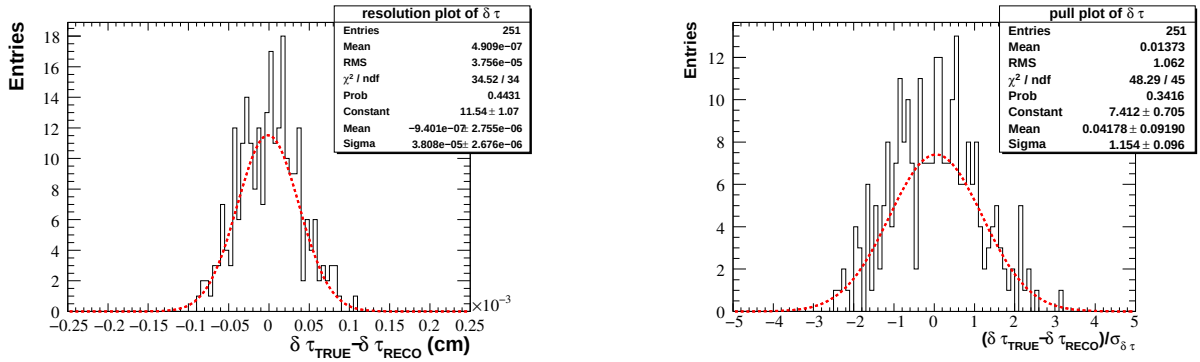


Figure 8.1: IT layers, Toy MC results, resolution and pull-plots of $\delta\tau$ DoF for 251 independent alignment runs with distinct track samples

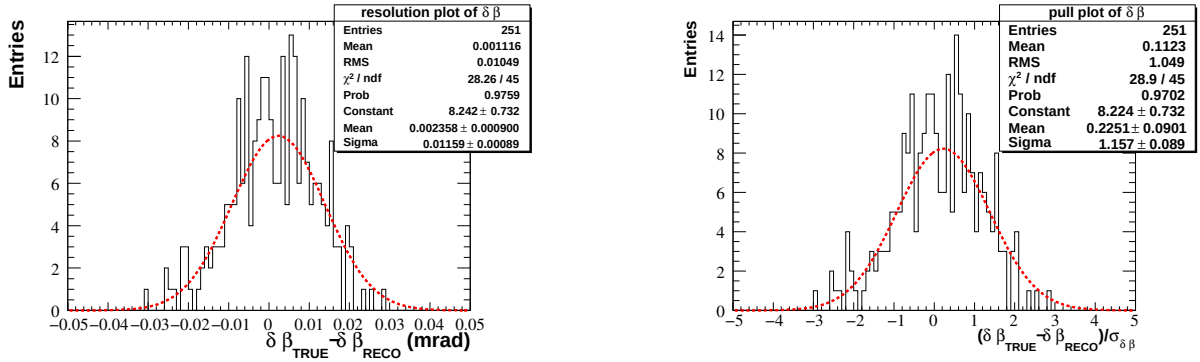


Figure 8.2: IT layers, Toy MC results, resolution and pull-plots of $\delta\beta$ DoF for 251 independent alignment runs with distinct track samples

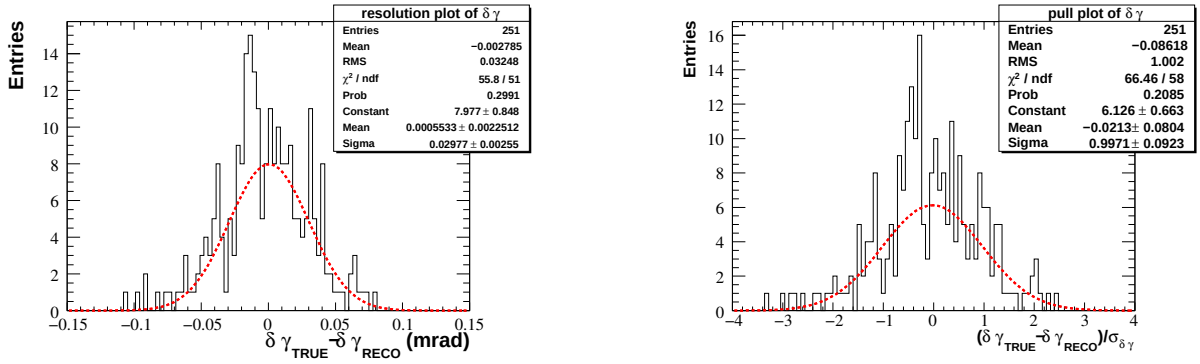


Figure 8.3: IT layers, Toy MC results, resolution and pull-plots of $\delta\gamma$ DoF for 251 independent alignment runs with distinct track samples

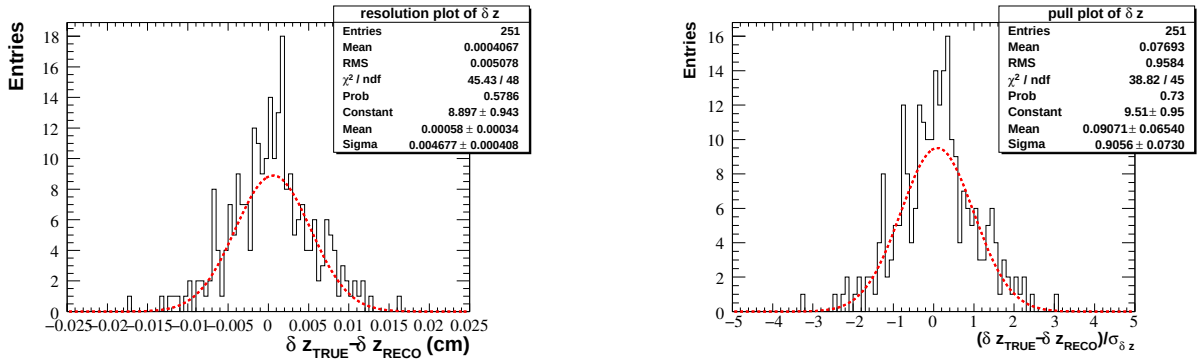


Figure 8.4: IT layers, Toy MC results, resolution and pull-plots of δz DoF for 251 independent alignment runs with distinct track samples

8.2 Alignment Model for a non-Null Field

8.2.1 Model Term Dependent on the Rotation around LHCb Y-axis, $\delta\beta$

$$U'''_{model}(\delta\beta) = \delta\beta\times\Bigg(-\Bigg(\Bigg(-\tau_z - \frac{\xi_x t_y((\tau_z\xi_y - \tau_y\xi_z)t_x + \xi_x(\tau_y - \tau_z t_y))}{\xi_y^2} +$$
$$\frac{(\xi_y + \xi_z t_y)((\tau_y\xi_x - \tau_x\xi_y)t_x + \tau_x\xi_x t_y + \xi_z(\tau_y - \tau_z t_y))}{\xi_y^2} + \frac{1}{\xi_y^3}2(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 2\xi_z t_y)$$
$$(-\xi_y X_c - \xi_x y_0 + \xi_x(Y_c + Y_p)) - \frac{4\xi_x(\tau_z\xi_y - \tau_y\xi_z)Qt_y(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))}{\xi_y^3} +$$
$$\frac{1}{\xi_y^3}2(\tau_y\xi_x - \tau_x\xi_y)Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) +$$
$$\frac{1}{\xi_y^4}6d(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 3\xi_z t_y)(-\xi_y X_c - \xi_x y_0 + \xi_x(Y_c + Y_p))$$
$$(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) -$$
$$\frac{9d\xi_x(\tau_z\xi_y - \tau_y\xi_z)Qt_y(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2}{\xi_y^4} +$$
$$\frac{1}{\xi_y^4}3d(\tau_y\xi_x - \tau_x\xi_y)Q(\xi_y + 3\xi_z t_y)\left(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)\right)^2\Big)$$
$$\Bigg(-x_0 + X_c + X_p - \frac{Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2}{\xi_y^3} -$$
$$\frac{dQ(\xi_y + 3\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^3}{\xi_y^4} - \frac{1}{\xi_y^2}(\xi_y + \xi_z t_y)$$
$$(t_x(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) +$$
$$\xi_x(-y_0 + Y_c + Y_p + t_y z_0 - t_y(Z_c + Z_p))))\Big/\Bigg(-\tau_x + \frac{(\xi_y + \xi_z t_y)((\tau_z\xi_y - \tau_y\xi_z)t_x + \xi_x(\tau_y - \tau_z t_y))}{\xi_y^2} +$$
$$\frac{1}{\xi_y^3}2(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) +$$
$$\frac{1}{\xi_y^4}3d(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 3\xi_z t_y)\left(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)\right)^2\Bigg)^2 +$$
$$\Bigg(Z_c + Z_p - \frac{1}{\xi_y^3}2Q(\xi_y + 2\xi_z t_y)(-\xi_y X_c - \xi_x y_0 + \xi_x(Y_c + Y_p))\left(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)\right) +$$
$$\frac{2\xi_x Qt_y(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2}{\xi_y^3} - \frac{1}{\xi_y^4}3dQ(\xi_y + 3\xi_z t_y)(-\xi_y X_c - \xi_x y_0 + \xi_x(Y_c + Y_p))$$
$$(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2 + \frac{3d\xi_x Qt_y(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^3}{\xi_y^4} +$$
$$\frac{1}{\xi_y^2}\xi_x t_y(t_x(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) + \xi_x(-y_0 + Y_c + Y_p + t_y z_0 - t_y(Z_c + Z_p))) -$$
$$\frac{1}{\xi_y^2}(\xi_y + \xi_z t_y)(\xi_x t_y X_c + t_x(-\xi_y X_c - \xi_x y_0 + \xi_x(Y_c + Y_p)) +$$
$$\xi_z(-y_0 + Y_c + Y_p + t_y z_0 - t_y(Z_c + Z_p))) - Z_p)\Big/\Bigg(-\tau_x + \frac{(\xi_y + \xi_z t_y)((\tau_z\xi_y - \tau_y\xi_z)t_x + \xi_x(\tau_y - \tau_z t_y))}{\xi_y^2} +$$
$$\frac{1}{\xi_y^3}2(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) +$$
$$\frac{1}{\xi_y^4}3d(\tau_z\xi_y - \tau_y\xi_z)Q(\xi_y + 3\xi_z t_y)\left(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)\right)^2\Big)\Big)$$

8.2.2 Model Term Dependent on the Rotation around LHCb X-axis, $\delta\gamma$

$$
\begin{aligned}
&U'^{v}_{model}\left(\delta\gamma\right) = \delta\gamma\times\Big(-\Big(\Big(-x_0+X_c+X_p-\tfrac{Q(\xi_y+2\xi_z t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))^2}{\xi_y^3}-\\
&\tfrac{dQ(\xi_y+3\xi_z t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))^3}{\xi_y^4}-\\
&\tfrac{1}{\xi_y^2}(\xi_y+\xi_z t_y)(t_x(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))+\\
&\xi_x(-y_0+Y_c+Y_p+t_y z_0-t_y(Z_c+Z_p))))\\
&\Big(\tfrac{\xi_x(-\tau_z-\tau_y t_y)(\xi_y+\xi_z t_y)}{\xi_y^2}+\tfrac{(-\xi_z+\xi_y t_y)((\tau_z\xi_y-\tau_y\xi_z)t_x+\xi_x(\tau_y-\tau_z t_y))}{\xi_y^2}+\\
&\tfrac{2\xi_z(\xi_y+\xi_z t_y)((\tau_z\xi_y-\tau_y\xi_z)t_x+\xi_x(\tau_y-\tau_z t_y))}{\xi_y^3}+\\
&\tfrac{1}{\xi_y^3}2(\tau_z\xi_y-\tau_y\xi_z)Q(-\xi_z+2\xi_y t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))+\\
&\tfrac{1}{\xi_y^4}6\xi_z(\tau_z\xi_y-\tau_y\xi_z)Q(\xi_y+2\xi_z t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))+\\
&\tfrac{1}{\xi_y^4}3d(\tau_z\xi_y-\tau_y\xi_z)Q(-\xi_z+3\xi_y t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))^2+\\
&\tfrac{1}{\xi_y^5}12d\xi_z(\tau_z\xi_y-\tau_y\xi_z)Q(\xi_y+3\xi_z t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))^2+\\
&\tfrac{1}{\xi_y^3}2(\tau_z\xi_y-\tau_y\xi_z)Q(\xi_y+2\xi_z t_y)\\
&(\xi_y y_0-\xi_y(Y_c+Y_p)+\xi_y Y_c+\xi_z z_0-\xi_z(Z_c+Z_p)+\xi_z Z_c)+\\
&\tfrac{1}{\xi_y^4}6d(\tau_z\xi_y-\tau_y\xi_z)Q(\xi_y+3\xi_z t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))\\
&(\xi_y y_0-\xi_y(Y_c+Y_p)+\xi_y Y_c+\xi_z z_0-\xi_z(Z_c+Z_p)+\xi_z Z_c)))/\\
&\Big(-\tau_x+\tfrac{(\xi_y+\xi_z t_y)((\tau_z\xi_y-\tau_y\xi_z)t_x+\xi_x(\tau_y-\tau_z t_y))}{\xi_y^2}+\\
&\tfrac{1}{\xi_y^3}2(\tau_z\xi_y-\tau_y\xi_z)Q(\xi_y+2\xi_z t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))+\\
&\tfrac{1}{\xi_y^4}3d(\tau_z\xi_y-\tau_y\xi_z)Q(\xi_y+3\xi_z t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))^2\Big)^2+\\
&\Big(-\tfrac{Q(-\xi_z+2\xi_y t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))^2}{\xi_y^3}-\\
&\tfrac{3\xi_z Q(\xi_y+2\xi_z t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))^2}{\xi_y^4}-\\
&\tfrac{dQ(-\xi_z+3\xi_y t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))^3}{\xi_y^4}-\\
&\tfrac{4d\xi_z Q(\xi_y+3\xi_z t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))^3}{\xi_y^5}-\\
&\tfrac{1}{\xi_y^2}(-\xi_z+\xi_y t_y)\\
&(t_x(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))+\xi_x(-y_0+Y_c+Y_p+t_y z_0-t_y(Z_c+Z_p)))-\\
&\tfrac{1}{\xi_y^3}2\xi_z(\xi_y+\xi_z t_y)\\
&(t_x(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))+\xi_x(-y_0+Y_c+Y_p+t_y z_0-t_y(Z_c+Z_p)))-\\
&\tfrac{1}{\xi_y^3}2Q(\xi_y+2\xi_z t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))\\
&(\xi_y y_0-\xi_y(Y_c+Y_p)+\xi_y Y_c+\xi_z z_0-\xi_z(Z_c+Z_p)+\xi_z Z_c)-\\
&\tfrac{1}{\xi_y^4}3dQ(\xi_y+3\xi_z t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))^2\\
&(\xi_y y_0-\xi_y(Y_c+Y_p)+\xi_y Y_c+\xi_z z_0-\xi_z(Z_c+Z_p)+\xi_z Z_c)-\\
&\tfrac{1}{\xi_y^2}(\xi_y+\xi_z t_y)\\
&(\xi_x(-t_y Y_c-Z_c)+\\
&t_x(\xi_y y_0-\xi_y(Y_c+Y_p)+\xi_y Y_c+\xi_z z_0-\xi_z(Z_c+Z_p)+\xi_z Z_c)))/\\
&\Big(-\tau_x+\tfrac{(\xi_y+\xi_z t_y)((\tau_z\xi_y-\tau_y\xi_z)t_x+\xi_x(\tau_y-\tau_z t_y))}{\xi_y^2}+\\
&\tfrac{2(\tau_z\xi_y-\tau_y\xi_z)Q(\xi_y+2\xi_z t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))}{\xi_y^3}+\\
&\tfrac{1}{\xi_y^4}3d(\tau_z\xi_y-\tau_y\xi_z)Q(\xi_y+3\xi_z t_y)(\xi_z y_0-\xi_z(Y_c+Y_p)-\xi_y z_0+\xi_y(Z_c+Z_p))^2\Big)\Big)
\end{aligned}
$$

8.2.3 Model Term Dependent on the Shift along the LHCb Z-direction, δz

$$U^{v}_{model}\left(\delta z\right) = \delta z \times \Big(\Big(-\frac{(\xi_y t_x - \xi_x t_y)(\xi_y + \xi_z t_y)}{\xi_y^2} - \frac{2Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))}{\xi_y^2} -$$

$$\frac{3dQ(\xi_y + 3\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2}{\xi_y^3} \Big) \Big/$$

$$\Big(-\tau_x + \frac{(\xi_y + \xi_z t_y)((\tau_z \xi_y - \tau_y \xi_z) t_x + \xi_x(\tau_y - \tau_z t_y))}{\xi_y^2} +$$

$$\frac{2(\tau_z \xi_y - \tau_y \xi_z) Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))}{\xi_y^3} +$$

$$\frac{1}{\xi_y^4} 3d(\tau_z \xi_y - \tau_y \xi_z) Q(\xi_y + 3\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2 \Big) -$$

$$\Big(\Big(\frac{2(\tau_z \xi_y - \tau_y \xi_z) Q(\xi_y + 2\xi_z t_y)}{\xi_y^2} +$$

$$\frac{1}{\xi_y^3} 6d(\tau_z \xi_y - \tau_y \xi_z) Q(\xi_y + 3\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) \Big)$$

$$\Big(-x_0 + X_c + X_p - \frac{Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2}{\xi_y^3} -$$

$$\frac{dQ(\xi_y + 3\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^3}{\xi_y^4} -$$

$$\frac{1}{\xi_y^2}(\xi_y + \xi_z t_y)(t_x(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) + \ \xi_x(-y_0 + Y_c + Y_p + t_y z_0 - t_y(Z_c + Z_p))))) /$$

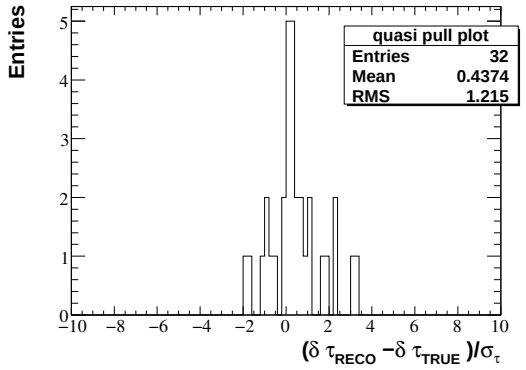
$$\Big(-\tau_x + \frac{(\xi_y + \xi_z t_y)((\tau_z \xi_y - \tau_y \xi_z) t_x + \xi_x(\tau_y - \tau_z t_y))}{\xi_y^2} +$$

$$\frac{1}{\xi_y^3} 2(\tau_z \xi_y - \tau_y \xi_z) Q(\xi_y + 2\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p)) +$$

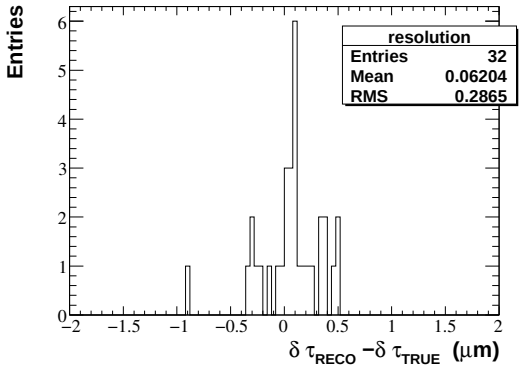
$$\frac{1}{\xi_y^4} 3d(\tau_z \xi_y - \tau_y \xi_z) Q(\xi_y + 3\xi_z t_y)(\xi_z y_0 - \xi_z(Y_c + Y_p) - \xi_y z_0 + \xi_y(Z_c + Z_p))^2 \Big)^2 \Big)$$

8.3 Plots of Alignment Parameters in the Absence of Fields

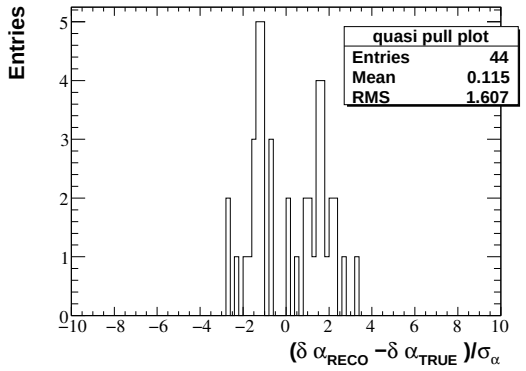
8.3.1 Particle Gun Data



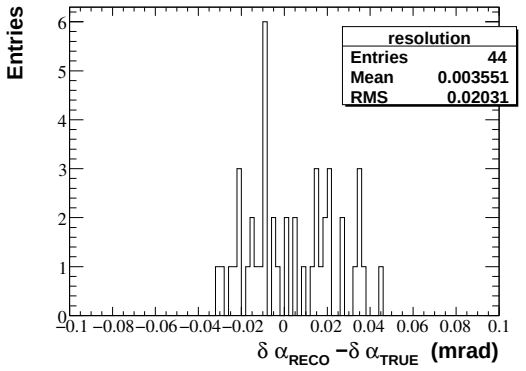
(a) Quasi-pull plot $\delta\tau$



(b) Error-Resolution plot $\delta\tau$



(c) Quasi-pull plot $\delta\alpha$



(d) Error-Resolution plot $\delta\alpha$

Figure 8.5: Alignment results after 27 iterations, quasi-pulls and resolution plots for a simultaneous alignment of $\delta\tau$ - shift in direction of measurement - and $\delta\alpha$ - stereo angle misalignment.

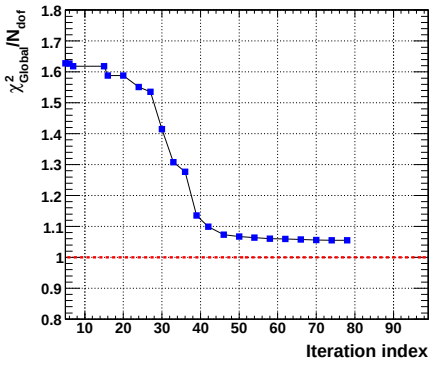
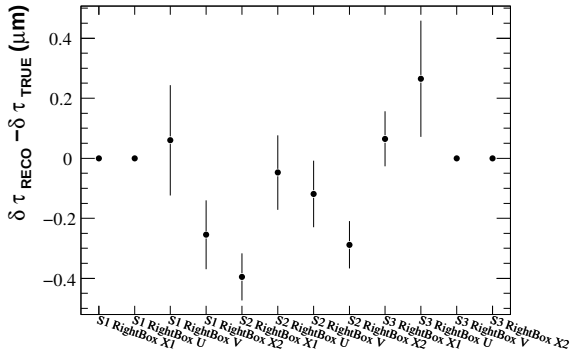


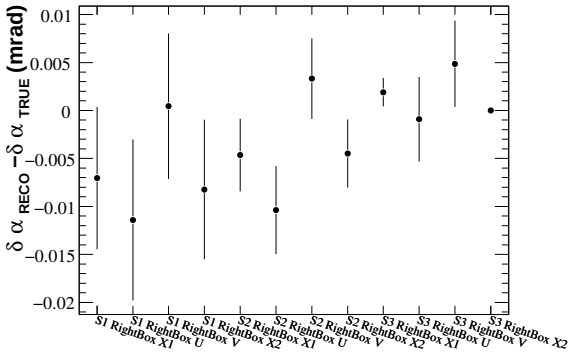
Figure 8.6: $\frac{\chi^2}{N_{dof}}$ evolution with convergence to 1.05 value.

8.3.2 10 TeV, Open-VELO Alignment Plots Data

Resolution Plots for Newton-Raphson Procedure starting from an "Ideal" Alignment

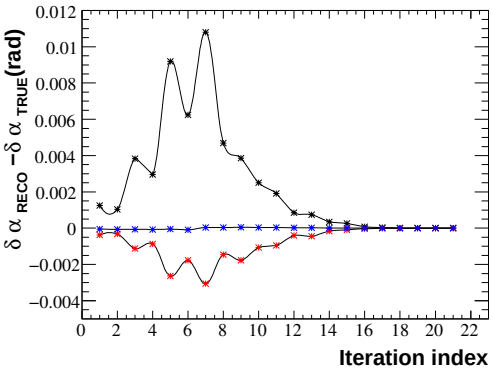


(a) $\delta\tau$ resolution plot, with Millepede error estimates



(b) $\delta\alpha$ resolution plot, with Millepede error estimates

Figure 8.7: Alignment results after 21 iterations, with corresponding Station, Box, Layer index label



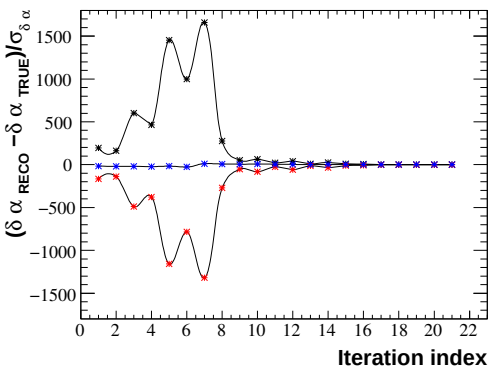


Figure 8.8: Alignment progression with iterations for rotation around beam axis, the effect of the χ^2 quality cut is seen after the 7th iteration when the bias due to random scattering is reduced, and convergence becomes evident. The plotted quantities correspond to: Station 1 Top Box Layer X2, Station 2 Left/A-Box Layer U, Station 3 Bottom Box Layer U. (rank 3, 25, 37 in 5.16 plots)

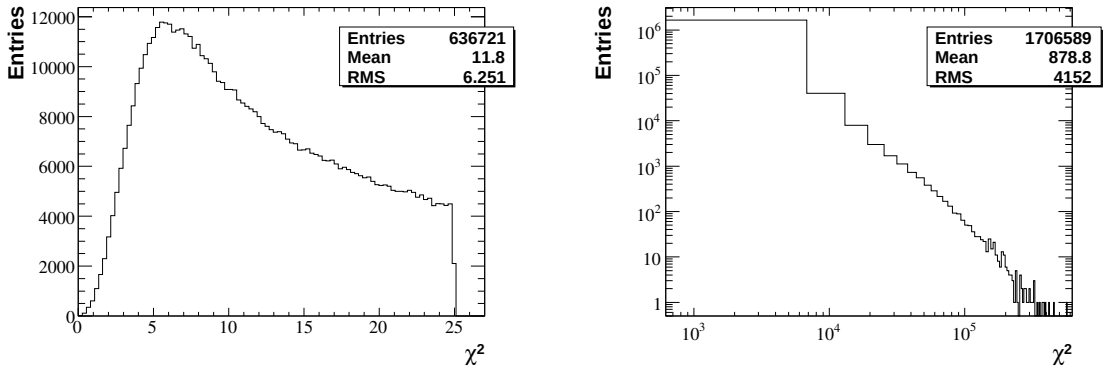
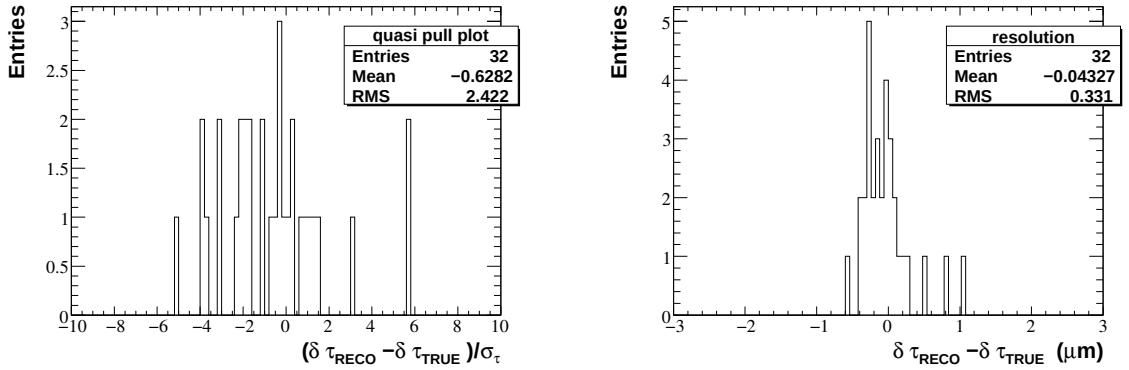
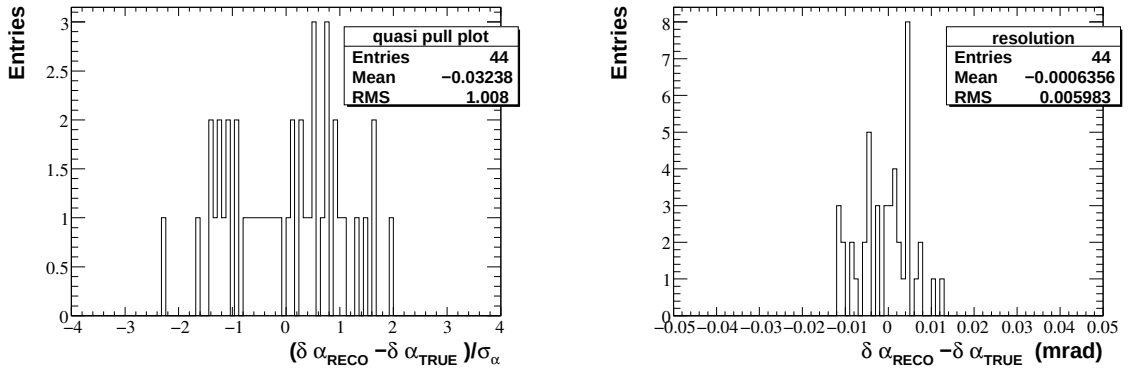


Figure 8.9: Estimated track-χ^2 in the last iteration, here the 21st. Quality cut equivalent with a 3.15 σ probability rejection. The effect of random scattering still evident in the smearing of the data points



(a) Quasi-Pull of $\delta\tau$

(b) Error plots $\delta\tau$

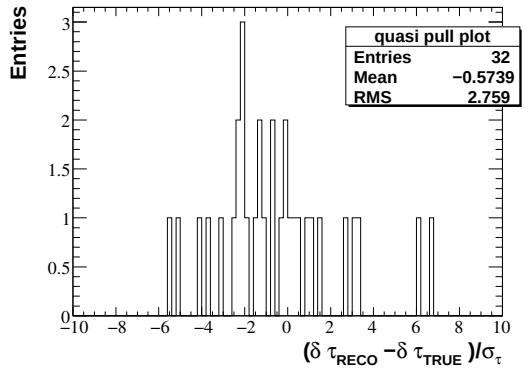


(c) Quasi-Pull of $\delta\alpha$

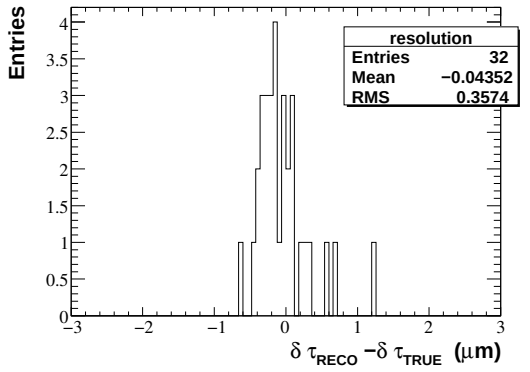
(d) Error plots $\delta\alpha$

Figure 8.10: Alignment results after 21 iterations, quasi-pulls and residual plots for a simultaneous alignment of $\delta\tau$ - shift in direction of measurement - and $\delta\alpha$ - stereo angle misalignment. All non-a priori fixed DoF for IT-Layers

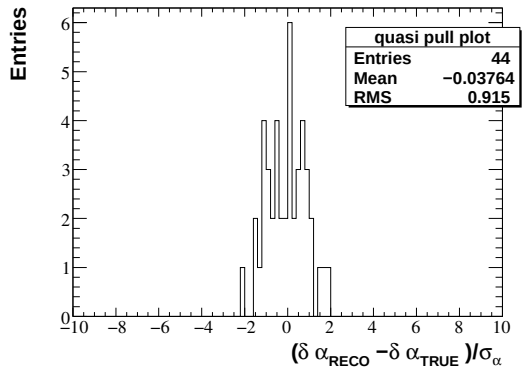
Quasi-Pull Plots for Newton-Raphson Procedure starting from a Misaligned Geometry



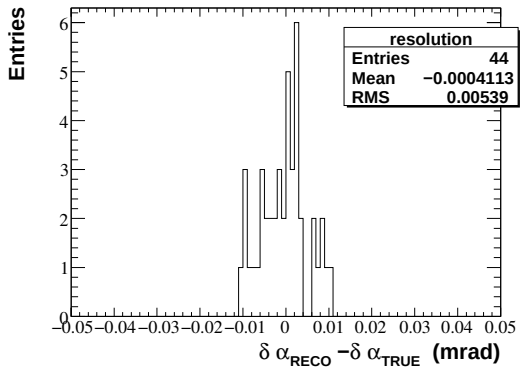
(a) Quasi-Pull of $\delta\tau$



(b) Error plots $\delta\tau$



(c) Quasi-Pull of $\delta\alpha$



(d) Error plots $\delta\alpha$

Figure 8.11: Alignment results after 21 iterations, quasi-pulls and residual plots for a simultaneous alignment of $\delta\tau$ - shift in direction of measurement - and $\delta\alpha$ - stereo angle misalignment. Non-null starting values

8.3.3 Residuals for 3074 Run Data , 5TeV+5TeV B-off Minimum Bias, Misaligned Geometry

The plots in 8.12 and 8.13 show what happens if the alignment in figures 5.22 and 5.23 gets 25 times less tracks in the used sample. The alignment impact on the residual distributions of previous sensors can be compared with the same effect for distinct additional sensors, effect quantified in plots 8.14 and 8.15.

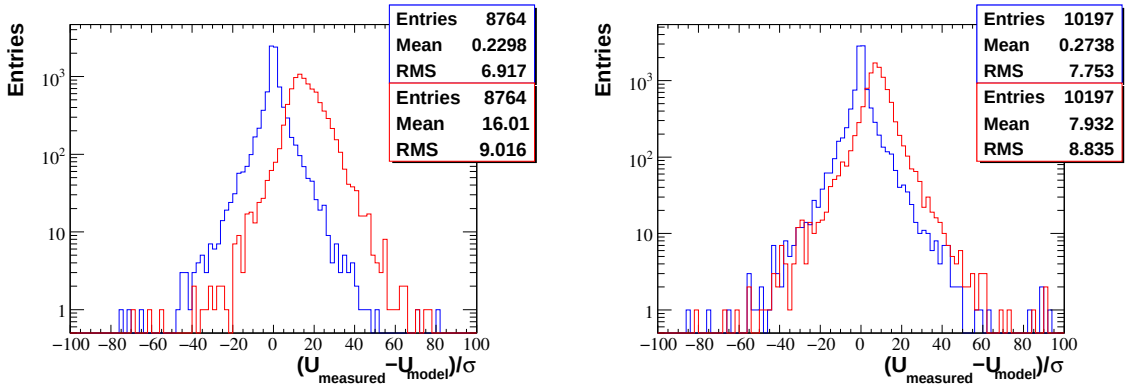


Figure 8.12: Plot of residuals for measurements in two sensors: left a sensor in 1st Station, right a sensor in 3rd Tracking Station. Red histograms are before alignment and blue after alignment. Only 30K Minimum Bias events were used in alignment. Tracks not used in alignment dominant in blue histogram. The sensors are the central-Top X2 and central-Bottom X1 respectively

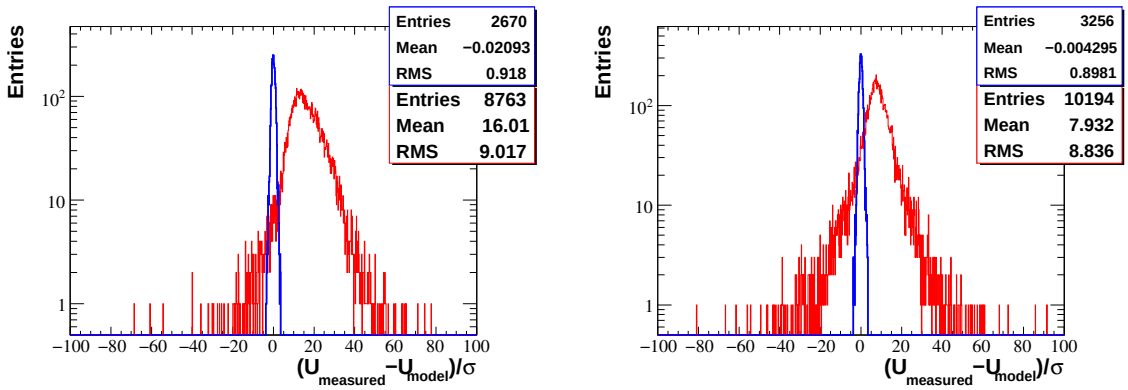


Figure 8.13: Plot of residuals for measurements in two sensors: left a sensor in 1st Station, right a sensor in 3rd Tracking Station. Red histograms are before alignment and blue after alignment. Only 30K Minimum Bias events were used in alignment. The blue plot include only the track used in the last iteration step of alignment. Sensors as in 8.12

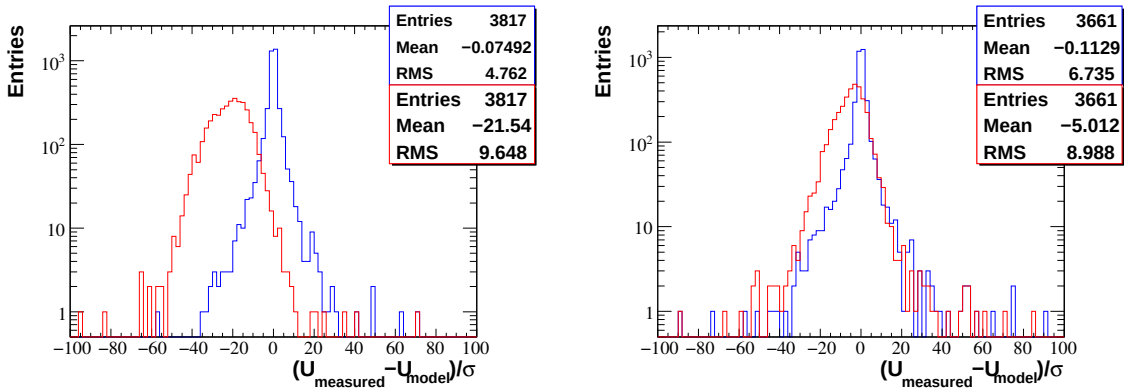


Figure 8.14: Plot of residuals for measurements in two sensors, left a sensor in 1st Station, right a sensor in 3rd Tracking Station. Red histograms are before alignment and blue after alignment. Only 30K Minimum Bias events were used in alignment. Tracks not used in alignment dominant in blue histogram. Sensors are in A-Box layers

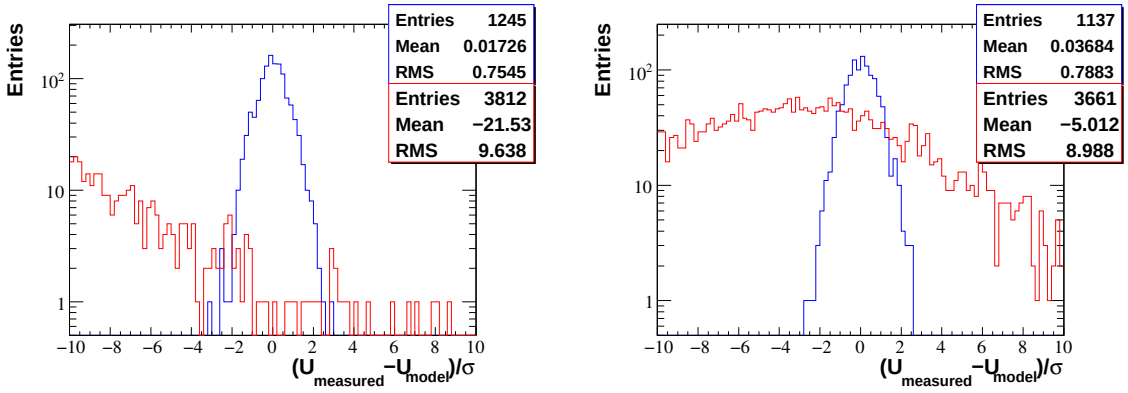


Figure 8.15: Plot of residuals for measurements in two sensors, left a sensor in 1st Station, right a sensor in 3rd Tracking Station. Red histograms are before alignment and blue after alignment. Only 30K Minimum Bias events were used in alignment. The blue plot includes only the tracks used in the last iteration step of alignment.

Alignment job for 21K Minimum Bias at 14 TeV and with Closed VELO

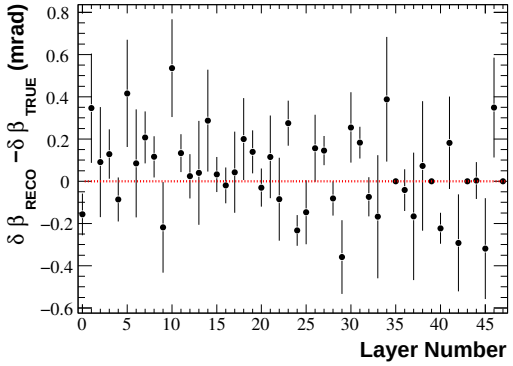
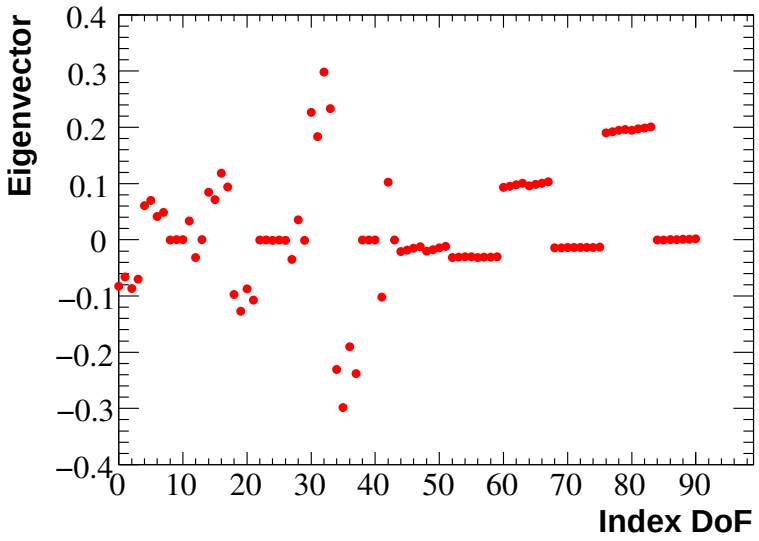
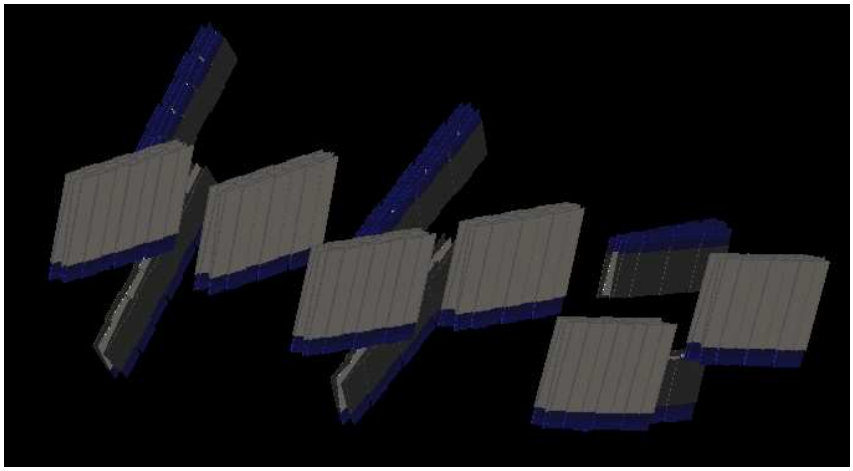


Figure 8.16: $\delta\beta$ resolution plots

8.3.4 Weak Modes in a Null Field

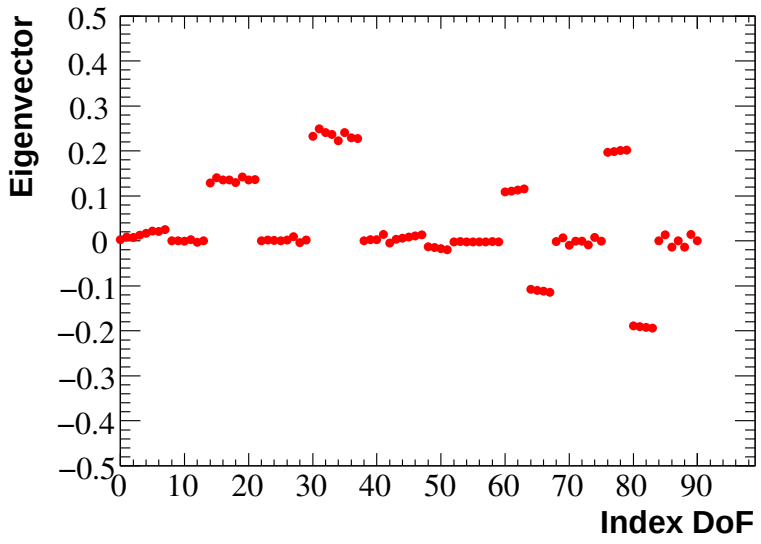


(a) 2nd Weak Mode, eigenvector

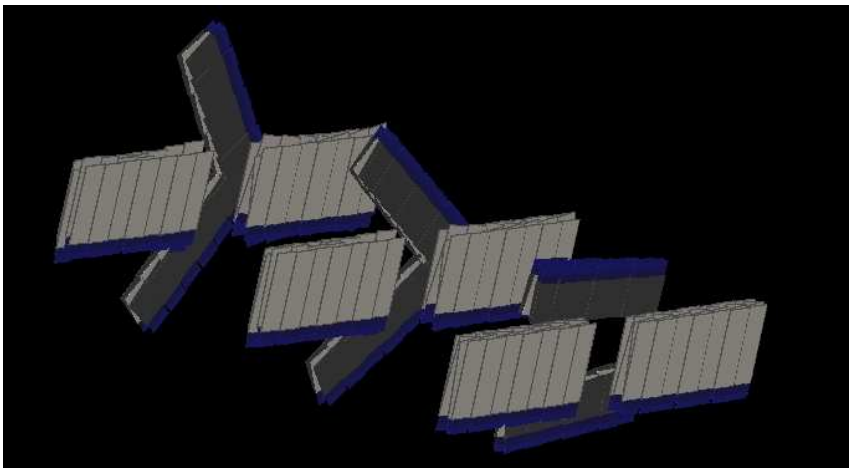


(b) 2nd Weak Mode, Panoramix display

Figure 8.17: Alignment matrix second eigenvector with 91 non-fixed alignment parameters, 47 $\delta\alpha$, and 44 $\delta\tau$

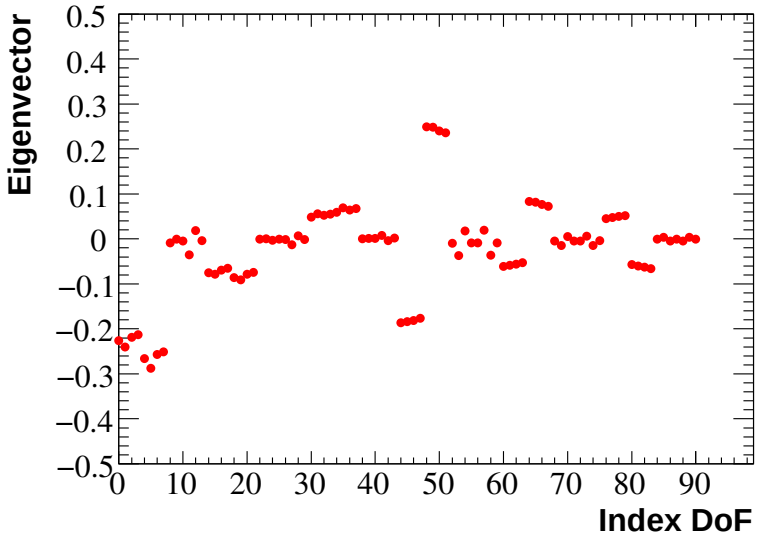


(a) 3rd Weak Mode, eigenvector

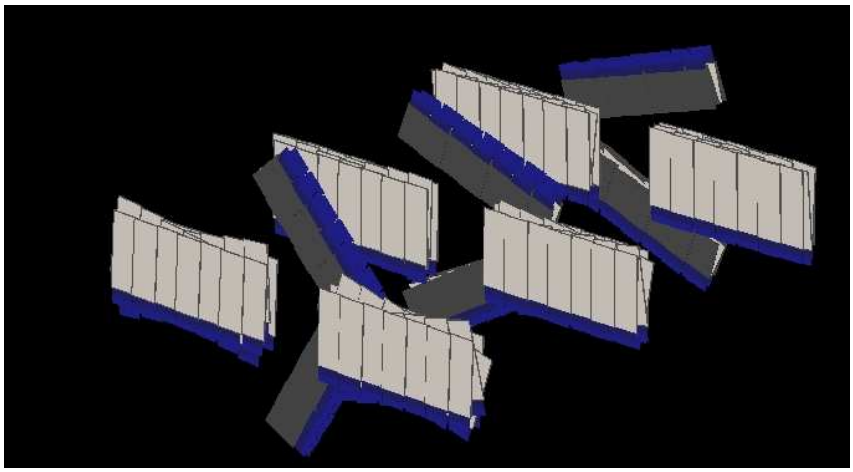


(b) 3rd Weak Mode, Panoramix display

Figure 8.18: Alignment matrix third eigenvector with 91 non-fixed alignment parameters, 47 $\delta\alpha$, and 44 $\delta\tau$



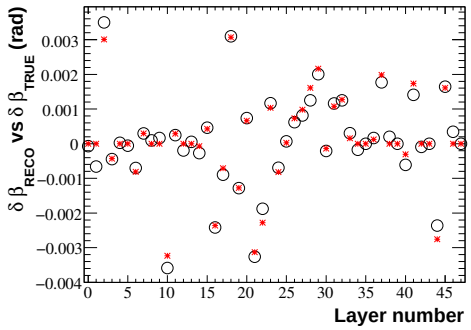
(a) 4th Weak Mode, eigenvector



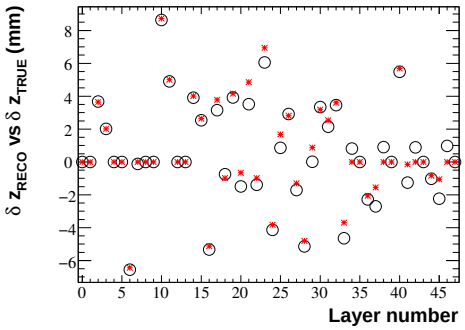
(b) 4th Weak Mode, Panoramix display

Figure 8.19: Alignment matrix fourth eigenvector with 91 non-fixed alignment parameters, 47 $\delta\alpha$, and 44 $\delta\tau$

8.3.5 Alignment for non-Null Field, all 5 Alignment Parameters Simultaneously

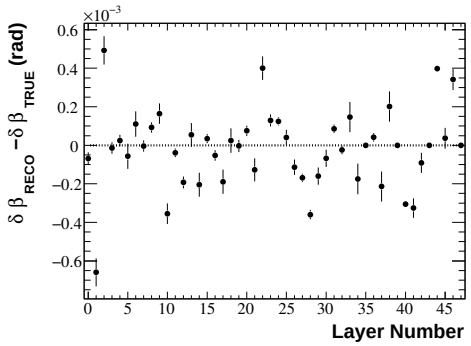


(a) $\delta\beta$ degrees of freedom

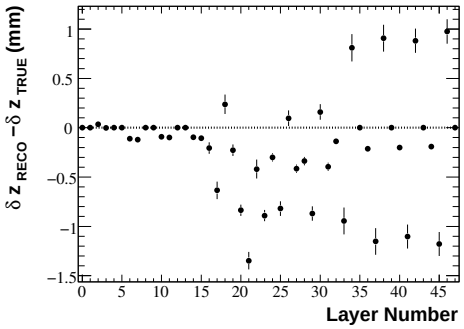


(b) δz degrees of freedom

Figure 8.20: $\delta\beta$ and δz input misalignments versus the reconstructed Alignment parameters, Millepede errors estimates are too small on this plot. Alignment done for all 5 types of alignment parameters

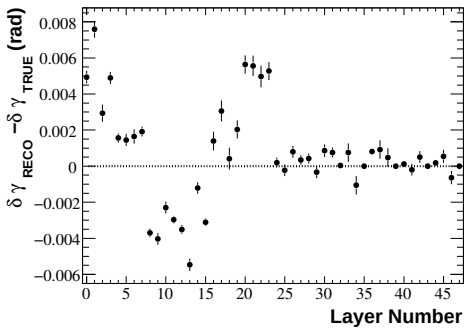


(a) $\delta\beta$ degrees of freedom

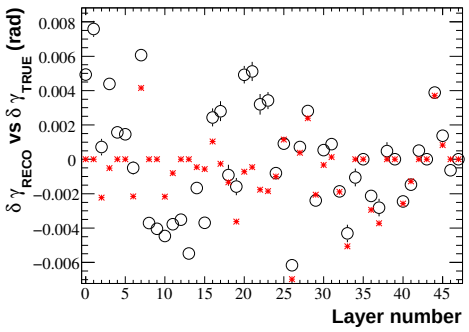


(b) δz degrees of freedom

Figure 8.21: Alignment resolutions with Millepede errors estimates, for 2 types of alignment parameters given, a simultaneous alignment of all 5 DoF.



(a) Resolution plot of the $\delta\gamma$ degrees of freedom



(b) reconstructed $\delta\gamma$'s versus the input misalignment

Figure 8.22: Alignment resolutions with Millepede errors estimates, for 1 type of alignment parameter given, a simultaneous alignment of all 5 DoF.

8.3.6 Weak Modes in a non-Null Field

Layer Alignment

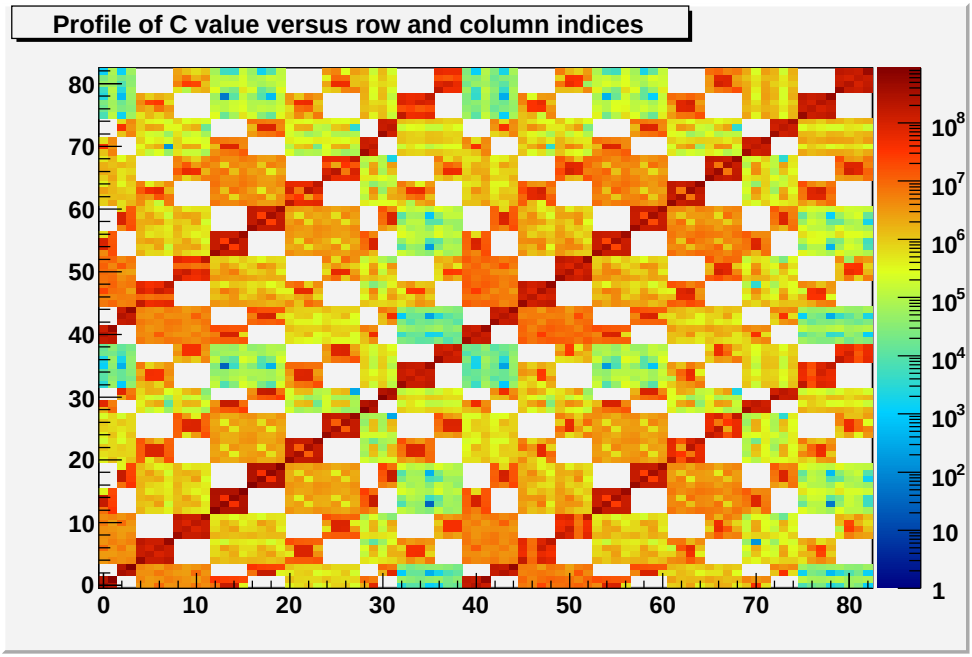


Figure 8.23: Normalized alignment matrix for a minimally constrained IT-layers system, 83 $\delta\alpha$ and $\delta\tau$ degrees of freedom.

Ladder Alignment

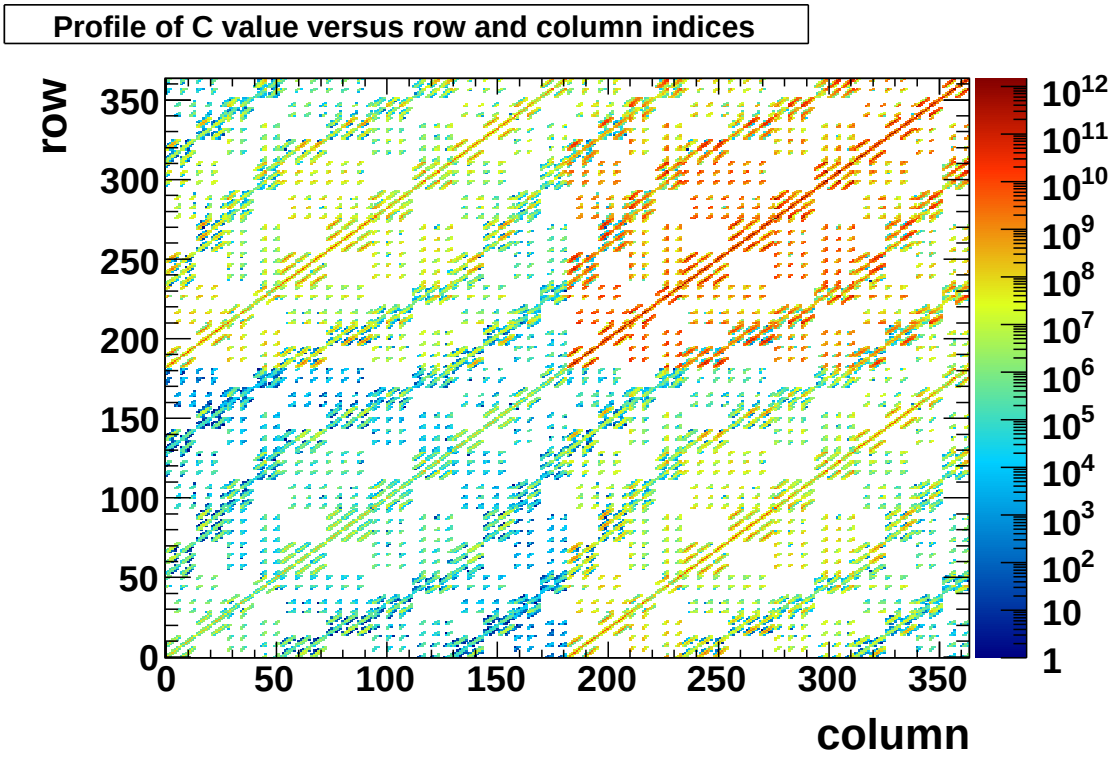


Figure 8.24: Alignment matrix for the IT ladders-sensors system. Without an a priori normalization scheme applied. $\delta\alpha$, $\delta\tau$ degrees of freedom.

Bibliography

[1] D. H. Perkins, *Introduction to High Energy Physics.*, 4th ed. (Cambridge University Press, Cambridge, UK, 2000).

[2] T.D. Lee and C.N. Yang, Phys. Rev. **104**, 257 (1956).

[3] C. Wu et al., Phys. Rev. **105**, 1413 (1957).

[4] J.H. Christenson, J.W. Cronin, V.L. Fitch, and R. Turlay, Phys. Rev. Lett. **13**, 138 (1964).

[5] M. Kobayashi and T. Maskawa, Prog. Theor. Phys. **49**, 652 (1973).

[6] Cheng, Ta-Pei and Li, Ling-Fong, *Gauge theory of elementary particle physics* (Oxford University Press Inc., New York, US, 1996).

[7] D. J. Griffiths, *Introduction to Elementary Particles.* (Wiley, John and Sons, Inc., New York, US, 1987).

[8] C. Amsler and others (Particle Data Group), Physics Letters **B667**, 1 (2008).

[9] Phys. Rev. Lett. **97**, 242003 (2006).

[10] M. Beyer, *CP Violation in Particle, Nuclear and Astrophysics* (Springer, Berlin, Germany, 2002).

[11] U. Uwer, Advanced particle physics, Heidelberg University Lecture, 2008.

[12] M. Britsch, Private communication, 2008.

[13] M. Schmelling, Private communication, 2008.

[14] The LHCb Collaboration, CERN/LHCC 2003-030 (2003).

[15] Lyndon Evans and Philip Bryant, Journal of Instrumentation **3** (2008).

[16] The LHCb Collaboration, A Augusto Alves Jr et al , JINST **3** (2008).

[17] P. B.-M. e. a. LHCb Collaboration, CERN Report No. CERN-LHCC/2002-029, 2003 (unpublished).

[18] S. Brandt, *Data Analysis*, Third ed. (Springer-Verlag, New York, US, 1998).

[19] M. Schmelling, From raw data to physics - statistical and numerical methods in data analysis, XVI. Heidelberger Graduiertenkurse Physik, 2006.

[20] P. R. Bevington and D. K. Robinson, *Data Reduction and Error Analysis for the Physical sciences*, Second ed. (McGraw-Hill, New York, US, 1992).

[21] V. Blobel, Nuclear Instruments and Methods A **566**, 5 (2006).

[22] V. Blobel, *Millepede II: The Manual*, http://www.desy.de/~blobel/mptalks.html.

[23] F. Maciuc, *Alignment of the LHCb Inner Tracker Modules*, http://old.dpg-tagungen.de/program/dortmund/t710.pdf, Talk given at the "Deutsche Physikalische Gesellschaft Frühjahrstagungen", Dortmund, 28-30 of March 2006.

[24] F. Maciuc, *Alignment of the LHCb Tracking System*, http://indico.cern.ch, Talk given at the Silicon Tracker Meeting , Zurich, 7th of March 2006.

[25] F. Maciuc, *Alignment of the LHCb Tracking System with straight tracks*, http://indico.cern.ch, Talk given at the Track Reconstruction for DC06 Meeting in Lausanne, 21st of March 2006.

[26] H. W. Press *et al.*, *Numerical Recipes in C, the art of scientific computing*, Second ed. (Cambridge University Press, New York, US, 1998).

[27] Wolfram Research Inc., *Mathematica*, Version 7.0 ed. (Wolfram Research Inc., Champaign, Illinois, US, 2008).

[28] D. Brown, Alignment experience from babar/aleph, The 1st LHC Detector Alignment Workshop, June, 2007, http://physics.syr.edu/~lhcb/public/alignment/LHCAlignmentWorkshop, 2007.

[29] O. Callot and M. Schiller, Private communication, 2008.

[30] J. Blouw, Private communication, 2007.

[31] J. Blouw and M. Schiller, Private communication, 2007-2008.

[32] S. Blusk, Private communication, 2007.

[33] GSL Team, *Singular Value Decomposition*, 2007, GNU Scientific Library, chapter 13.4 of the Reference Manual.

[34] S. Blusk, Private communication, 2007.

[35] W. Hulsbergen, Private communication, 2008.

[36] B. Andreea and W. Hulsbergen, ATLAS Note indet-pub-2007-009 (2007), cdsweb.cern.ch/record/1039585/files/indet-pub-2007-009.pdf.