TECHNISCHE UNIVERSITÄT WIEN

VIENNA UNIVERSITY OF TECHNOLOGY

DISSERTATION

Novel Concepts for Optimization of the CERN Large Hadron Collider Injection Lines.

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines Doktors der technischen Wissenschaften unter der Leitung von

Dr. Jörg Wenninger
CERN, BE-OP-SPS

Privatdoz. Dipl.-Ing.
Dr.techn. Michael Benedikt
E141, Atominstitut

eingereicht an der Technischen Universität Wien
bei der Fakultät für Physik

von

Mag. Kajetan Fuchsberger
Matrikelnummer: 9630500
Ebenweg 11, 5321 Koppl

Wien, am 16.5.2011

Diese Dissertation wurde unterstützt vom österreichischen Bundesministeriums für Wissenschaft und Forschung, im Rahmen des CERN Technologie-Doktorandenprogramms.

Kurzfassung

Der *Large Hadron Collider* (LHC) ist derzeit der Teilchenbeschleuniger mit der größten Schwerpunktsenergie weltweit und ist daher das vielversprechendste Instrument für teilchenphysikalische Entdeckungen in der nahen Zukunft. Die Transferlinien TI2 und TI8, die den Teilchenstrahl vom letzten Vorbeschleuniger, dem *Super Proton Synchrotron*, (SPS) zum LHC transportieren, sind mit einer Gesamtlänge von ca. 6 km die längsten der Welt, was eine hochpräziese Abstimmung der Strahloptik unumgänglich macht.

Tests in den Jahren 2004 bis 2008 zeigten einige, bis dahin unerwartete, Effekte in diesen Linien auf: Eine Assymetrie der Betatronphase zwischen den beiden transversalen Ebenen, eine Fehlanpassung der Dispersion am Übergang zwischen den Transferlinien und dem LHC und eine unerwartet hohe transversale Kopplung an der selben Stelle.

In dieser Arbeit stellen wir die Methoden und Softwaretools vor, die wir speziell für die Untersuchung dieser Ungereimtheiten entwickelt haben. Wir beschreiben die Analyse der vorhandenen Messdaten, Messungen der Strahloptik der Transferliniene und die Berechnung von entsprechenden Korrekturen. Weiters zeigen wir, dass die Fehlanpassungen der Optik durch eine Sextupolkomponente in den Hauptdipolmagneten der Transferlinien erklärt werden können. Diese Sextupolkomponente wurde von uns aus Optikmessungen mit Strahl abgeleitet und konnte später durch numerische Simulationen der Magnete bestätigt werden.

Schließlich beschreiben wir die Maßnahmen, die getroffen wurden um die zugrundeliegenden numerischen Modelle zu verbessern und demonstrieren die sehr gute Übereinstimmung der Messungen mit den neuen Modellen anhand von Kick-Response und Dispersion bis zu zweiter Ordnung, was eine excellente Strahlqualität im LHC garantiert.

Abstract

The *Large Hadron Collider* (LHC) is presently the particle accelerator with the highest center of mass energy in the world and is for that reason the most promising instrument for particle physics discoveries in the near future. The transfer lines TI2 and TI8 which transfer the beam from the last pre-accelerator, the *Super Proton Synchrotron* (SPS), to the LHC are with a total length of about 6 km the longest ones in the world, which makes it necessary to do optics matching with high precision.

Tests between 2004 and 2008 revealed several, previousely unpredicted, effects in these lines: An assymetry in betatron phase between the two transverse planes, a dispersion mismatch at the injection point from the transfer lines to the LHC and unexpectedly strong transverse coupling at the same location.

In this thesis, we introduce the methods and tools that we developed to investigate these discrepancies. We describe the analysis of the available data, measurements of the transfer line optics and the calculation of optics corrections. Further we show that the optics mismatch can be explained from a sextupolar field component in the injection main bends, which we deduced from beam measurements and later was confirmed by numerical magnet simulations.

Finally, we describe the measures taken to improve the underlaying numerical models and demonstrate the very good measurement-model agreement for kick response measurements and dispersion up to second order, which guarantees an excellent beam quality in the LHC.

Acknowledgements

First of all, I want to thank my parents, Johann and Monika Fuchsberger, for giving me the chance to visit an excellent school, which I know was not a matter of course. Thanks for making it possible for me to study and for all the support in all the years. I also thank all other members of my family: My brother Daniel, for our friendly relationship, although we are so different in many aspects and so similar at the same time. My sisters Bernadette and Lucia, for being as they are; I always felt for them as they were my own children, and now that they are grown up they are the best friends one could wish to have! Finally, my only grandmother who is still alive, for all the confidence she has in me.

Concerning this thesis, I especially want to thank my supervisor at CERN, Dr. Jörg Wenninger for his excellent support. He was always present to discuss various topics, to answer my frequent questions and to explain relevant facts with endless patience. Especially, for my natural integration into the operations group and the related hands-on experience he should take the credit. Further thanks to Dr. Michael Benedikt, my supervisor at the technical university of vienna, as well as to Dr. Gerald Badurek and Dr. Johannes Aiginger for their dedication to the Austrian Doctoral Student Program and their down-to-earth approach in supervising external students.

For the many very fruitful and enlightening discussions, I want to thank above all my friend Tobias Bär, who shared an office with me for almost two years. He always was prepared to interrupt his work to answer my silly questions or listen to my neverending sermon on software design and clean code. Especially, I will miss his many instantanous ideas and his analysis approaches, which were very often totally orthogonal to mine.

All the work on the transfer lines was done in very close collaberation with TE-APT group. Special thanks to Malika Meddahi, Brennan Goddard and Volker Mertens for sharing their thoughts, for their great support, and for respecting my carrier decisions. Great thank also to Marcel Gyr for organizing the numerous section events of the BTP section, which made me feel welcome. Related to the transfer line optics, also special thanks to Stephane Farthouk for his many ideas in this matter and his help with the accelerator physics part.

The list of people which crossed my way at CERN is endless and every single one of them helped me to develop my personality and my skills in its own way. Apologizing in advance to those I have forgotten, I cannot resist to name at least some of them: Rüdiger Schmidt for helping develop the thread through my thesis. Ralph Steinhagen,

Verena Kain, Laurette Ponce, Stefano Redaelli, Reyes Alemany and all other EICs for the many discussions on orbit feedback and any kind of operational matters. Frank Schmidt and Werner Herr for the MadX support. Gabriel Müller and Xavier Buffat for the discussions and their contributions to JMad. The whole BE-CO-APP section, most of all Vito Baggiolini and Roman Gorbonosov, for their support and especially for respecting my opinion in software related discussions.

All my colleagues and friends in my home village Koppl, in Graz and here in Geneva deserve a special round of applause. It would have been impossible for me to even come close to finish my studies without your encouraging presence, the exchanges of ideas and of often very controversial opinions and beliefs, both in professional and private matters.

Thank you all!

Contents

1. Introduction

1.1. Motivation

The *Large Hadron Collider* (LHC) is presently the particle accelerator with the highest center of mass energy in the world and is for that reason the most promising instrument for particle physics discoveries in the near future. To make such discoveries possible for the experiments located in four caverns in the ring, it is essential to provide a beam of high quality. The particle beam has to pass several pre-accelerators before finally arriving in in the LHC where it is accelerated to its final energy. The challenge is to ensure the quality right from the beginning and especially to pass the beam from one pre-accelerator to the next without major quality-losses.

The transfer lines TI2 and TI8 which transfer the beam from the last pre-accelerator, the *Super Proton Synchrotron* (SPS), to the LHC are with a total length of about 6 km the longest ones in the world. While traditionally transfer lines were considered as less critical (because they were relatively short and had a large aperture), in the case of the LHC transfer lines it is the first time that it is necessary to do optics matching in high precision.

Therefore, a lot of effort was put in the previous years into the careful design and a detailed understanding of their properties. On one hand, a detailed model description of the lattice is crucial for successful operation. Such a model was developed and unified during the design phase [GGKR04]. On the other hand, a lot of experience could be gained already before the LHC startup, since the transfer lines were ready for operation already in 2007 [MAG+09]. Already during these transfer line tests, an unexplained discrepancy between the numerical model and the measurements was observed: An asymmetry in betatron phase between the two transverse planes [Wen06, Wen08].

In 2008, during injection tests, beam was brought into the LHC the very first time. During these tests the correct functionality of the injection systems were tested and the quality of the optics matching between the transfer lines and the LHC was verified. These injection tests revealed two more, previously unpredicted, effects, namely a clear dispersion mismatch at the injection point from the transfer lines to the LHC and unexpectedly strong transverse coupling at the same location [MAF+08].

Since both effects can lead to emittance growth in the LHC, several attempts were made to find the sources of these discrepancies by analysis of trajectory- and

dispersion data with all in all unsatisfying results. The present thesis is an attempt to step back and find a physically plausible solution to these problems. We will elaborate the idea that field errors in the main bending magnets of the transfer lines could be the sources of the problems. One major challenge of this project is to measure the various optics parameters of the transfer lines. While for a closed ring there exist very well established methods to measure the betatron phase and beta functions [CC96], the measurement of lattice functions of a transfer line is a non trivial task. The beam only passes the transfer line once and therefore the resolution of the *Beam Position Monitors* (BPMs) is limited and the trajectory depends strongly on the initial conditions and therefore on the stability of the injector.

The basic idea is to use model-fits to kick response data, also known as the LOCO[1]-principle [Saf97], to determine the lattice parameters. This analysis method requires a close coupling between observations and the numerical model. The previous work in unifying the models will enable us to combine these models. Instead of treating each part of the beam-path (SPS, TI2/TI8, LHC) separately, the whole system is considered as one long 'transfer line'. By this method e.g. problems between the different parts become visible, while they tend to be hidden in the classical (separated) approach.

The extraction from the SPS into the transfer lines is already covered in [KS05]. This thesis focuses on the transition from the transfer lines to the LHC (TI2/TI8-LHC junctions). The objective is to develop a model for the transfer lines which includes all the observed effects and can be used to match the optics of the transfer lines correctly to the LHC optics and thus ensure the required emittance preservation.

To conveniently perform the necessary numerical analysis of the measured data, new software tools are necessary. Since such tools are of general interest, for example to find optics errors in the initial commissioning phase of an accelerator when the whole ring is not available yet, the tools can be used for various problems. This is possible by generalizing the access to the underlying model of the machine under investigation and by unifying the definition of such accelerator models. The related software design considerations and implementation are also part of this work.

1.2. Structure of this Document

Chap. 2, Chap. 3 and Chap. 4 are intended to be introductory chapters. Chap. 2 describes the environment in which the topics treated in this thesis are settled, in Chap. 3 the most important facts of accelerator physics related to this thesis are stated and in Chap. 4 some measurement methods for accelerator lattices are introduced.

Then in Chap. 5 we introduce the LHC injection transfer lines and sketch in detail the related inconsistencies which were observed during the initial beam comissioning

[1] LOCO is the abbreviation for '*Linear Optics from Closed Orbits*'.

phases. Chap. 6 is denoted to the theoretical background of the analysis procedure which was used to investigate the transfer line optics. Then in Chap. 7 the most important features and implementation details of the computational tools, which were developed especially for that analysis, are summarized.

The analysis of the transfer line optics is described in detail and results are presented in Chap. 8. Finally, Chap. 9 contains the relevant conclusions and provides some ideas for further developments.

2. CERN and the LHC

2.1. CERN

The *European Organization for Nuclear Research*, CERN (from french: *Conseil Européen pour la Recherche Nucléaire*) was founded in 1954. Its purpose is clearly stated in the CERN convention of 1953 [CER53]:

> The Organization shall provide for collaboration among European States in nuclear research of a pure scientific and fundamental character, and in research essentially related thereto. The Organization shall have no concern with work for military requirements and the results of its experimental and theoretical work shall be published or otherwise made generally available.

While the currently 20 member states of CERN contribute the main part of the capital and the operational costs of CERNs programs, many contributions also come from observer states and non-member states. CERN is located near Geneva at the border of Switzerland and France. A detailed summary of facts about CERN, its history and its mission can for example be found on the CERN public website [CER08].

2.2. The LHC and the CERN Accelerator Chain

The first accelerator which became operational at CERN was the 600 MeV *Synchro-cyclotron* (SC) which was built in 1957 and was operational until 1990. The oldest accelerator at CERN which is still under operation is the PS, the *Proton Synchrotron*, which became operational in 1959. Today CERN plays a leading role in high energy physics research, because it hosts the particle accelerator with the highest center of mass energy in the world, the LHC (*Large Hadron Collider*).

The LHC is a proton-proton collider with a design top-energy of 14 TeV (center of mass). A ion program is also foreseen with a design top-energy of 5.52 TeV per nucleus (center of mass). The LHC is located in the tunnel formerly used by LEP (*Large Electron Positron Collider*), on average 100 m underground. Figure 2.1 illustrates the location of the tunnel below the border of France and Switzerland.

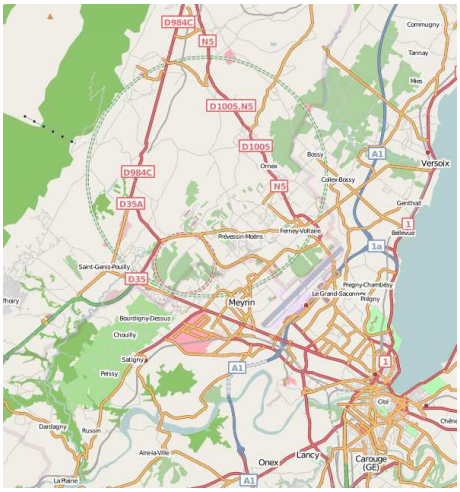


Figure 2.1.: Location of the LHC in the Geneva area (green tunnel).

Although many other research programs use beams from the various particle accelerators at CERN, we will focus on the role of these accelerators as pre-accelerators for the LHC.

The accelerator chain leading from protons at rest to protons at an energy of 7 TeV is shown in Fig. 2.2: First the protons are accelerated by a linear accelerator, LINAC2 and transferred to the *PS Booster* (PSB). Then they are injected into the PS, passed on to the SPS (*Super Proton Synchrotron*) and finally they reach the LHC. Table 2.1 summarizes the way of the protons through the pre-accelerators and states the corresponding top energies of each accelerator for LHC operation.

For ion operation of the LHC, lead ions are accelerated in LINAC3, accumulated in LEIR (*Low Energy Ion Ring*) and transferred into the PS, from whereon they follow the same paths as protons.

Accelerator	Circumference	Top Energy [GeV]
LINAC2		0.05
PSB	157 m	1.4
PS	628 m	26
SPS	7 km	450
LHC	27 km	7000

Table 2.1.: The LHC accelerator chain for proton operation.

Some of the characteristic parameters of the LHC are listed in Table 2.2. This

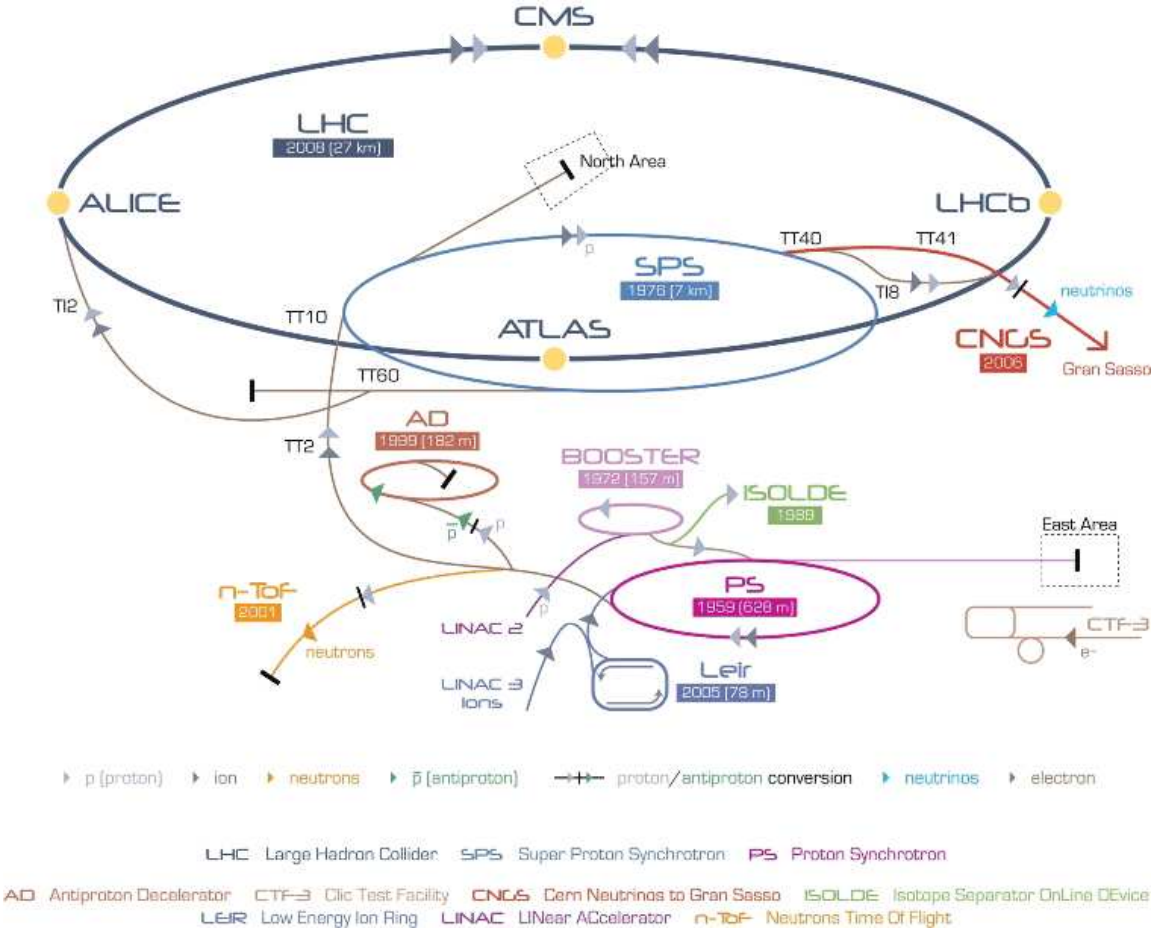


Figure 2.2.: The CERN accelerator complex.

information was taken from the LHC design report [BCL+04], where also explanations of the physical meanings of these parameters can be found. Some of them are also explained in Chap. 3.

parameter	unit	injection	collisions
General			
Ring circumference	[m]	26658.883	
Number of collision points		4 (IP1, IP2, IP5, IP8)	
Ring separation in arcs	[mm]	194	
Number of main bends		1232	
Length of main bends	[m]	14.3	
Field in main bends	[T]	0.535	8.33
Bending radius	[m]	2803.95	
Lattice			
Horizontal tune		64.28	64.31
Vertical tune		59.31	59.32
Momentum compaction factor		$3.225{\cdot}10^{-4}$	
Gamma transition γ_{tr}		55.68	
Maximum dispersion in arc	[m]	2.018 (H), 0.0 (V)	
Minimum horizontal dispersion in arc	[m]	0.951	
Maximum β in arc	[m]	177 (H), 180 (V)	
Minimum β in arc	[m]	30 (H), 30 (V)	
β at IP1 and IP5	[m]	18	0.55
β at IP2	[m]	10	0.5 for Pb, 10 for p
β at IP2	[m]	10	1.0 . . . 50
RF System			
Revolution frequency	[kHz]	11.245	
RF frequency	[MHz]	400.8	
Harmonic number		35640	
Total RF voltage	[MV]	8	16
Beam			
Proton energy	[GeV]	450	7000
Relativistic gamma		479.6	7461
Number of particles per bunch		$1.15{\cdot}10^{11}$	
Number of bunches		2808	
Longitudinal emittance	[eVs]	1.0	2.5
Transverse normalized emittance	[μm rad]	3.5	3.75
Circulating beam current	[A]	0.582	
Stored energy per beam	[MJ]	23.3	362

Table 2.2.: LHC characteristic parameters.

3. Short Summary of Accelerator Physics

3.1. Introduction

The aim of this chapter is to summarize the basic concepts of accelerator physics that will be used in the remaining chapters. No detailed derivations will be shown, since these are beyond the scope of this text and can be found in any textbook of accelerator physics, for example [Wil00, Hin08, Cha99, Lee04]. On the contrary, we will simply state the properties which define accelerator optics and give the most important related formulas.

3.2. Notation

For the description of a particle trajectory in an accelerator one commonly introduces a local co-rotating coordinate system. This system uses the ideal particle's trajectory (design trajectory) as reference for its origin (Fig. 3.1). Using this coordinate system, the movement of the individual particles is thus treated as a small perturbation around the specified design trajectory.

In this coordinate system, s denotes the position along the design trajectory relative to an arbitrary but fixed position s_0 in a ring or relative to the start of the transfer line. x and y denote the positions in the two transverse planes, where x is the horizontal position with positive values towards the outside of the ring and y is the vertical position with positive values upwards. Sometimes it is convenient to use a more general notation for transverse motions. In these cases we use u to label one of the two transverse planes (x or y). x' and y' denote the derivatives of the transverse coordinates with respect to the longitudinal coordinate s,

$$u' = \frac{\partial u}{\partial s}. \tag{3.1}$$

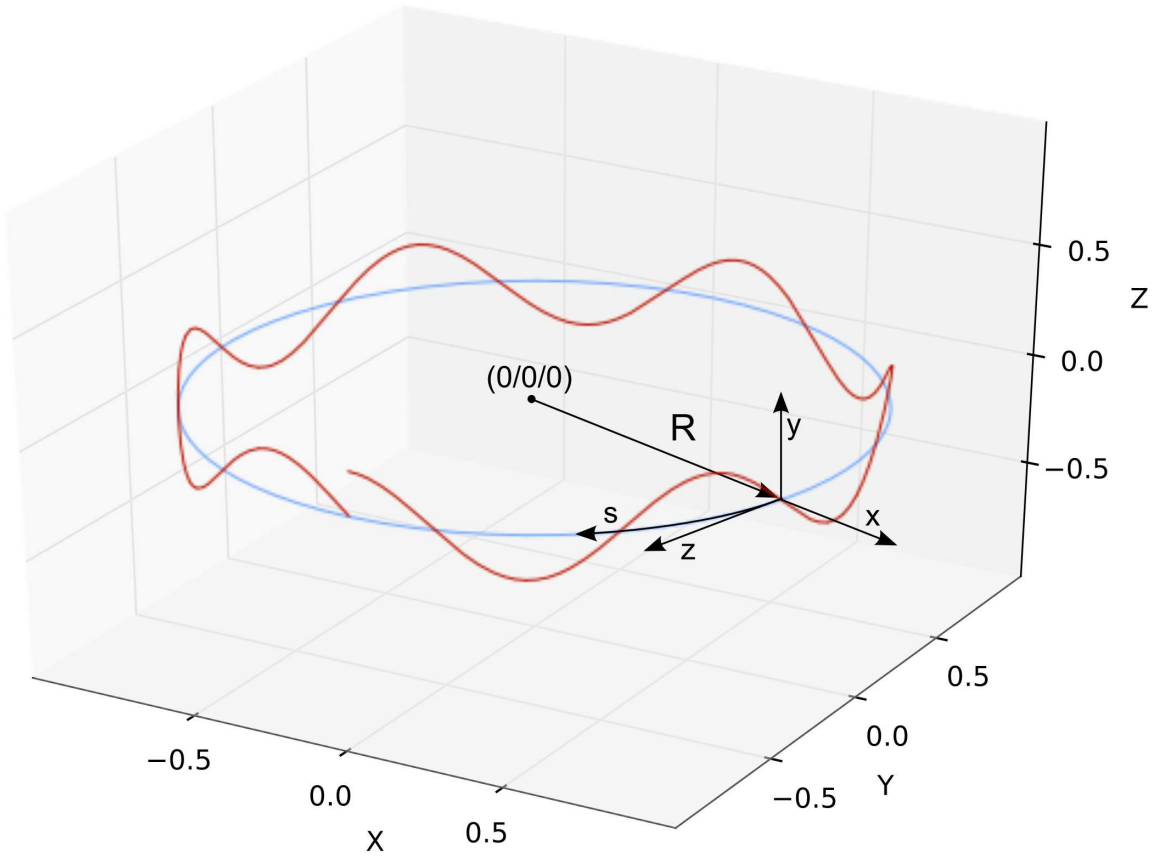


Figure 3.1.: Co-rotating coordinate system.

3.3. Transverse Beam Dynamics

In a particle accelerator, the transverse motion is dominated by magnetic fields. Their influence on the momentum $\vec{p}$ of the particle with charge q is given by the Lorentz force

$$\dot{\vec{p}} = q \cdot \left(\vec{v} \times \vec{B} \right), \tag{3.2}$$

with $\vec{B}$ being the magnetic field vector and $\vec{v}$ the particle velocity. If the field vector $\vec{B}$ is orthogonal to the momentum $\vec{p}$ then the equilibrium of Lorentz force and centrifugal force leads to the definition of the beam rigidity $B\rho$ [Wie07, p 39]:

$$B\rho = \frac{p}{q}. \tag{3.3}$$

Here ρ denotes the bending radius and B and p are given by $B = \left|\vec{B}\right|$ and $p = |\vec{p}|$.

The equation of motion for the particles are commonly derived from Eq. (3.2), keeping only the leading terms in linear approximation. This leads to (see e.g. [Hin08, p 126 ff]):

$$x''(s) + K_x(s)x(s) = \frac{1}{\rho(s)}\frac{\Delta p}{p}, \tag{3.4a}$$

$$y''(s) + K_y(s)y(s) = 0, \tag{3.4b}$$

with $\frac{\Delta p}{p}$ denoting the momentum offset and s the longitudinal position within the ring. K_x and K_y are related to the strength of the quadrupolar fields $K(s)$ and the term $\frac{1}{\rho^2}$, stemming from the dipolar fields:

$$K_x = \frac{1}{\rho^2} - K(s) = \frac{1}{\rho^2} + \frac{1}{B\rho}\frac{\partial B_y}{\partial x}, \tag{3.5a}$$

$$K_y = K(s) = -\frac{1}{B\rho}\frac{\partial B_y}{\partial x}. \tag{3.5b}$$

The sign of the quadrupole strength is arbitrary. Here it was chosen such that $K < 0$ represents a horizontally focusing quadrupole. The equations of motions are uncoupled for the horizontal and the vertical plane. Therefore, the horizontal and vertical movements are independent and can be treated separately.

Equations (3.4) are Hill's type differential equations. The homogeneous part of these equations (i.e. $\frac{\Delta p}{p} = 0$) can be solved by the use of *Floquet's Theorem*. The solution is given by

$$u_\beta(s) = \sqrt{\varepsilon_u \beta_u(s)}\cos\left(\mu_u(s) + \mu_u(s_0)\right), \tag{3.6}$$

with the initial conditions ε_u and $\mu_u(s_0)$. β_u is the so-called *beta function* for the plane u which is per definition always positive. It has maxima at focusing quadrupoles and minima at defocussing ones. The *betatron phase* μ for the plane u is related to the beta function via

$$\mu_u(s) = \int_{s_0}^{s} \frac{d\bar{s}}{\beta_u(\bar{s})}. \tag{3.7}$$

s_0 denotes an arbitrary but fixed longitudinal position in the ring or the starting point in a transfer line. ε_u is the transverse emittance for the plane u. It is discussed in more detail in Sec. 3.3.3.

Due to periodic boundary conditions, in a ring a stationary solution of type Eq. (3.6) can be found. This is called the *(closed) orbit*. In a transfer line, an infinite amount of solutions exist, which only depend on the initial conditions at the start of the line.

3.3.1. Dispersion

As indicated by Eq. (3.5a), the trajectory of particles with non-zero momentum deviations $\frac{\Delta p}{p}$ are modified by dipolar fields. A linear ansatz can be made for the solution of the inhomogeneous equation:

$$u = u_\beta + D_u \frac{\Delta p}{p}. \tag{3.8}$$

u denotes the position in the plane u, D_u the (linear) dispersion function in the plane u, u_β the solution of the homogeneous Hill's equation (betatron oscillation, Eq. (3.6)) and $\frac{\Delta p}{p}$ is the relative momentum change w.r.t. the nominal momentum. So we can define the linear dispersion for the plane u as

$$D_u = \frac{\partial u}{\partial \left(\frac{\Delta p}{p}\right)}. \tag{3.9}$$

3.3.2. Tune and Chromaticity

The particle trajectory has oscillatory behavior as indicated by Eq. (3.6). These transverse oscillations are called *betatron oscillations*, since they were first observed in betatron accelerators [Ker41]. In a ring, the number of betatron oscillations per revolution is called the *tune*. The tune Q_u of an accelerator for the plane u can be calculated from the phase advance and thus from the beta function by

$$Q_u = \frac{1}{2\pi}\left(\mu_u(s+C) - \mu_u(s)\right) = \frac{1}{2\pi}\oint_C \frac{d\overline{s}}{\beta_u(\overline{s})}. \tag{3.10}$$

C denotes the circumference of the accelerator.

From Eq. (3.5) together with Eq. (3.3) it follows that the normalized strength $K(s)$ is inversely proportional to the particle momentum. This means that at a given energy e.g. a particle with higher momentum than the design momentum experiences less focussing strength than the design focussing strength. Therefore, the tune of an individual particle depends on its momentum. In a linear approximation, the chromaticity ξ is defined for the plane u as proportionality factor between momentum offset and tune change:

$$\Delta Q_u = \xi_u \cdot \frac{\Delta p}{p}. \tag{3.11}$$

In order to reduce tune spread (defines the bandwidth of the different tune values of

the particles per beam) and to avoid head-tail instabilities[1], sextupole magnets are used to correct the negative natural chromaticity of a machine (which we denote by Q'_u) [Tur96, pp. 77-100].

3.3.3. Transverse Emittance and Beam Size

For one transverse plane u, each particle in a storage ring follows an ellipse in phase space (u, u') which is described by the so-called Courant-Snyder invariant a, which is defined by

$$a_u = \gamma_u u^2 + 2\alpha_u u u' + \beta_u u'^2. \tag{3.12}$$

The area of the ellipse is given by πa_u. The functions α and γ are (together with β) the so-called Courant-Snyder functions which are defined for the plane u by

$$\alpha_u(s) = -\frac{\beta'_u(s)}{2} \tag{3.13}$$

and

$$\gamma_u(s) = \frac{1 - \alpha_u^2(s)}{\beta_u(s)}. \tag{3.14}$$

For a particle beam, which consists of many particles with a certain distribution in phase space, each particle follows its own Courant-Snyder ellipse within the distribution. As a characteristic property of such a beam, the so-called transverse emittance ε for the plane u is defined by

$$\varepsilon_u = \sqrt{\langle u^2 \rangle \langle u'^2 \rangle - \langle u u' \rangle^2}. \tag{3.15}$$

$\langle .. \rangle$ denotes the average value of the coordinates over the distribution. If the accelerator is composed only of linear elements, like dipoles and quadrupoles, then the emittance is conserved. It is equal to the Courant-Snyder ellipse of the rms particle [Lee04, p54].

During acceleration, the longitudinal momentum is increased while the transverse momentum is not affected. Thus u', the angle between particle trajectory and design trajectory, decreases. Therefore, the transverse emittance decreases according to Eq. (3.15). This effect is called *adiabatic damping*. For that reason, a more convenient quantity for the operation of a hadron storage ring is the *normalized emittance* ε_n which is defined for the plane u as

$$\varepsilon_{un} = \varepsilon_u \gamma_r \beta_r, \tag{3.16}$$

[1] Short range interactions between particles within a bunch, in combination with the interplay of betatron and longitudinal oscillation, can lead to unstable motions of the particles. The amplitude of these oscillations depends on the chromaticity [San69].

where β_r and γ_r are the relativistic factors given by

$$\beta_r = \frac{v}{c} \tag{3.17}$$

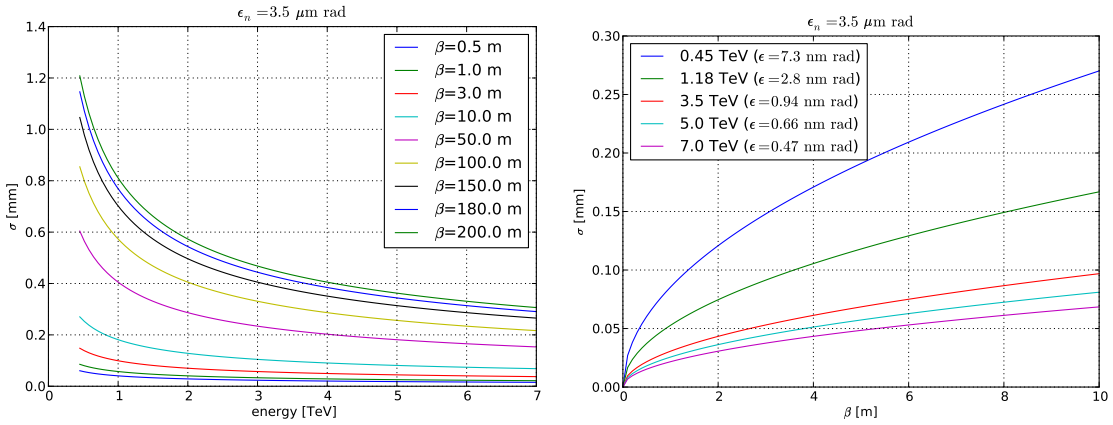
(with v the particle velocity and c the speed of light in vacuum) and

$$\gamma_r = \frac{1}{\sqrt{1 - {\beta_r}^2}}. \tag{3.18}$$

The transverse size of the beam σ_u in the plane u is defined as the standard deviation of the transverse particle distribution and is given by

$$\sigma_u = \sqrt{\beta_u \varepsilon_u}. \tag{3.19}$$

Since the beta-function depends on the longitudinal position s in the accelerator, $\beta_u = \beta_u(s)$, also the beam size depends on the position in the ring. As an example, Figs. 3.2 illustrate the dependence of the beam size in the LHC on the two relevant parameters energy and beta.



(a) versus energy (b) versus beta

Figure 3.2.: LHC Beam size dependence on beam energy and beta-value.

3.3.4. Transfer Matrices and Orbit Response

As soon as the phase space coordinates (beam position and its derivative) at one point in the ring are known, the coordinates can be calculated for any other position

in the ring using the transfer matrix M:

$$\begin{pmatrix} u_2 \\ u_2' \end{pmatrix} = M_{12} \begin{pmatrix} u_1 \\ u_1' \end{pmatrix}. \tag{3.20}$$

M_{12} denotes the transfer matrix from point 1 to point 2 in the ring, which is parametrized by the Courant-Snyder functions as follows [Cha99, p 65]:

$$M_{12} = \begin{pmatrix} \sqrt{\frac{\beta_2}{\beta_1}}\big(\cos(\mu_{12}) + \alpha_1 \sin(\mu_{12})\big) & \sqrt{\beta_1\beta_2}\sin(\mu_{12}) \\ -\frac{1+\alpha_1\alpha_2}{\sqrt{\beta_1\beta_2}}\sin(\mu_{12}) + \frac{\alpha_1-\alpha_2}{\sqrt{\beta_1\beta_2}}\cos(\mu_{12}) & \sqrt{\frac{\beta_1}{\beta_2}}\big(\cos(\mu_{12}) - \alpha_2\sin(\mu_{12})\big) \end{pmatrix} \tag{3.21}$$

For the sake of simplicity, we omitted the index u for the Courant-Snyder functions. μ_{12} denotes the phase difference between the two points:

$$\mu_{12} = \mu_2 - \mu_1. \tag{3.22}$$

Without derivation (this can e.g. be found in [Hin08, pp 289 ff]) we also state the relation for the orbit change at the position 2, resulting from a dipolar kick, which is given by

$$\Delta u_2 = \frac{\sqrt{\beta_1\beta_2}\cos\left(|\mu_1 - \mu_2| - \pi Q\right)}{2\sin\left(\pi Q\right)} \cdot \delta_1 \tag{3.23}$$

for a ring and by

$$\Delta u_2 = \begin{cases} \sqrt{\beta_1\beta_2}\sin\left(\mu_1 - \mu_2\right) \cdot \delta_1 & \text{for } \mu_2 > \mu_1, \\ 0 & \text{otherwise.} \end{cases} \tag{3.24}$$

for a transfer line. δ_1 denotes the dipolar kick (in the plane u) at the position 1 in the ring and Δu_2 the orbit change at the position 2. Again the plane indices u are omitted for the Courant-Snyder functions.

3.4. Luminosity

The luminosity L is the the key measure for the performance of a particle collider. It relates the event rate $\frac{dN_i}{dt}$ for a certain particle process i to the cross section σ_i of the process:

$$\frac{dN_i}{dt} = L\sigma_i. \tag{3.25}$$

The Luminosity (without any crossing angles) is given by

$$L = \frac{N^2 k_b f}{4\pi\sigma_x\sigma_y}. \tag{3.26}$$

N denotes the particles per bunch, k_b the number of bunches per beam and σ_x and σ_y are the horizontal and vertical beam sizes at the interaction point, respectively. The unit of the instantaneous luminosity is $\mathrm{cm}^{-2}\mathrm{s}^{-1}$ and the integrated luminosity is given in inverse barns b^{-1} ($1\ \mathrm{barn} = 10^{-24}\,\mathrm{cm}^2$).

4. Lattice Measurement Methods

4.1. Introduction

In this chapter we will sketch how some of the optics properties of an accelerator can be measured and explain how this is done in practice. An overview of lattice measurement methods can be found e.g. in [ZM03] or [Bra09, pp 361 ff]. Although there exist other measurement methods, like multi turn measurements [CC96], which might have their advantages for a ring, we will only focus on the two measurement methods which we have chosen to analyze the optics of the transfer lines.

4.2. Kick Response

The key quantity for kick response measurements is the so-called orbit response matrix R. Assuming a perfect linear optics, then the orbit response matrix describes the effect of a set of N_C corrector kicks on the position readings of each of the N_M *Beam Position Monitors* (BPMs),

$$\vec{u} = R\vec{\delta}. \tag{4.1}$$

$\vec{u}$ is defined as the vector of positions (x or y) measured at the monitors

$$\vec{u} = \begin{pmatrix} u_1 \\ u_2 \\ \cdots \\ u_{N_M} \end{pmatrix}, \tag{4.2}$$

(N_M is the number of monitors) and $\vec{\delta}$ is the vector of kicks of corrector dipole-magnets

$$\vec{\delta} = \begin{pmatrix} \delta_1 \\ \delta_2 \\ \cdots \\ \delta_{N_C} \end{pmatrix}, \tag{4.3}$$

(N_C is the number of correctors). So the response matrix has N_M rows and N_C columns. While the response matrix is the key tool for orbit- and trajectory-steering

(where in general the theoretical response matrix is used), it also can be easily measured as described in the following section. So it can be used to verify the optics model of an accelerator as will be described in Chap. 6.

4.2.1. Measurement

The easiest way to measure the response matrix is to acquire two trajectories where the strength of only one kicker is changed between the two acquisitions. Then all elements of one column of the matrix (corresponding to the corrector j) are simply calculated by taking the difference of the two trajectories (or to orbits in a ring) at each monitor and dividing by the kick-difference between the two acquisitions. So the elements of the response matrix are then given by

$$R_{ij}^{\text{meas}} = \frac{\Delta u_i}{\Delta \delta_j}, \tag{4.4}$$

where Δu_i is the change in the reading of ith monitor and $\Delta \delta_j$ the change in the kick of corrector j between the two acquisitions. R^{meas} denotes the measured response matrix.

For the CERN accelerators this type of measurement is smoothly integrated in the controls environment: An automated procedure in the steering tool YASP (*Yet Another Steering Program* [Wen05]), which trims the necessary kicks one after the other and writes the resulting orbits to files. These files can be directly imported into Aloha (*Another Linear Optics Helper Application*, see Sec. 7.2) for the analysis.

4.2.2. Model

The derivation of the orbit response matrix from the model is as straightforward as the measurement. We sketch two different methods here, which have their own advantages and disadvantages.

Analytical Calculation

The simplest approach is to calculate the response matrix by the use of the analytical expressions of the transfer matrix between two points in the ring or transfer line, respectively. Eq. (3.23) and Eq. (3.24) directly result in analytical expressions for the elements of the response matrix. They are given by

$$R_{ij}^{\text{model}} = \frac{\sqrt{\beta_i \beta_j}\cos\left(|\mu_i - \mu_j| - \pi Q\right)}{2\sin\left(\pi Q\right)} \tag{4.5}$$

for a ring and by

$$R_{ij}^{\text{model}} = \begin{cases} \sqrt{\beta_i \beta_j} \sin\left(\mu_i - \mu_j\right) & \text{for } \mu_i > \mu_j, \\ 0 & \text{otherwise.} \end{cases} \tag{4.6}$$

for a transfer line. β_i denotes the beta function at the monitor and β_j the beta function at the corrector, μ_i and μ_j are the phases at the monitor and corrector, respectively and Q denotes the tune of the machine.

The advantage of this method is, that it only needs the optics functions for the lattice (β, μ and Q) which only have to be calculated once (e.g. by an optics code like MadX) or can be retrieved from other sources (like a database or pre-calculated file). The calculation of the response matrix elements is then quickly done by using Eq. (4.5) or Eq. (4.6). This is sufficient for lattices where the optics is (at least nearly) linear.

Figure 4.1 shows the effect of a corrector kick in a ring (only a part of the ring is shown). As also visible from Eq. (4.5), the kick changes the closed orbit at all locations in the ring. For a transfer line, as illustrated in Fig. 4.2, the trajectory is only altered at locations downstream of the used corrector. This is also expressed in Eq. (4.6).

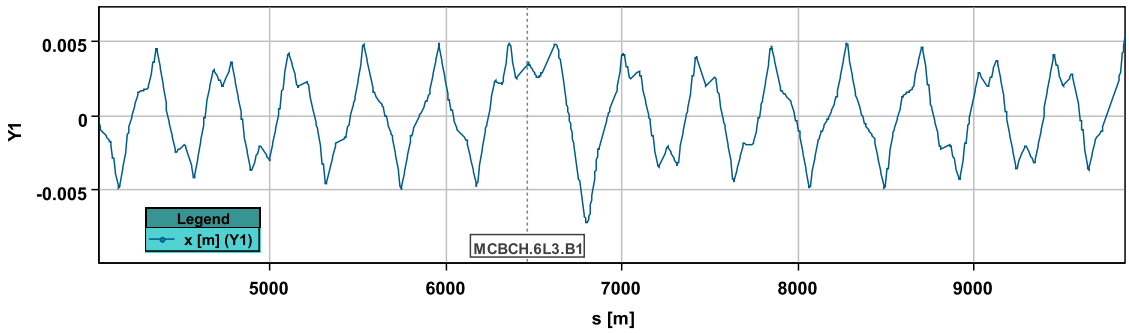


Figure 4.1.: Effect of a corrector-kick (horizonzal corrector MCBCH.6L3.B1) on the orbit in a ring. The orbit without the kick would be zero at all s-positions.

Numerical Calculation

A drawback of the previously described method is that it does for example not include coupling or nonlinear effects. The necessity for such features arose for example during the investigation of the coupling effects at the TI8[1]-LHC junction. To be able to

[1]TI8 denotes the transfer line injecting into LHC beam 2. See Chap. 5

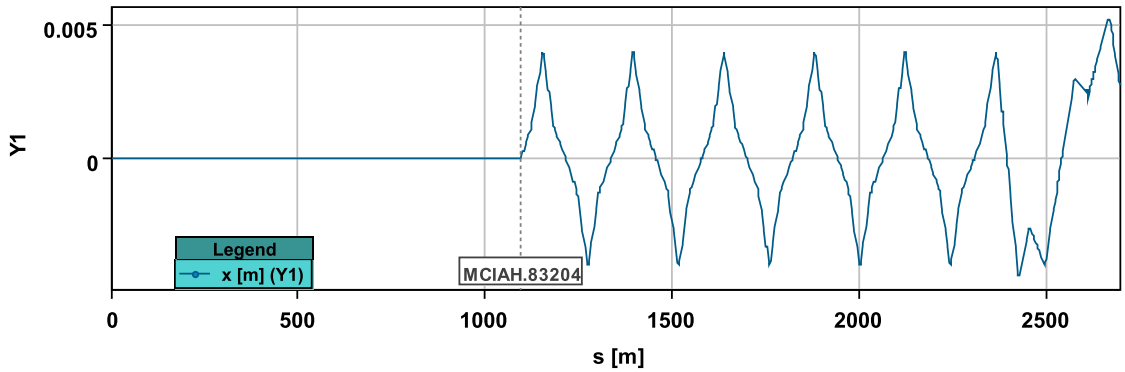


Figure 4.2.: Effect of a corrector-kick (horizontal corrector MCIAH.83204) on the trajectory in a transfer line. The trajectory without the kick would be zero at all s-positions.

model all these effects, the response matrix can be calculated just in the same way as the measurement is done:

One calculates the effect of the kick of one corrector at a time, ideally by the same kick strength as during the measurement. The response matrix can then be calculated using the same formula as for the measurement (Eq. (4.4)):

$$R_{ij}^{\text{model}} = \frac{\Delta u_i}{\Delta \delta_j}, \tag{4.7}$$

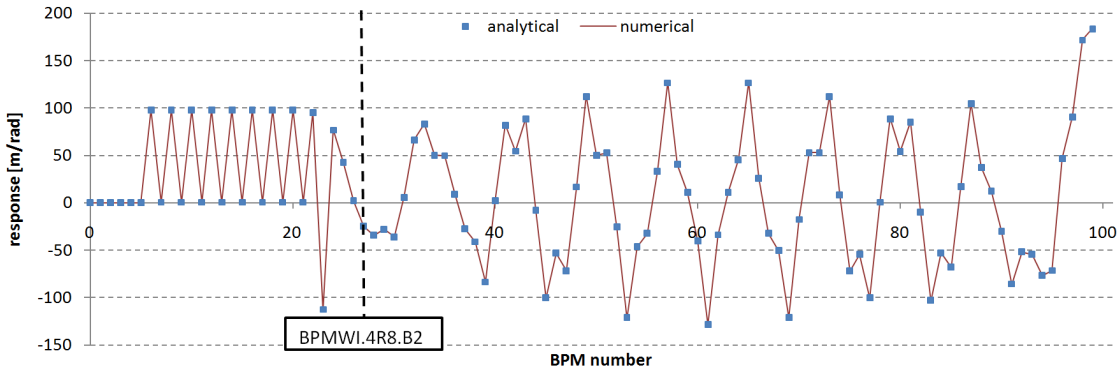
The drawback of this method is that the required multiple *twiss runs*[2] take considerably more time, especially for closed orbits and large machines (like the LHC). Nevertheless, this is in general the preferred method.

Comparison

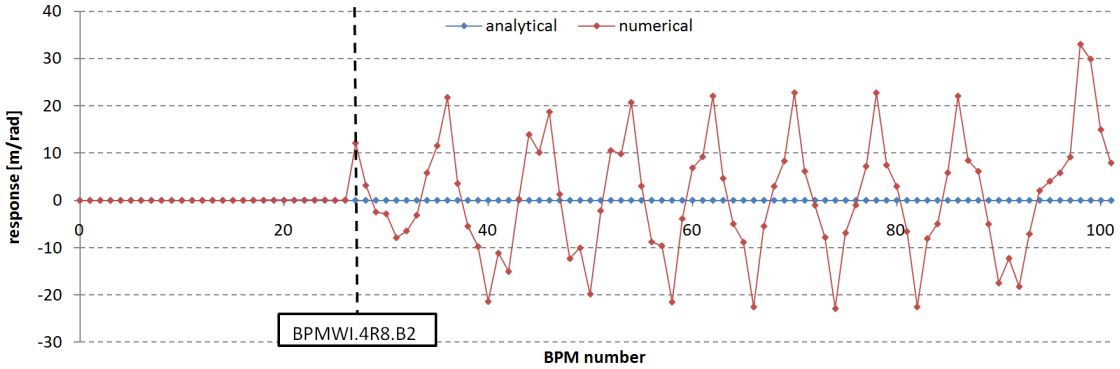
Figures 4.3 show the orbit-response for one corrector calculated by the both methods described above. The corrector used in the example is a horizontal one in the TI8 transfer line. The plots show the TI8 transfer line and the adjacent sector 78 of the LHC. It is visible that the calculation methods agree nicely in the plane of the corrector (Fig. 4.3(a)), while the out-of-plane result is totally wrong if coupling is involved as it is the case at the TI8-LHC junction.

The difference of the two calculation methods is plotted in Fig. 4.4 (analytical minus the numerical calculation). The rms of the error of the analytical model is 0.4 m/rad

[2] As *twiss run* we understand the calculation procedure of the Courant-Snyder functions and the beam positions by a numerical optics code (MadX in our case [HS04]).



(a) horizontal response (in-plane)



(b) vertical response (out-of-plane)

Figure 4.3.: Comparison of response for one horizontal corrector in TI8+LHC sector 78, calculated from Eq. (4.6) ('analytical') and numerically using MadX (Note the different scales of the vertical axes: The peak of the vertical response is about 20% of peak of the horizontal response). The marker 'BPMWI.4R8.B2' indicates the first BPM in the LHC.

for the in-plane result (only taking into account values downstream of the corrector), which is about 0.6 % of the rms of the positions (70.8 m/rad). The out-of-plane error relative to the numerical model is clearly 100 %, because coupling is not taken into account at all. Therefore, in the presence of coupling, the numerical calculation of the response matrix is the preferred method.

4.2.3. Pros and Cons

Compared to other optics measurement methods, the use of the kick response method has the following disadvantages:

- The measurement is not independent of the monitor and corrector gains. There-

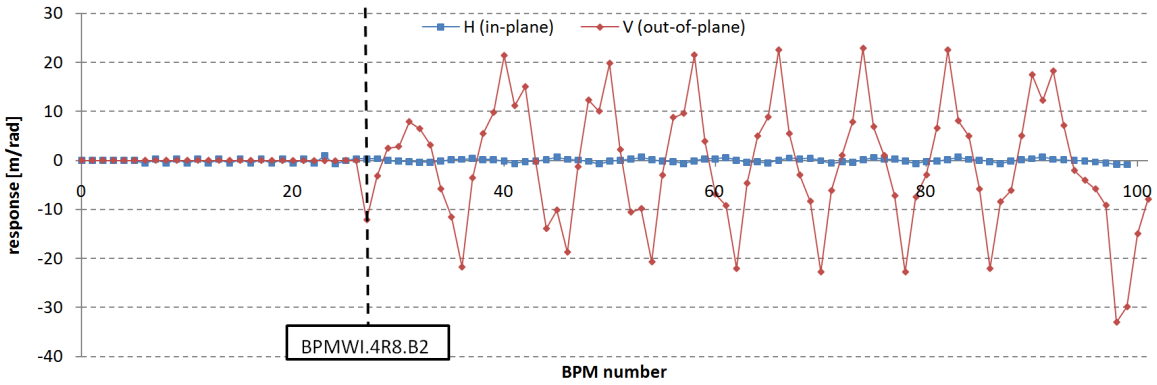


Figure 4.4.: Difference between analytically and numerically calculated response. The marker 'BPMWI.4R8.B2' indicates the first BPM in the LHC.

fore these contribute to errors, if not explicitly taken into account.

- In a ring, it is slow compared to multi turn measurements, because the used correctors have to be ramped one by one.

and the following advantages:

- Compared to multi turn measurements, kick response measurements do not blow up the emittance, since the the orbit changes are very slow (adiabatic) and so the whole closed orbit is moved.

- Since the measurement of the response matrix depends on corrector and monitor gains, it is possible to derive those values from the measured data.

- Because kick response measurements use the closed orbit (in a circular machine), this results in a higher precision of position measurement because of the closed orbit averaging compared to multi turn-measurements which is dependent on a turn-by-turn position measurement.

- Kick response measurements can be easily used for single pass beam lines, since no multi turn data is required.

The most important of these advantages is in our case the last point in this list, since therefore it was easy to apply this measurement technique to the injection lines of the LHC.

4.3. Dispersion

The dispersion function $D_u(s)$ defines the local sensivy of a beam trajectory or orbit $u(s)$ to a relative energy error $\frac{\Delta p}{p}$ as indicated by Eq. (3.9). Based on its definition, the dispersion can be obtained from beam position measurements (for both closed orbit and trajectory) performed for different values of $\frac{\Delta p}{p}$. The dispersion is then simply given by the slope of the position change w.r.t. to $\frac{\Delta p}{p}$. In a ring, the simplest way to induce an energy change is to change the RF frequency f. The resulting energy change is then given by

$$\frac{\Delta p}{p} = \frac{\frac{\Delta f}{f}}{\eta}, \tag{4.8}$$

with

$$\eta = \frac{1}{\gamma_r^2} - \alpha_C. \tag{4.9}$$

f is the RF frequency, γ_r the relativistic gamma and α_C the momentum compaction factor of the ring.

For this kind of measurements an half-automated procedure exists in the steering tool YASP: The momentum trims have to be done manually. After YASP has acquires a set of trajectories, the linear dispersion can be calculated from the data. The values for the momentum change can either be estimated by YASP or entered manually as trimmed before. We will discuss these issues in more detail in Sec. 8.4.

For a transfer line, obviously the frequency of the upstream accelerator has to be modified. So when considering the LHC injection lines, the energy in the SPS has to be modified. The values for the relevant parameters for the two accelerators are summarized in Table 4.1.

parameter	unit	SPS	LHC
α_C	[1]	$1.93{\cdot}10^{-3}$	$3.225{\cdot}10^{-4}$
f	[Hz]	200 394 400	400 788 860
γ_r (LHC injection)	[1]	479.6	

Table 4.1.: Parameters relevant for dispersion calculation in SPS and LHC.

5. The LHC Injection Transfer Lines

5.1. Introduction

In this chapter, we shortly summarize the main purpose and features of the two transfer lines between the SPS and the LHC. A more detailed discussion of this topic can be found for example in [BCM$^+$04, pp. 189-262]. We also outline the measurements which were done during the SPS-extraction- and LHC-injection-tests in 2008 and 2009 and describe the issues encountered, which provide the main motivation for the current thesis.

5.2. Overview

The two transfer lines, which transport protons at 450 GeV and ions from the SPS to the LHC, are denoted TI2 and TI8 (french: "*Tunnel d'Injection*"). Figure 5.1 shows their overall layout.

The TI2 line branches off from the SPS extraction line TT60[1] (french: "*Tunnel de Transfert*") which starts in the SPS long straight section LSS6[2] and ends in the LHC, ring 1, before LHC point 2, where the ALICE experiment is located (*Interaction Point 2*; IP2). The transfer line TI8 branches off from the SPS extraction line TT40[3] coming from LSS4 in the SPS and joins LHC, ring 2 some 180 m right of LHC point 8, where the LHCb experiment is located (*Interaction Point 8*; IP8).

Table 5.1 summarizes the main properties of the transfer lines. The values in this table are mainly taken from [BCM$^+$04, p. 194]. While TI2 is longer than TI8, many observed effects, as described later, are more significant in TI8. This is because of the larger horizontal bending angle and therefore more than the double amount of bending magnets (MBIs) located in TI8.

Figure 5.2 shows the vertical profile of the TI2 transfer line and Fig. 5.3 of TI8.

[1] The other branch of TT60 lead to the former *West Area* (experimental area) and will soon take SPS beam to the new material test facility *HiRadMat*.

[2] The SPS has a sixfold symmetry: It consists of six arcs of 1028 m length each and six *Long Straight Sections* (LSS1 to LSS6) in between of 128 m length each.

[3] The other branch of TT40 takes SPS beam to the CNGS (*Cern Neutrinos to Gran Sasso*) target.

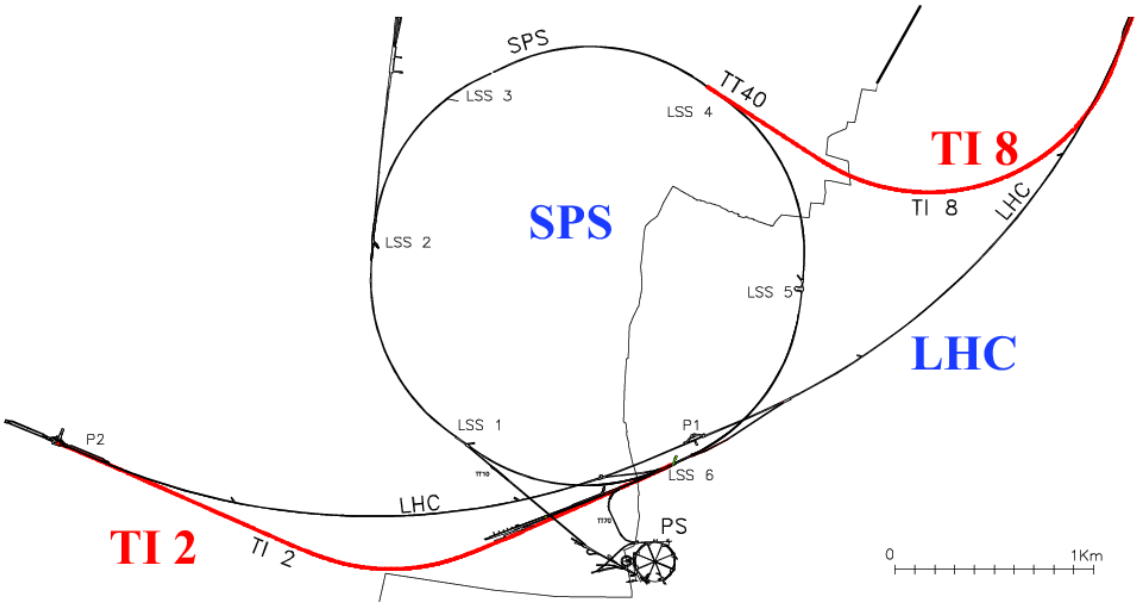


Figure 5.1.: Overall layout of the SPS to LHC transfer lines TI2 and TI8 (taken from [UAC+05]).

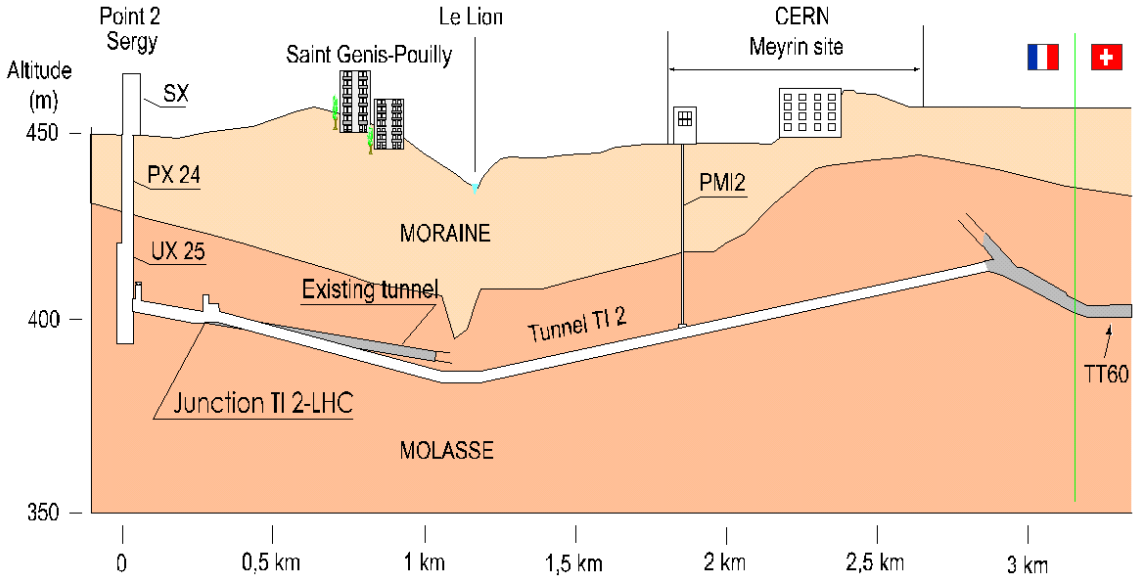


Figure 5.2.: Vertical profile of TI2 (taken from [UAC+05]).

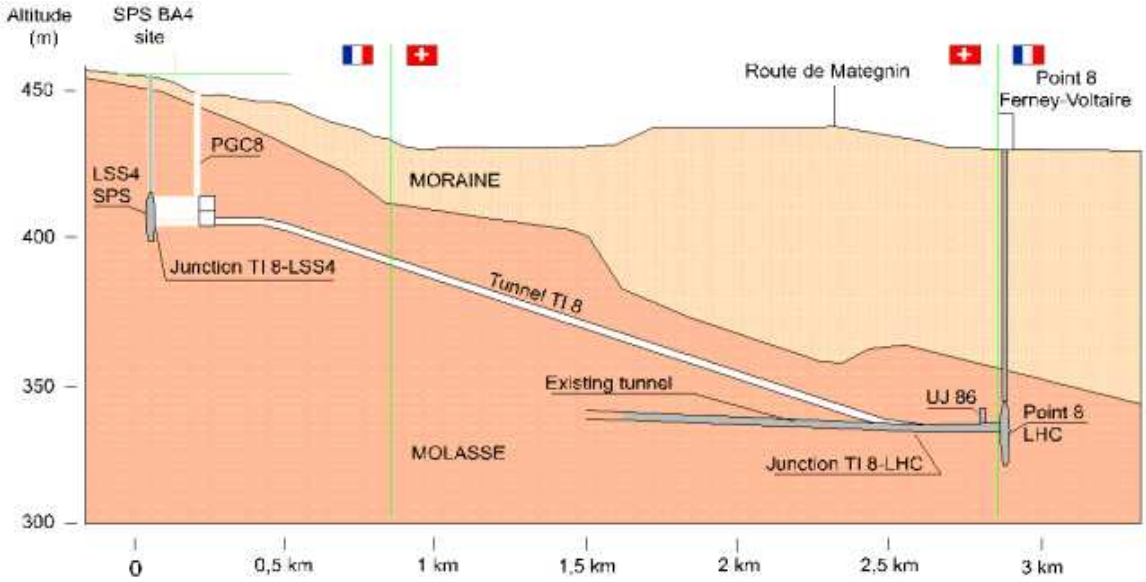


Figure 5.3.: Vertical profile of TI8 (taken from [UAA+08]).

property	unit	TI2	TI8
length	[m]	3116.732	2627.501
vertical height difference	[m]	1.124	70.875
horizontal bending angle	[°]	48.22	103.04
maximum vertical slope	[°]	4.3	3.8
number of MBIs		112	228

Table 5.1.: Key properties of the LHC injection transfer lines.

5.2.1. Optics

Both transfer lines use a FODO[4] lattice with 90 ° phase advance per cell and a half-cell length of 30.3 m. In TI8 each half-cell contains four dipoles, while in TI2 the numbers of dipoles per cell vary between zero and four. Short straight sections with space for instrumentation and dipole corrector magnets follow each quadrupole. The layout of an half-cell as well as of a short straight section is shown in Fig. A.1. The characteristic optics functions (beta-function and dispersion) are plotted in Fig. 5.4

[4] A FODO cell describes a periodic structure of transverse focussing accelerator elements. Focussing (F) and defocussing (D) quadrupole magnets are arranged alternatingly, such that a net transverse focussing is achieved [Hin08, p. 63].

for TI2 and in Fig. 5.5 for TI8. Table 5.2 summarizes some of the characteristical optics values for both lines.

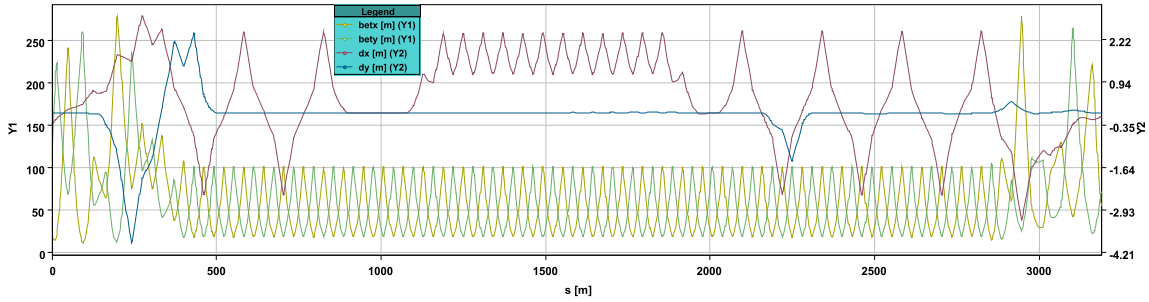


Figure 5.4.: Optics functions of the TI2 transfer line.

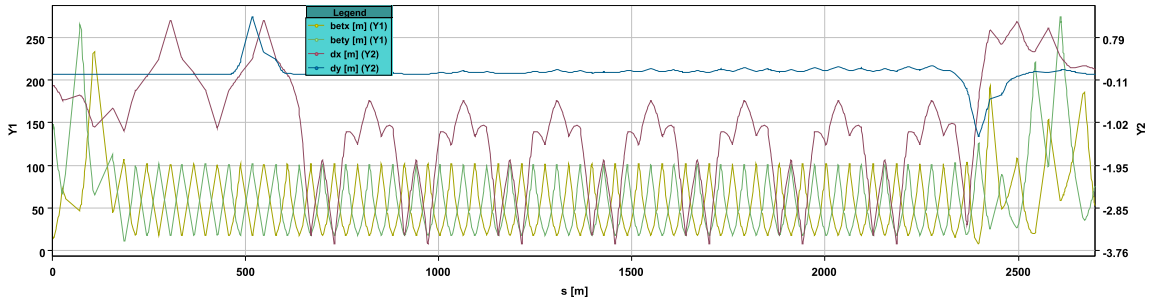


Figure 5.5.: Optics functions of the TI8 transfer line.

5.2.2. Distribution of Beam Position Monitors

To save costs, the transfer lines were not equipped with a *Beam Position Monitor* (BPM) at every quadrupole which would give a sampling of eight BPMs per betatron oscillation, like in the LHC. Instead, in the original version, only two BPMs per oscillation were installed, namely at two neighboring focusing quadrupoles of the respective plane. This is the configuration for TI2 and is sketched in Fig. 5.6(a). Although this covers oscillations of all possible phases at least with one BPM per oscillation, the sampling is very asymmetric, as illustrated in Fig. 5.6(b). This makes it difficult to distinguish between orbit contributions resulting from betatron oscillations and dispersive effects.

For TI8, the situation was improved during the shutdown in 2008/2009. During this period, all the installed BPMs were upgraded to measure both planes. Although

parameter	unit	TI2	TI8
β_x max	[m]	308.5	240.8
β_y max	[m]	289.5	274.6
β_x max (arc section)	[m]	101.2	101.2
β_y max (arc section)	[m]	101.1	101.1
$\lvert D_x \rvert$ max	[m]	3.82	3.88
$\lvert D_y \rvert$ max	[m]	3.97	1.34
D_x rms	[m]	1.42	1.78
D_y rms	[m]	0.55	0.20
μ_x total	$[2\pi]$	12.07	10.54
μ_y total	$[2\pi]$	12.24	10.32
Half-cell length	[m]	30.3	
Number of half-cells		95	85

Table 5.2.: Characteristic optics parameters of TI2 and TI8.

this doubles the number of BPMs per plane for this transfer line, the newly installed BPMs sit at locations with small beta functions for the plane (see Fig. 5.7(a)) and the sampling of an oscillation is still not symmetric as illustrated in Fig. 5.7(b).

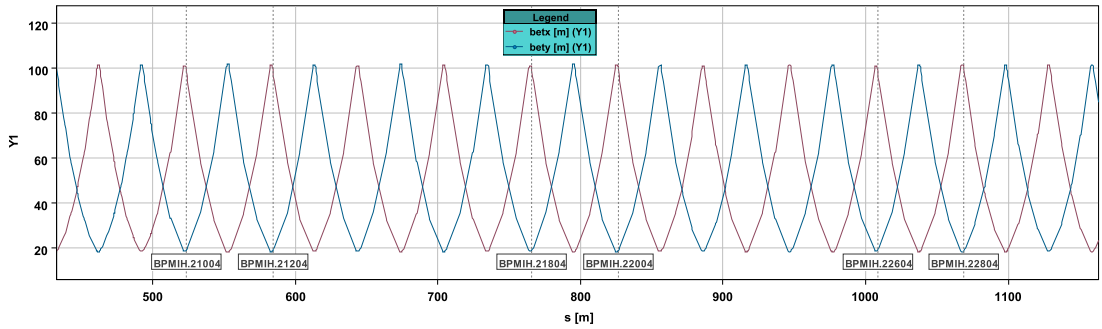
During the shutdown in 2010/2011 the same improvement was also implemented in TI2. Nevertheless, all the TI2 data used in this thesis was taken before this improvement and thus have only the BPM sampling as explained before.

5.2.3. The Injection Main Bending Magnets

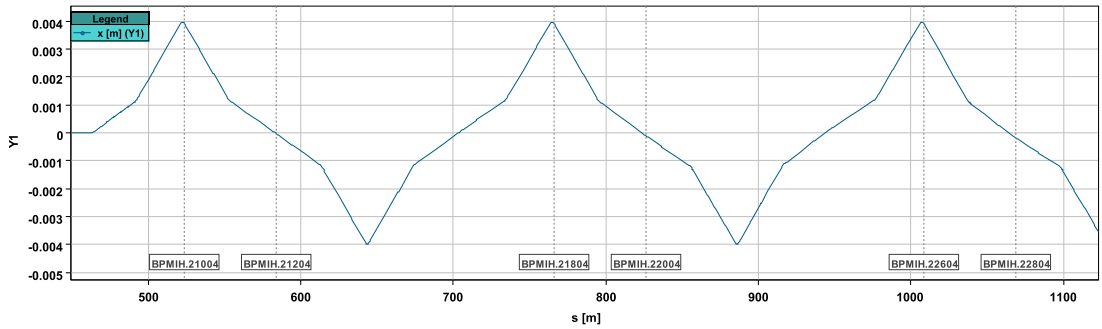
The *Main Bending* magnets of the *Injection* lines (MBIs) will play a key role in our future analysis (see Chap. 8). They are normal conducting magnets with a nominal current of 5270 A and a resulting field of 1.81 T. Their magnetic length is 6.33 m and their total length is 6.7 m. Figure 5.8 shows a picture of such an MBI. Schematics are included in the appendix (Fig. A.2).

5.3. Measured inconsistencies

Beam commissioning started in 2004 for TI8 [UAC+05] and in 2008 for TI2 [UAA+08]. During this period, detailed optics measurements were performed. Hereby three main inconsistencies were discovered. Since these inconsistencies were the main motivations for the investigations described in this theseis, they are introduced in the following.



(a) BPM distribution.



(b) Sampling of example oscillation.

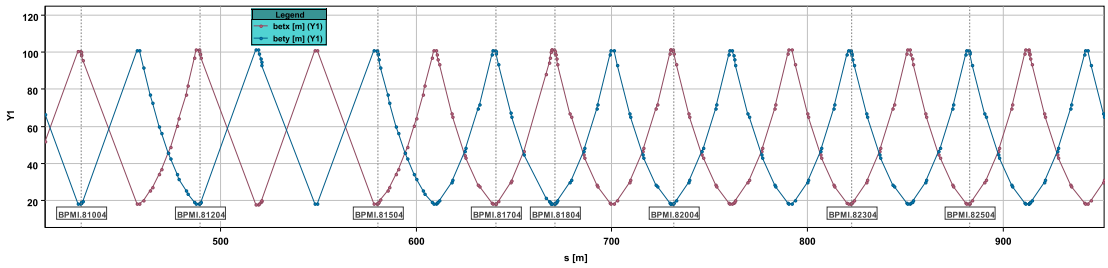
Figure 5.6.: BPM distribution and oscillation sampling in TI2.

5.3.1. Phase Advance Error

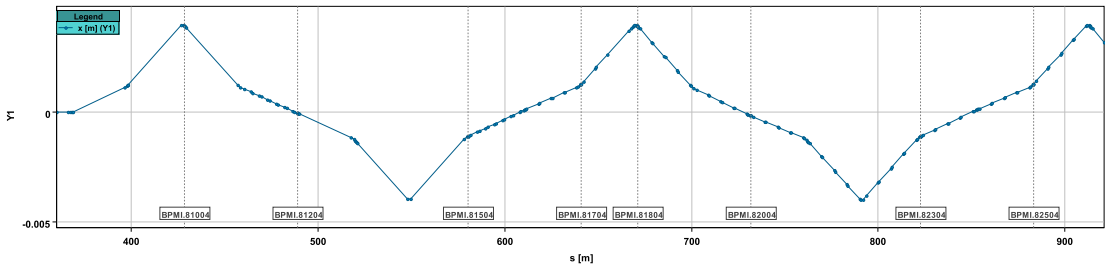
Kick response measurements in 2006 and 2008 revealed a visible phase error in the vertical plane in the transfer lines. Examples of such orbit responses[5] are shown in Figs. 5.9. While the horizontal response is in good agreement with the model, a phase error visibly adds up throughout the line for the vertical plane (Fig. 5.9(b)).

By fitting the strengths of the two main quadrupole families, a relatively large detuning (6.5 permill) of the main vertical focusing quadrupole strength was found while the horizontal focusing quadrupole strength was in good consistency with the model (error of about 0.6 permill) for TI8 [Wen06]. The results for TI2 were similar for the vertical plane (7.6 permill detuning) but worse than for TI8 for the horizontal plane (3.5 to 6.5 permill) [Wen08].

[5]Data taken: 2008-05-24



(a) BPM distribution.



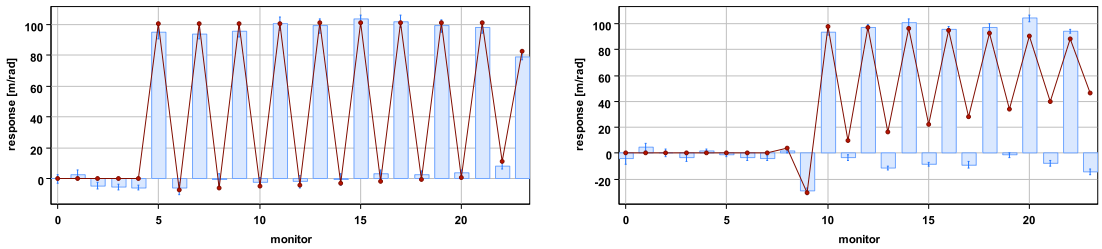
(b) Sampling of example oscillation.

Figure 5.7.: BPM distribution and oscillation sampling in TI8.

Figure 5.8.: Photo of an LHC injection main bend (MBI).

5.3.2. Dispersion Mismatch

Later, during the injection tests into the LHC in 2008 [AAA+08], a strong dispersion mismatch with the onset around the junction between TI 8 and the LHC was



(a) Horizontal response (corrector MCIAH.80204) (b) Vertical response (corrector MCIAV.81304)

Figure 5.9.: Example responses to demonstrate phase error in vertical plane of TI8. Each blue bars represents a measured trajectory response at one BPM. Red dots represent the respective values, calculated from the nominal model.

observed, as shown in Fig. 5.10.

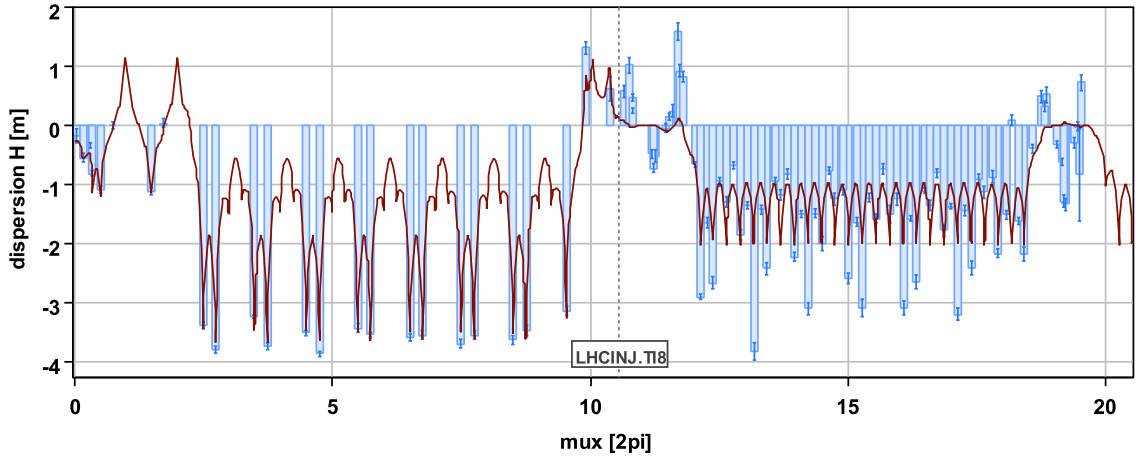
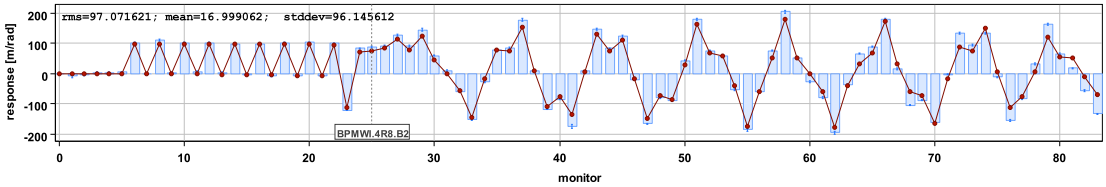


Figure 5.10.: Horizontal dispersion of TI 8 and LHC sector 78. Bars represent the measured dispersion and the line respresent the dispersion calculated from the nominal model. The marker 'LHCINJ.TI8' indicates the junction between the transfer line and the LHC.

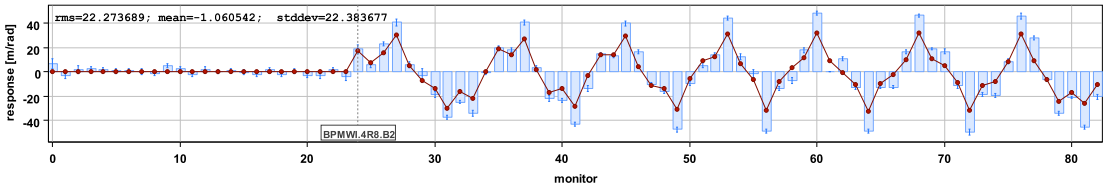
5.3.3. Coupling at the LHC Injection Points

The transverse coupling at the junctions between the transfer lines and the LHC was originally estimated to be lower than 5 %, which would result in an emittance growth

of only 0.1 % and could therefore be neglected [BCM+04, p. 189]. Nevertheless, the coupling which was observed by the help of kick response measurements during the 2008 injection tests, turned out to be up to 20 % for TI8, and thus was much larger than expected originally [GFK+09]. Figures 5.11 show an example of such response measurements. A horizontal kick was applied and the horizontal response is shown in Fig. 5.11(a). It is visible that a vertical oscillation (Fig. 5.11(b)) starts at the TI8-LHC junction (BPMWI.4R8.B2 is the first BPM in the LHC in these plots). This issue was further investigated and relatively well understood, although the measurements still indicate a slightly larger coupling of a few percent w.r.t. the model, for which the reason is still not totally clear [KFG+09].



(a) Horizontal response for horizontal corrector.



(b) Vertical response for horizontal corrector.

Figure 5.11.: Transverse coupling from the horizontal into the vertical plane at the TI8-LHC junction. The pictures show the responses of the horizontal corrector MCIAH.80804. Blue bars represent the measured data, red lines the model. The marker 'BPMWI.4R8.B2' indicates the first BPM in the LHC. Note the different scales: The maximum of the vertical response is about 20% of the horizontal response.

6. Analysis Principle

6.1. Introduction

In this chapter, we will shortly summarize the numerical methods used to understand the problems described in Chap. 5. For kick response data, this principle is well known and often referred to as the LOCO[1]-principle [Saf97]. Although we will use the method mainly for kick-response measurements, we will describe it here in a more general form. This will be useful later when we will describe the implementational details of the used software, which can handle different kinds of data (see Sec. 7.2).

6.2. General Principle

Any observable that can be measured in the machine and computed from a numerical model can be used to fit the numerical model to the measurement. We will denote such a set of observables as elements of a vector $\vec{K}$ in the following:

$$\vec{K} = \begin{pmatrix} K_1 \\ K_2 \\ \cdots \\ K_{N_K} \end{pmatrix}. \tag{6.1}$$

N_K denotes the number of available observable-values K_i. As concrete values for a K_i one could think for example of an element of the kick-response matrix or a dispersion value at a certain monitor as will be described in Sec. 6.5.1 and Sec. 6.5.2. To describe the observable difference between the measured values ($\vec{K}^{\mathrm{meas}}$) and the calculated values ($\vec{K}^{\mathrm{model}}$) we define the difference vector $\vec{V}$ as

$$\vec{V} = \vec{K}^{\mathrm{diff}} = \vec{K}^{\mathrm{meas}} - \vec{K}^{\mathrm{model}}. \tag{6.2}$$

The norm of this vector

$$\left\| \vec{V} \right\| = \sqrt{\sum_k |V_k|^2} \tag{6.3}$$

[1]LOCO is the abbreviation for '*Linear Optics from Closed Orbits*'.

represents the error of the model relative to the measurement. The goal of the fit is to minimize this norm and therefore the difference between the measured and modeled observables by adjusting N_f parameters of the model:

$$\left\|\vec{V}\right\|^2 = \text{minimum}. \tag{6.4}$$

To prepare for the fit N_f parameters have to be selected, which shall be denoted c_l and again can be seen as elements of a vector $\vec{c}$:

$$\vec{c} = \begin{pmatrix} c_1 \\ c_2 \\ \dots \\ c_{N_f} \end{pmatrix}. \tag{6.5}$$

To calculate an adjusted set of parameters $\vec{c}\,'$ the problem is linearized. Therefore, a knowledge of the dependency of the difference vector on each of the parameters is required. This feature is provided by the Jacobi Matrix J which in this case would be defined as

$$J_{kl} = \frac{\partial V_k}{\partial c_l}. \tag{6.6}$$

The resulting linearized problem for the parameter changes $\Delta\vec{c}$ then reads

$$\vec{V} + J\Delta\vec{c} = 0. \tag{6.7}$$

The fitting algorithm has to solve this equation for $\Delta\vec{c}$. Then a new set of parameters $\vec{c}\,'$ is calculated by

$$\boxed{\vec{c}\,' = \vec{c} + \Delta\vec{c}.} \tag{6.8}$$

The procedure is then iterated by updating the model with the new parameter values, calculating a new $\vec{V}$ and J and solving again Eq. (6.7). The iteration then can be continued until a stable solution is found, which corresponds to $\Delta\vec{c} \cong 0$.

In order to unify the process, it is useful to define a slightly different matrix S as defined by

$$S^{kl} = -J_{kl} = -\frac{\partial V_k}{\partial c_l}, \tag{6.9}$$

which we will call the sensity matrix. By inserting this definition into Eq. (6.7) finally we are left with a standard system of linear equations

$$S\Delta\vec{c} = \vec{V}. \tag{6.10}$$

It may be noted that of course both, the difference vector $\vec{V}$ and the sensitivity matrix S in general depend on the actual values of the parameters c_l, they should be

denoted as

$$\vec{V} = \vec{V}(\vec{c}), \qquad S = S(\vec{c}). \tag{6.11}$$

but for the sake of simplicity in general we stick to the short notations $\vec{V}$ and S.

Finally, it shall be noted that we consistently define all parameters as acting on the model. So the parameter values resulting from the fits are values, which have to be applied to the model in order to match the measured and calculated data (e.g. kick-response) as closely as possible. So only the model-part of the difference vector depends on the parameters:

$$\vec{V}(\vec{c}) = \vec{K}^{\text{meas}} - \vec{K}^{\text{model}}(\vec{c}). \tag{6.12}$$

This fact, together with the sign convention of Eq. (6.9), results in the small simplification that the sensitivity-matrix elements are simply given by the partial derivatives of the model- observable vector,

$$\begin{aligned} S^{kl} &= -\frac{\partial V_k}{\partial c_l} = -\left(-\frac{\partial K_k^{\text{model}}}{\partial c_l}\right) \\ &= \frac{\partial K_k^{\text{model}}}{\partial c_l}. \end{aligned} \tag{6.13}$$

6.3. Algorithms

Eq. (6.10) may be solved for $\Delta\vec{c}$ by the use of various linear equation solvers. In this work we used the following two for different purposes:

- *Singular Value Decomposition* (SVD): This algorithm finds the best least square solution, using all the available parameters. A detailed discussion of this algorithm can e.g. found in [PFTV07].
- MICADO (*MInimisation des CArrés des Distortions d'Orbite*) is an algorithm which was originally used for orbit correction [AM73]. It uses a fixed number of the most effective parameters to find a least square solution. So if e.g. an optics error is assumed to be localized, this algorithm with few (or even only one) most effective parameter(s) can be used, to 'pinpoint' the error.

6.4. Parameters

We distinguish two different classes of parameters:

- parameters, that are linear in some elements of $\vec{V}$ (e.g. corrector- and monitor-calibration factors (gains).

- all other (arbitrary) model parameters.

The differences in handling of these two classes of parameters is described in the following sections.

6.4.1. Parameters Linear in V_k

Sensitivity matrix elements for parameters that are linear in elements of $\vec{V}$ are calculated in a slightly different way than for other parameters (as well as in the former LOCO code). Therefore, also the update procedure for the fitted parameters (Eq. (6.8)) has to be slightly modified.

In order to illustrate this procedure, we will look at the model component of the difference vector and explicitely write it as a product of a parameter and an unscaled observable:

$$K_k^{\text{model}} = c_m \tilde{K}_k^{\text{model}}. \tag{6.14}$$

By using this to calculate the sensitivity matrix according to Eq. (6.13) one obtains

$$S^{kl} = \frac{\partial}{\partial c_l}\left(c_m \tilde{K}_k^{\text{model}}\right) = \tilde{K}_k^{\text{model}} \delta_{ml} = \frac{K_k^{\text{model}}}{c_m}\delta_{ml}. \tag{6.15}$$

δ_{ml} is the Kronecker-delta, which is defined as

$$\delta_{ml} = \begin{cases} 1 & \text{if } m = l, \\ 0 & \text{otherwise.} \end{cases} \tag{6.16}$$

Instead of using the values obtained in Eq. (6.15) as elements for the sensitivity matrix, we use the unchanged entries of the model observable vector

$$\tilde{S}_{kl} = K_k^{\text{model}} \delta_{ml} \tag{6.17}$$

for $m \in L$, where L denotes the set of the indizes of all parameters on which K_k^{model} is linearly dependent. This procedure has the slight advantage of avoiding divisions by small values in Eq. (6.15) when parameters are getting close to zero. When new parameter values $c_l{}'$ are obtained from calculated parameter changes Δc_l one has to take care of this fact and calculate the new parameter values in a slightly different way. By comparing Eq. (6.15) and Eq. (6.17) we immediately get

$$S_{kl} = \frac{\tilde{S}_{kl}}{c_l}. \tag{6.18}$$

Using this relation in the linearized problem Eq. (6.10) in elements notation,

$$V_k - \sum_l S^{kl} \Delta c_l = 0, \quad \forall k, \tag{6.19}$$

results in

$$V_k - \sum_l \frac{\tilde{S}_{kl}}{c_l} \Delta c_l = V_k - \sum_l \tilde{S}_{kl} \Delta \tilde{c}_l = 0, \quad \forall k, \tag{6.20}$$

where the newly defined $\Delta \tilde{c}_l$ is related to the original parameter changes by

$$\Delta \tilde{c}_l = \frac{\Delta c_l}{c_l}. \tag{6.21}$$

Inserting this relation into Eq. (6.8) we get for parameters linear in V_k

$$\begin{aligned} c_l{}' &= c_l + c_l \, \Delta \tilde{c}_l \\ &= c_l \left(1 + \Delta \tilde{c}_l\right). \end{aligned} \tag{6.22}$$

This relation is used to determine the new values for this kind of parameters after a fit iteration.

6.4.2. Arbitrary Model Parameters

Since the developed software (Sec. 7.2) directly interacts with the numerical model for the machine under investigation (almost) all kinds of parameters of this model can be used as free parameters for the fits. For these parameters the entries in the sensitivity matrix have to be calculated using a linear approximation.

Therefore the model observable vector has to be computed for each parameter c_l for two different values c_l and $c_l + \delta c_l$. Then the linear approximation of the sensitivity is calculated by

$$S_{kl} = \frac{K_k^{\text{model}}(c_l + \delta c_l) - K_k^{\text{model}}(c_l)}{\delta c_l}. \tag{6.23}$$

This corresponds to a local fit gradient and the parameter increment δc_l has to be chosen carefully for each parameter. The new parameter values for these entries after a fit iteration are calculated in the standard way as given by Eq. (6.8):

$$c_l{}' = c_l + \Delta c_l. \tag{6.24}$$

6.5. Input Data

We will describe in detail the treatment of the type of data which we used during the analysis, i.e. kick response data and dispersion data. Nevertheless, one could imagine many other types of measurement data which could be treated in a similar way, like e.g. multiturn measurements. And indeed, some of these types are already supported by the developed software package (see Sec. 7.2).

6.5.1. Kick Response Data

This response matrix can be determined for both the measurement (R^{meas}) and the model (R^{model}). The difference between the measurement and the model can be expressed e.g. by a difference response matrix R^{diff} whose elements we define by

$$R_{ij}^{\text{diff}} = \frac{R_{ij}^{\text{meas}} - R_{ij}^{\text{model}}}{\sigma_i}. \tag{6.25}$$

Here σ_i represents the measurement noise of the ith monitor. This results in a smaller weighting for readings of noisy monitors and higher weighting of reading of monitors with lower noise. In case the noise is zero or unknown, or if we want to ignore the noise in our analysis, then we set $\sigma_i = 0$. The difference vector, according to Eq. (6.2) is then given by

$$V_k = R_{ij}^{\text{diff}}, \tag{6.26}$$

where the indices are mapped by the relation

$$k = i(N_C - 1) + j. \tag{6.27}$$

The model part of the difference vector is then given by

$$K_k^{\text{model}} = \frac{R_{ij}^{\text{model}}}{\sigma_i}, \tag{6.28}$$

which leads to sensitivity matrix elements for the parameters linear in V_k (the monitor and corrector gains in this case) of

$$\tilde{S}_{kl}^{K,M} = \frac{R_{ij}^{\text{model}}}{\sigma_i}\delta_{jl} \tag{6.29}$$

for the monitor part,

$$\tilde{S}_{kl}^{K,C} = \frac{R_{ij}^{\text{model}}}{\sigma_i}\delta_{il} \tag{6.30}$$

for the corrector part and of

$$S_{kl}^{K,P} = \frac{R_{ij}^{\text{model}}(c_l + \Delta c_l) - R_{ij}^{\text{model}}(c_l)}{\Delta c_l \, \sigma_i} =: \frac{\Delta R_{ij}^{\text{model}}(\Delta c_l)}{\sigma_i} \tag{6.31}$$

for the other parameters. Here the label K denotes the correspondence to the kick-response measurement, M to the monitor gains, C to the corrector gains and P to other arbitrary model parameters.

It may be noted, that the norm of the difference vector as defined in Eq. (6.3) in this case is equivalent to the Frobenius norm of the difference matrix, which is defined for a arbitrary matrix A with elements A_{ij} as

$$\|A\|_F = \sqrt{\sum_{i,j} |A_{ij}|^2}. \tag{6.32}$$

This is used internally by the analysis software (Sec. 7.2) since methods to calculate this norm are provided by standard packages.

Model Quality

To estimate the quality of the model we use the rms of the elements of the relative difference-matrix:

$$\Delta_{\text{rms}}^{K} = \sqrt{\frac{1}{N_v} \sum_{i,j} \left(\frac{R_{ij}^{\text{meas}} - R_{ij}^{\text{model}}}{\frac{\sigma_i}{\delta_j}} \right)^2}. \tag{6.33}$$

σ_i is the noise for monitor i, δ_j is the kick of corrector j and N_v is the number of all elements of the response-matrix. This value can be used e.g. for comparison of two different fitted models, but has no absolute meaning, since also invalid readings contribute to this value. Invalid readings are for example readings that come from monitors, which gave no or wrong readings for one or more measurement acquisitions. These contributions are ignored by the fits and set to zero in the difference matrix, and therefore reduce the result of Eq. (6.33). The unit of Δ_{rms}^{K} as given in Eq. (6.33) is [1]. Nevertheless, in some cases we do not normalize by the error. In these cases, the unit will then be $[\text{m}/\text{rad}]$.

6.5.2. Dispersion Data

We treat dispersion data in a similar way as kick-response data, as explained in Sec. 6.5.1. Instead of the response matrix, this time we use directly the dispersion values at each monitor as data that has to be compared between measurement and

model. These dispersion values can be seen as vector

$$\vec{D} = \begin{pmatrix} D_1 \\ D_2 \\ \dots \\ D_{N_M} \end{pmatrix}. \tag{6.34}$$

These values can be measured for the real machine ($\vec{D}^{\text{meas}}$) and calculated for the model ($\vec{D}^{\text{model}}$). We define the difference between the measurement and the model as

$$V_k = D_k^{\text{diff}} = \frac{D_k^{\text{meas}} - D_k^{\text{model}}}{\sigma_k}. \tag{6.35}$$

σ_k again denotes the measurement noise of the monitor k. The model part of the difference vector is given by

$$K_k^{\text{model}} = \frac{D_k^{\text{model}}}{\sigma_k}. \tag{6.36}$$

The only parameters we consider as linear in V_k this time are the monitor gains. For this part of the sensitivity matrix we get from Eq. (6.17)

$$\tilde{S}_{kl}^{D,M} = \frac{D_k^{\text{model}}}{\sigma_k}\,\delta_{kl}. \tag{6.37}$$

δ_{kl} is meant to be the Kronecker-delta, expressing that this part of the sensitivity matrix is a diagonal matrix.

In all our dispersion analysis, we use numerical models with all orbit correctors set to zero and therefore ignore the generation of dispersion from orbit correctors. Thus, we consider corrector gains as not contributing to the dispersion. Of course this is only approximately true. In a real machine they would contribute to a certain degree, if there is a trim in the machine. Nevertheless, to model this correctly, one would need the actually trimmed values together with a correct alignment model. Both of these informations are not available in general. So for the dispersion-part we set the corrector-gain part of the sensitivity matrix to

$$\tilde{S}_{kl}^{D,C} = 0. \tag{6.38}$$

The matrix elements corresponding to the remaining fit parameters are again calculated via the linear approximation Eq. (6.23), which reads for dispersion data

$$S_{kl}^{D,P} = \frac{D_k^{\text{model}}(c_l + \Delta c_l) - D_k^{\text{model}}(c_l)}{\Delta c_l\,\sigma_k} =: \frac{\Delta D_k^{\text{model}}(\Delta c_l)}{\sigma_k}. \tag{6.39}$$

Model Quality

To estimate the quality of a dispersion model we define

$$\Delta^D_{\mathrm{rms}} = \sqrt{\frac{1}{N^D_v}\sum_i \left(\frac{D_i^{\mathrm{meas}} - D_i^{\mathrm{model}}}{\sigma_i}\right)^2}. \tag{6.40}$$

Here the D_i are the (measured and calculated) dispersion values at the monitor i, N^D_v is the number of dispersion values D_i and σ_i is the noise of the monitor i. The unit of Δ^D_{rms} as given in Eq. (6.40) is [1]. In cases, when we do not normalize by the monitor errors, the unit will be [m].

6.5.3. Normalization and Weighting of different Measurements and Parameter Responses.

Now that the construction of the sensitivity matrices for various data has been shown in the previous sections we have to combine all these matrices into one big sensitivity matrix in order to use it as input for a combined fit. Besides combining the matrices, there are some other aspects, which we have to consider:

1. Different model parameters c_l may in general have effects of different magnitude on the model compared to a change in a monitor- or corrector-gain. This would result in eigenvalues of different magnitude when solving for the parameter change vector and too small eigenvalues therefore would be ignored, because the algorithm ignores eigenvalues below a certain (configurable) threshold (default in the used software: below 2.5% of the highest eigenvalue). Normalization factors for the parts of the sensitivity matrices corresponding to arbitrary model parameters (columns of the sensitivity matrix calculated by Eq. (6.31) and Eq. (6.39)) are introduced in order to deal with this fact. These factors are denoted by $\omega^{\mathrm{auto}}_{P_l}$ in the following. Note that the label "auto" indicates, that these factors are calculated automatically.

2. The different measurements (different blocks of rows in the sensitivity matrix) have different physical units (e.g. [m] for the dispersion and [m/rad] for the kick-response) and in general also different magnitude. This is handled by adding additional normalization factors ω^{auto}_k to these blocks of lines. These factors are determined automatically (as again indicated by the label "auto") so that the different measurements are weighted equally.

3. On top of this, additional weighting factors ω^{man}_k for each measurement are introduced that are by default set to 1 and can be adjusted manually. This

allows fine-tuning of the fit behavior. If, for example a dispersion measurement is weighted higher than a kick-response measurement, then the fit might potentially find a better correction for the dispersion at the expense of the kick-response fit-quality. The label "man" here denotes, that these parameters are to be set manually.

Putting all this together, we end up with the total sensitivity matrix containing different sub-matrices and weighting factors.

$$
\begin{aligned}
S = &\begin{pmatrix}
\omega_1^{\text{man}}\,\omega_1^{\text{auto}} & \dots & 0 & \dots & 0 \\
\dots & \dots & \dots & \dots & \dots \\
0 & \dots & \omega_k^{\text{man}}\,\omega_k^{\text{auto}} & \dots & 0 \\
\dots & \dots & \dots & \dots & \dots \\
0 & \dots & 0 & \dots & \omega_{N_K}^{\text{man}}\,\omega_{N_K}^{\text{auto}}
\end{pmatrix} \cdot \\
&\cdot \begin{pmatrix}
\tilde{S}_1^M & \tilde{S}_1^C & S_1^{P_1} & \dots & S_1^{P_{N_P}} \\
\dots & \dots & \dots & \dots & \dots \\
\tilde{S}_k^M & \tilde{S}_k^C & S_k^{P_1} & \dots & S_k^{P_{N_P}} \\
\dots & \dots & \dots & \dots & \dots \\
\tilde{S}_{N_K}^M & \tilde{S}_{N_K}^C & S_{N_K}^{P_1} & \dots & S_{N_K}^{P_{N_P}}
\end{pmatrix} \cdot \\
&\cdot \begin{pmatrix}
1 & 0 & 0 & \dots & 0 \\
0 & 1 & 0 & \dots & 0 \\
0 & 0 & \omega_{P_1}^{\text{auto}} & \dots & 0 \\
\dots & \dots & \dots & \dots & \dots \\
0 & 0 & 0 & \dots & \omega_{P_{N_P}}^{\text{auto}}
\end{pmatrix}
\end{aligned}
\tag{6.41}
$$

Note that in this equation e.g. $S_k^{P_l}$ denotes the column l of a sensitivity matrix for an arbitrary model parameter calculated e.g. by Eq. (6.31) for a kick-response measurement.

The weighting factors for the different parameter columns, $\omega_{P_l}^{\text{auto}}$ have to be the same for all data-contributions (e.g. kick-response and dispersion). How they are calculated depends on which data combination is used for fitting. We always calculate them from the first contribution and apply them also to the other contributions if there are some.

Column Factors from Kick Response Data

For calculating these factors from kick response data, we use the same method as the LOCO code: The columns are normalized to the unperturbed response matrix,

$$\omega_{P_l}^{\text{auto}} = \frac{\left\|R^{\text{model}}\right\|_F}{\left\|\Delta R^{\text{model}}(\Delta c_l)\right\|_F}, \tag{6.42}$$

where $\Delta R^{\text{model}}(c_l)$ is the difference between the original model response matrix and the one, where parameter l is varied, as defined by Eq. (6.31).

Column Factors from Dispersion Data

For calculating these factors from the dispersion data, a similar procedure is used: The columns are normalized to the unperturbed dispersion vector by

$$\omega_{P_l}^{\text{auto}} = \frac{\left\|D^{\text{model}}\right\|_F}{\left\|\Delta D^{\text{model}}(\Delta c_l)\right\|_F}, \tag{6.43}$$

where the $\Delta D^{\text{model}}(c_l)$ is the difference between perturbed and unperturbed model dispersion values, as used in Eq. (6.39).

Row Factors

As mentioned before the weighting factors ω_k^{man} can be set manually. In order to avoid manual treatment most of the time, the additional automatic factors ω_k^{auto} are applied. This ensures that the contributions of the different measurements are weighted equally. If we rewrite Eq. (6.41) by combining the second and the third matrix,

$$S = \begin{pmatrix} \omega_1^{\text{man}}\,\omega_1^{\text{auto}} & \dots & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & \omega_k^{\text{man}}\,\omega_k^{\text{auto}} & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & \dots & \omega_{N_K}^{\text{man}}\,\omega_{N_K}^{\text{auto}} \end{pmatrix} \begin{pmatrix} S_1 \\ \dots \\ S_k \\ \dots \\ S_{N_K} \end{pmatrix}, \tag{6.44}$$

a convenient choice is to set

$$\omega_1^{\text{auto}} = 1 \tag{6.45}$$

and (in analogy to the weighting factors for the columns)

$$\omega_k^{\text{auto}} = \frac{\|S_1\|_F}{\|S_k\|_F}. \tag{6.46}$$

The row-sections of the difference vector V are multiplied by the same factors, so that in the end, the resulting parameter changes are calculated correctly from the matrix equation Eq. (6.10).

6.5.4. Measurements at Different Machine Conditions

Sometimes it is desirable or necessary to use measurements which were taken at different machine conditions. One reason could be that some machine parameter was changed (not on purpose) between two measurements. Sometimes this might be done on purpose in order to gain additional constraints to the fits.

The later principle was extensively used to gain the results explained in Chap. 8, where kick-response measurements were taken at different values of $\frac{\Delta p_0}{p}$ (energy mismatch between the SPS and the transfer line). By the use of the software described in Sec. 7.2 these measurements can still be fitted alltogether. Care has to be taken that different models (e.g. with the corresponding $\frac{\Delta p_0}{p}$ values) have to be used for creating the sensitivity submatrices (Eq. (6.23)) and the parameters of the models have to be varied in a consistent way (Eq. (6.8)). For more details see Sec. 7.2.

7. Computational Tools

7.1. Introduction

In this chapter, we describe the computational tools which were developed and used for the data analysis in this thesis. In Sec. 7.2 Aloha (*Another Linear Optics Helper Application*), a Java reimplementation of the LOCO response fitting code [Saf97] with many extensions, is introduced. Then in Sec. 7.3, the newly developed Java API[1] for MadX (*Methodical Accelerator Design*) [HS04], which is called JMad, is described. This API is intensively used by Aloha.

7.2. Aloha

7.2.1. Motivation and Design Considerations

Figure 7.1.: The Aloha logo

Traditionally, kick-response analysis was done using the LOCO-code [Saf97] together with some extensions and shell scripts to process the output and create the required input files. Due to the script-based nature, a detailed knowledge of the package and the scripts was necessary to apply this method to different accelerators and problems. For the startup of the LHC, especially in view of the inconsistencies described in Sec. 5.3, a more flexible and interactive solution was required, to be able to analyze data quickly and online in the control room. Therefore, the development of a new software package, Aloha,

[1] Application Programming Interface.

was launched. Parts of the explanations in this section were already published in [Fuc09].

Requirements

The main design requirements for Aloha were:

- **Interactivity:** The analysis of lattice measurements is in many cases a trial and error process, which requires the selection of different parameter-sets for fitting and comparing the results. For this reason a *Graphical User Interface* (GUI) is required, which provides the user with the available parameters and immediately displays the corresponding results.
- **Integration with MadX:** All the models for the accelerators at CERN are available as input files for the MadX software package. Since these models define the available parameters and provide the nominal optics functions it is a strong requirement to have a close integration to MadX, in order to interact directly from the fitting routine.
- **Integration into the LHC control system:** To have a tool which would be available online in the control room, it is necessary to integrate this new tool in the control room environment. This integration provides the possibility of importing data from various systems as well as direct usage of the software within other applications.
- **Generalization:** While LOCO is designed for kick response analysis only, the newly developed software must be able to handle various type of input data (e.g. kick response, dispersion, multi turn data ...) and fit them together in one go. This way, an immediate cross-check with other optics parameters is possible.
- **Extensibility:** It must be easy to add new modules to the software. This for example can be:
  - New readers for file formats for already handled types of data,
  - new types of data, that should be fitted together with existing ones,
  - new models for different accelerators,
  - new fit algorithms.

Design decisions

Based on the requirements described above, the following design decisions were taken:

- **Java:** Aloha was implemented in the Java programming language. This resulted in a full integrability with all other control room software, which is also implemented in Java. Existing tools and infrastructure could be reused e.g. for data-import and Aloha itself can be used as library within other applications.
- **Swing GUI:** With Java it was also easy to implement a GUI to interact with the underlaying fitting routines. The main features of this GUI will be sketched in Sec. 7.2.3.
- **Java API for MadX:** To profit from the existing models, it was necessary to create an encapsulation of MadX, which allows direct interaction with a MadX instance. Since this is a major part and an important spin-off project by itself, this API (JMad) will be described in a separate section (Sec. 7.3).
- **Generalized fitting algorithm:** The underlaying fitting algorithm is based on the principles described in Chap. 6. It creates one big sensitivity matrix composed of contribution from different measured data and performs the fitting. This is described in detail in Sec. 7.2.2
- **Plug-in system:** To provide users with the possibility to easily add new functionality to, aloha a simple plug-in system was implemented, which will be introduced in Sec. 7.2.4.

7.2.2. Fitting Procedure

The fitting procedure in Aloha is not restricted to one simple algorithm, but is variable and extendable by the use of plugins. Nevertheless, in order to simplify explanations, we will focus in the following on the only built-in algorithm, which reflects the fitting procedure described in Chap. 6.

The Calculator, Algorithms an Solvers

Figure 7.2 gives an overview of the main components involved in the fitting procedure. The main responsible class for fit calculations is the `Calculator`. It has one implementation, `CalculatorImpl`. A calculator has a method that calculates one iteration of a fit. This e.g. can be triggered from a GUI action. The way, how the iteration is calculated, is determined by the active `Solver`. A `Solver` is tightly coupled to an `Algorithm` which does the data-preparation and post-processing for one iteration. An `Algorithm` might be able to work with more than one `Solver`. This for example is the case for the built-in `MatrixAlgorithm`: It can work with either one of the solvers implementing the `MatrixSolver` interface, namely the `SvdSolver` or the `MicadoSolver`. The information, which `Solver` is the active one is managed by the `SolverManager`.

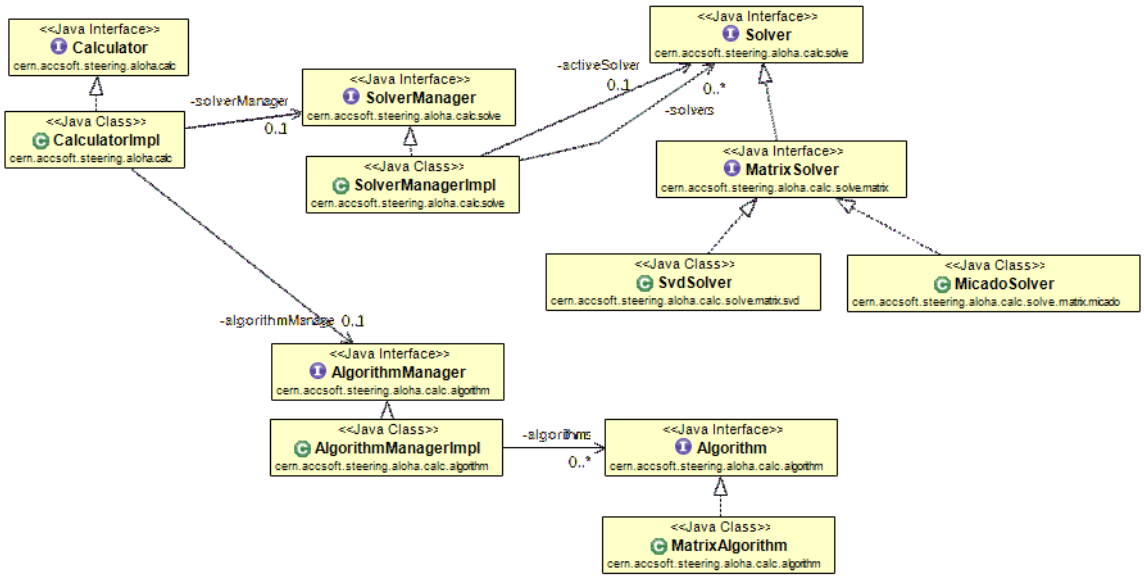


Figure 7.2.: Class diagram of classes involved in the fitting procedure in Aloha.

Thus, the whole program flow for one fit-iteration can be summarized as follows:

1. The `Calculator` retrieves the active `Solver` from the `SolverManager`.
2. It retrieves the correct algorithm for the active solver from the `AlgorithmManager`.
3. It calls the `calc()` method from the active algorithm.

This procedure is depicted in a sequence diagram in Fig. 7.3.

The Matrix Algorithm

The algorithm represented by the class `MatrixAlgorithm` is the only built-in algorithm of Aloha. As depicted in Fig. 7.2, Aloha provides two solvers that can be used by this algorithm, the `SvdSolver` and the `MicadoSolver`. The `MatrixAlgorithm` relies on the construction of one large sensitivity matrix as described in Chap. 6. The construction of this matrix will be explained in more detail in the next section. The actual calculation step is simple and is illustrated in Fig. 7.4:

1. All the necessary information is retrieved from the `SensitivityMatrixManager`: The big sensitivity matrix itself, the actual difference vector and the errors on the difference vector.

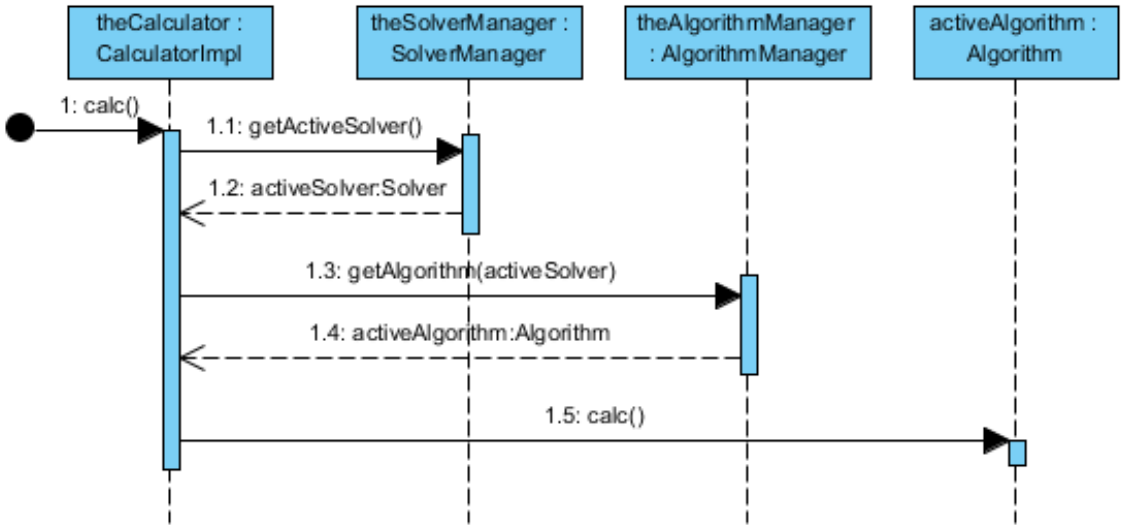


Figure 7.3.: Sequence diagram, showing `Solver` and `Algorithm` selection in Aloha.

2. This retrieved information is passed to the solver, which calculates the result for one iteration. This result includes the deltas for the parameters and an estimated error for each parameter.

3. The result calculated by the solver is passed on to the `SensitivityMatrixManager` in order to update the parameters and thus be able to prepare for the next iterations.

The Sensitivity Matrix Manager

The `SensitivityMatrixManager` is the main responsible class in Aloha to construct the sensitivity matrix. It has to collect all the information from different types of measurements. Figure 7.5 shows the main components involved in this process. Contributions can come from different measurements. Two examples of such contributors are shown:

- `KickResponseSensitivityMatrixContributor` and
- `DispersionSensitivityMatrixContributor`.

The actual creation of the sensitivity matrix is done as illustrated in Fig. 7.6:

1. Each contributor is instructed to store a state of its actual data (`initUnperturbed()`), which can be used later to calculate the differences to determine the local fit gradient.

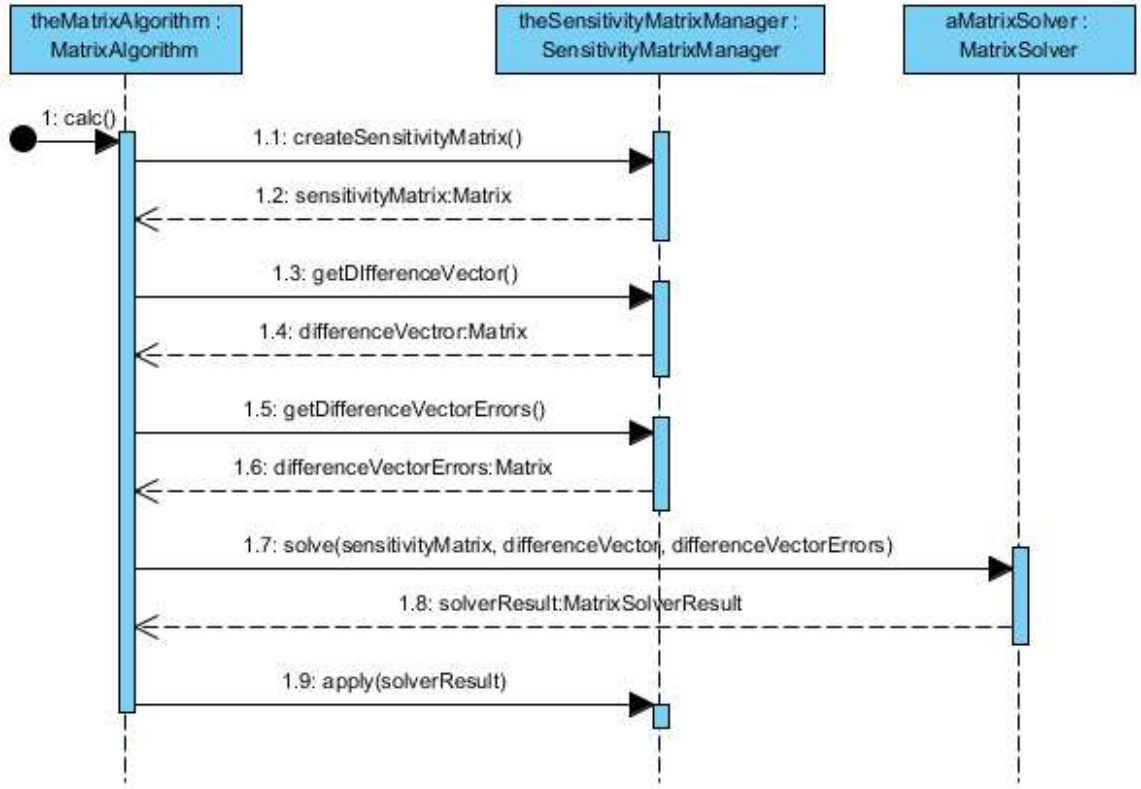


Figure 7.4.: Sequence diagram of the calculation method in matrix algorithm.

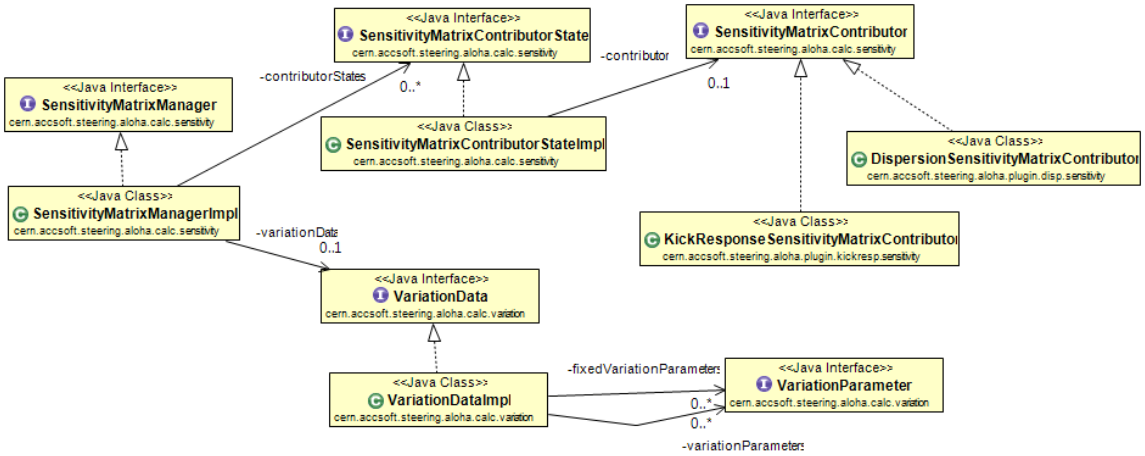


Figure 7.5.: Aloha classes involved in creating the sensitivity matrix.

2. The monitor part of the matrix is calculated for each contributor.
3. The corrector part of the matrix is calculated for each contributor.
4. An individual delta is added to one parameter and the resulting difference is calculated by each contributor (`calcPerturbedColumn()`), resulting in the local fit gradient given by Eq. (6.23). The column weighting factors ($\omega_{P_l}^{\text{auto}}$, see Eq. (6.41)) are calculated from the first contributor and used for the following ones.
5. Finally, the row normalization factors are calculated according to Eq. (6.46) and all the sub matrices are combined to one large sensitivity matrix.

All this finally results in the sensitivity matrix as defined by Eq. (6.41). The difference vector (Eq. (6.2)) is calculated in a corresponding manner, also by looping through all the contributors. This ensures the correct dimensions of the difference vector with respect to the sensitivity matrix.

7.2.3. Graphical User Interface

To be able to quickly perform online analysis, a *Graphical User Interface* (GUI) is essential. We will not try to give a full user manual of the GUI in this section, but will only show the main components and sketch the basic workflow to analyze data with Aloha.

The graphical user interface of Aloha consists of two main windows: One window displays the loaded measurement data and can show comparisons with the model. We will call it the *main-frame*. A separate window allows to configure the fit algorithms, launch a fit and display the fit results. This will be called the *fitting-frame* in the following.

The Main Frame

Figure 7.7 shows as screenshot of the *main-frame*, which is also the frame which shows up right after the launch of Aloha.

The numbers in the figure denote the main control elements, which have the following purpose:

(1) **Menus and toolbar:** These are on the top of the frame. They provide menu items and toolbar buttons to load new data or open additional frames.

(2) **List of measurements:** Here all the loaded measurements are listed. The type of the measurement is shown in the left column and an automatically generated name (e.g. from the file name or base-directory name) is shown in

the right column. If a measurement is selected in this table then its data is displayed at the respective display areas.

On another tab in the same area ('Models') all the running model instances are displayed and for which measurements they are used for comparison.

3. **Machine elements:** In this area lists of the available monitors and correctors are displayed. These lists show, which elements are currently active for the data display and analysis. The elements can be activated/deactivated by the user.

4. **Measurement options:** In this area, options for the currently selected measurement are displayed. In Fig. 7.7 the options for a kick-response measurement are displayed as an example: The user can select, which stability measurement (average over trajectories) shall be used to estimate the errors for the measurement.

5. **Measurement details:** In this area, details of the actually selected measurement are displayed. In the shown example, a kick-response measurement, a contour plot of the response matrix is shown: This matrix has a column for each used corrector and a row for each monitor. Thus each column of this matrix represents the response of one corrector for both planes.

6. **Dataviewer-explorer:** In this region a list of all available plots for the currently selected measurement is shown. The different views can be selected here by clicking on them. Which views are available depend on the type of the selected measurement.

7. **Dataviewer:** The views, which are selected in the dataviewer-explorer on the left, are shown in this region. In the example, a comparison between the measured response values (blue bars) and values calculated from the model (red dots) are shown.

The Fitting Frame

By pressing the 'fit' button 

in the toolbar of the main frame, a second frame, the fitting frame, is opened. This frame is used to configure the parameters for fitting and launch fit iterations.

A screenshot of the fitting frame is shown in Fig. 7.8, where the numbered areas have the following purposes:

1. **Fit control:** By clicking the button 'calc', the fit calculation is started. The amount of fit iterations for that calculation step can be changed by editing the value of the text field 'Iterations'. The button 'reset' sets all the actually

selected fit parameters back to their initial values. The button 'reset models' triggers a re-initialization of all the models. This is useful, when the model was changed e.g. by changing values of individual elements.

(2) **Fit configuration:** In this area, the detailed behavior of the next fit iteration(s) can be configured. On the tab pane 'Configure', the options for the solver can be configured (e.g. singular value threshold for SVD fits, include monitor- and corrector gains as fit-parameters etc.). The tab 'varied parameters' shows all the parameters (except gains) that will be varied during the next iteration and the resulting values after the fit. The tab 'fixed parameters' allows to set certain parameters to fixed values. These will not be varied during the next fit iteration. Finally, the tab 'Contributors' allows to define which measurements shall be taken into account for the next fit iteration(s) and allows to set manual weighting factors for each measurement (see Sec. 6.5.3).

(3) **Model panel:** In this area the 'model-operations panel' of JMad (see Sec. 7.3.4) is shown. It has additional functionality here: By selecting the check box in the column 'vary' in any of the panels, the respective strength in the MadX model will be added to the varied parameters of the fit. Be un-checking the check box, the parameter will be removed again.

(4) **Dataviewer-explorer:** Similar to the main frame, in this area the available graphs are listed and can be selected by clicking on them.

(5) **Dataviewer:** In this area the graphs, selected in the Dataviewer-explorer, are displayed. The example shows monitor gains calculated from a fit to kick-response data.

Workflow

To analyze data in Aloha, the following basic steps must be followed:

1. **Open measurement(s):** This is done by clicking the button 'File open'  in the Aloha main-frame. A file chooser dialog appears which allows the user to select one or more measurement files. If it is not clear from the selected files, how to load the measurement data, then the user will be asked in a separate dialog which parser to use for loading the measurement files. When the first measurement is loaded, the user will also be asked which model to use for comparison. When loading any subsequent measurement, the same type of model has to be used. Nevertheless, the user can select, if he wants to use the same instance (this is preferable, when the measurements were done at the same machine settings), or if a new instance of the model shall be created

(this is necessary if different machine settings shall be modeled for the different measurements).

2. **Select/deselect monitors/correctors:** In area ③ of the main-frame the elements which shall be taken into account for fitting can be selected. This way e.g. corrupt readings can be excluded.

3. **Choose parameters:** Open the fitting-frame by pressing the 'fit' button in the toolbar of the main-frame and select the parameters to vary in area ③ of the fitting-frame.

4. **Choose contributers:** In area ② of the fitting-frame select the measurements to take into account for the fit.

5. **Start the fit:** Click on the button 'calc' in area ① of the fitting-frame to start the fit iteration(s). The resulting values for the fit-parameters will be displayed in the fitting-frame, immediately after the fit. The model values resulting from the new parameters will be shown in the main-frame.

7.2.4. Plug-In System

To keep Aloha flexible and extensible, a simple plug-in system was put in place. The plug-in system is mainly based on interfaces and provides an auto-detection mechanism.

Extension Points

An Aloha plug-in is simply a class that implements one or more of the interfaces extending the interface `AlohaPlugin`. All the currently possible extensions are illustrated in Fig. 7.9. These extension points can provide the following additional functionalities to Aloha:

- `ReaderProvider`: A class implementing this interface can provide new readers for measurement data. This provides the possibility to read measurements from new types of files.

- `AlgorithmFactory`: A plug-in implementing this interface provides methods to create new types of algorithms.

- `SensitivityMatrixContributorFactory`: By implementing this interface, new ways of creating parts of the sensitivity matrix (see Sec. 7.2.2) for existing or new types of measurements can be added.

- `SolverProvider`: Such a plug-in can add additional functionality by adding new solvers for existing or new algorithms (see Sec. 7.2.2).
- `DisplaySetFactory`: A class implementing this interface may create display-sets for existing or new measurement types. A display-set is a combination of a panel and data-views for a measurement. This way a plug-in can define, how the data is displayed in the main-frame of Aloha.
- `AnalyzerFactory`: Such a plug-in is similar to a `DisplaySetFactory`, but a little more fine-grained: It simply can create data-views for certain type of measurements.
- `SolverConfigPanelFactory`: This type of plug-ins can create panels that will be displayed in the fitting-frame of Aloha to allow to configure parameters of certain solvers.

The class `AbstractAlohaPlugin` serves as base class for all other plugins. New types of measurements can be added by any class implementing the interface `Measurement` . In order to allow data import to Aloha, also corresponding readers have to be implemented.

Plug-in Management

Aloha contains an automatic plug-in detection mechanism. The main responsible class for plug-in handling is the so-called `PluginManager`. It simply searches in the classpath[2] for all classes implementing the interface `AlohaPlugin`. All the found classes will be instantiated by the help of the default constructor and will be registered to respective manager-classes.

To provide the plugins with the required data, a simple interface-based dependency-injection mechanism is in place: The plugins can implement one or more aware-type plugins. In such interfaces simple setter methods are defined. By implementing these methods, the plug-in (or any other class) can get references to required objects. The main class responsible for this mechanism is the `AlohaBeanFactory`. It can either create a new object from a class or configure an already created object. It checks, if the object implements one or more of the interfaces extending the interface `BeanAware` and sets the respective objects by the use of the setter methods.

[2]The Java classpath is an environment variable listing all the directories where Java applications search for libraries.

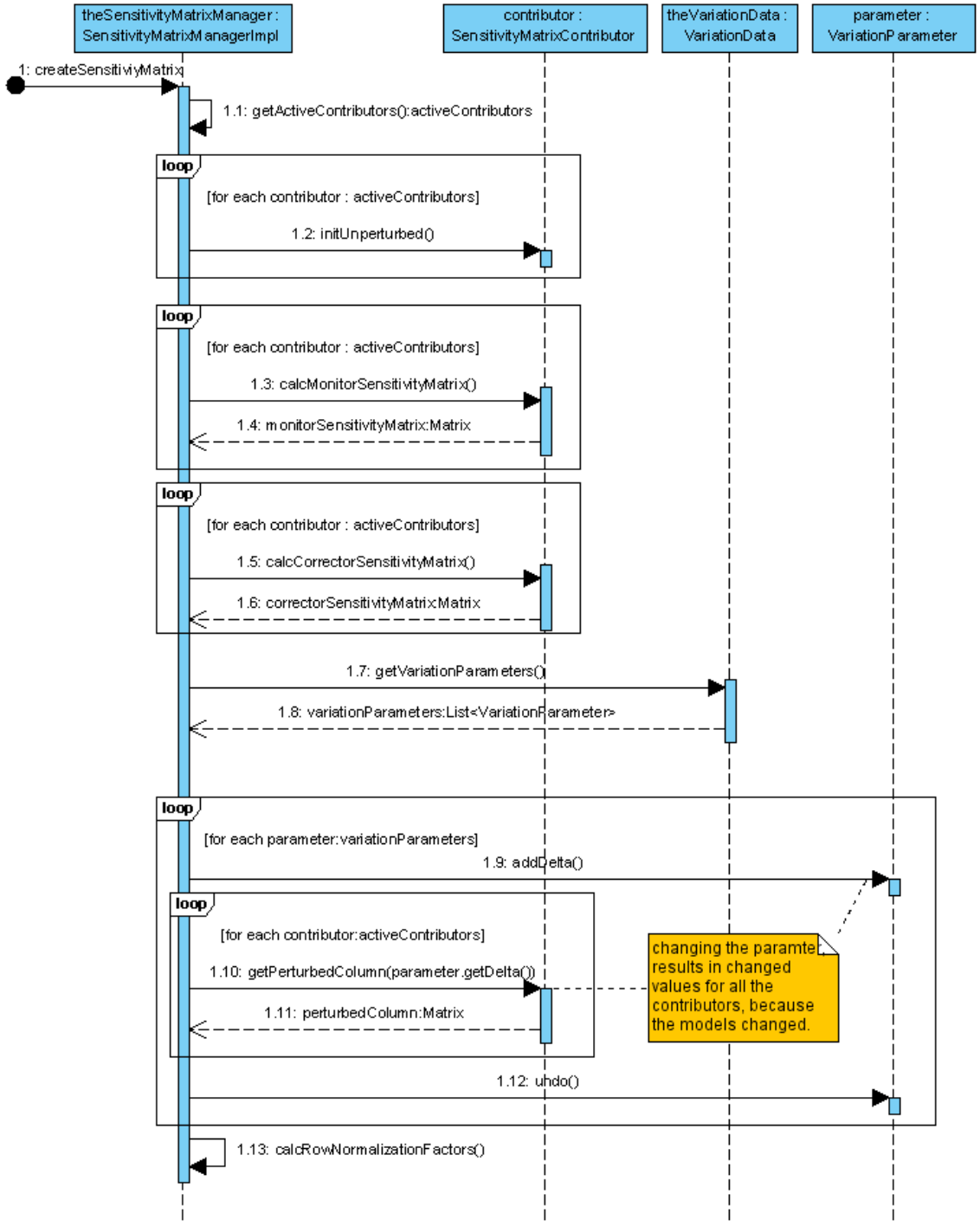


Figure 7.6.: Creation of the Sensitivity matrix.

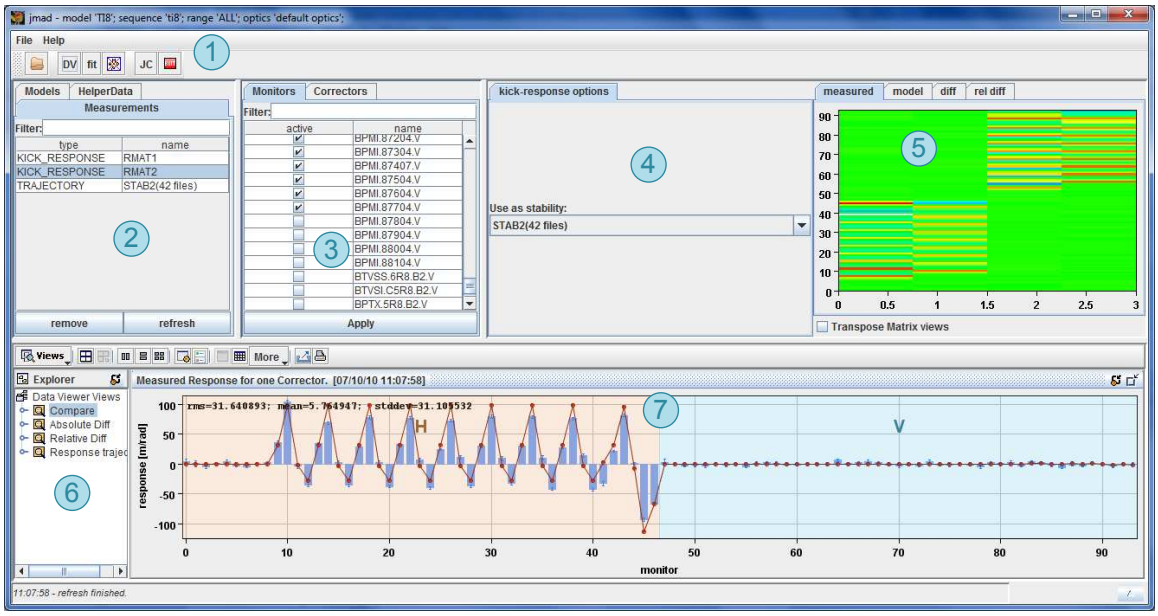


Figure 7.7.: Screenshot of Aloha main-frame.

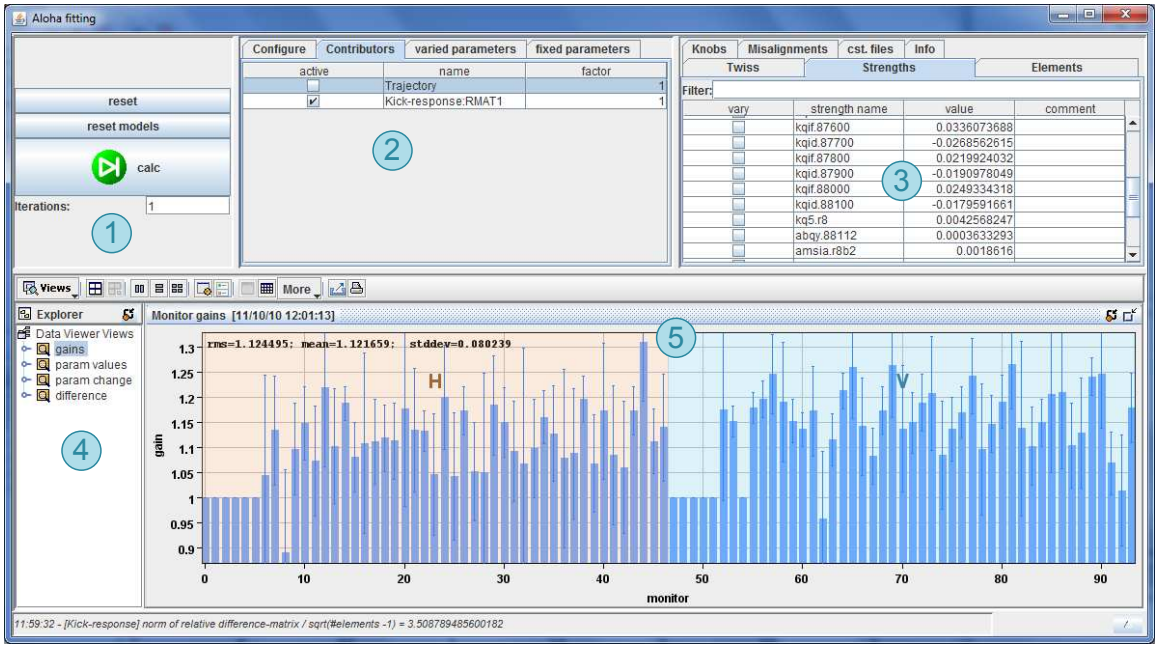


Figure 7.8.: Screenshot of Aloha fitting frame.

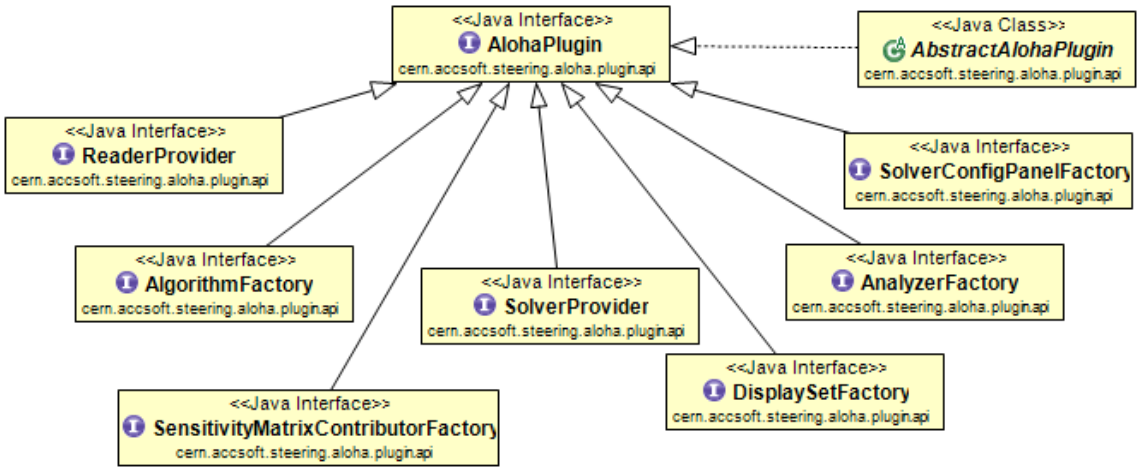


Figure 7.9.: Class diagram of Aloha plugin API.

7.3. JMad

7.3.1. Motivation and Design Considerations

Figure 7.10.: The JMad logo

As mentioned in Sec. 7.2.1, one of the design requirements of Aloha was the tight integration with the model of the accelerator under investigation, to allow fine-grained control over the fit-parameters. For the design of particle accelerators at CERN, MadX (*Methodical Accelerator Design*) [HS04] is the defacto standard software. This software is used by a very large community and there exist lattice models for many accelerators at CERN like the SPS, LHC and the transfer lines in between. These models are regularly maintained and updated. Therefore, the strong requirement arose during the design process of Aloha, to access these models as directly as possible and integrate the computing power of MadX into the fitting process. Furthermore, the need for a better integration of MadX models into the controls environment had been on the table already a long time before, e.g. for the LHC online model [MPB+10]. Because of this general interest, the decision was taken to implement a generic Java API for MadX (JMad) [FBG+10].

Interaction with MadX - the Classical Approach

MadX is implemented and maintained in the programming languages C and FORTRAN. Although it would in principle be possible to access C libraries from Java, a different approach was chosen for JMad, because MadX is per design not a library. In the contrary, it is a standalone software with its own proprietary scripting language, which is used to define the models and perform simulation tasks.

Although the MadX-language contains many elements of a scripting language (like loops or if/else statements) it is by no means (and never was intended to be) a full programming language with custom libraries. Therefore, the necessity arises to create MadX input files and post-process output data with other tools, especially when doing complex simulation tasks. The classical way of using MadX from a higher level programming language other then MadX scripting is:

1. Create a input file for MadX (ASCII file) containing model definition calls,

input parameters and commands to export the results.

2. Call MadX with the created input file.
3. Wait until MadX terminates.
4. Parse the MadX output files.
5. Post-process the data (e.g. plot).

Although this can be easily done because of the highly configurable MadX text file output features it has many disadvantages, like e.g.:

- Creating MadX files by simply composing strings as demanded by the application is very error-prone and makes the application code very dependent on the MadX scripting language as well as on the model in use.
- Running MadX with different input files requires to start and stop MadX every time. Since this also requires to load the sequence (model definition) every time, this becomes a very time consuming procedure, especially when many of such iterations are needed (e.g. for fitting purposes).
- Every application developer ends up in implementing its own MadX parser.

Interaction with MadX - the JMad Approach

All these disadvantages mentioned above can be avoided if steps 1 to 4 are encapsulated in a dedicated software package with a well defined interface for the higher level programming language. All the communication can then be done in the language-typical way which is normally (at least in the case of Java) compiler checked and type safe. Even the starting and stopping of MadX can be avoided by keeping a running instance with the actual model status in memory.

The JAVA programming language was chosen for the first implementation of this API simply because the controls environment of modern accelerators at CERN, e.g. of the LHC, is dominated by Java applications and because of the concrete requirements of Aloha (see Sec. 7.2), which was intended to serve as first use-case for the API. Meanwhile, also a Python implementation of such an API based on the same principles (PyMad) is under development.

7.3.2. Concepts

Figure 7.11 shows the key components of the JMad design. They are described in some more detail in the following.

JMad Service

The main facade component for an application which is using the API is the interface `JMadService`. The key responsibility of this interface is to find available model definitions and create model instances from these definitions. The following description will focus on these two key responsibilities, although a lot of additional functionality is provided by this service.

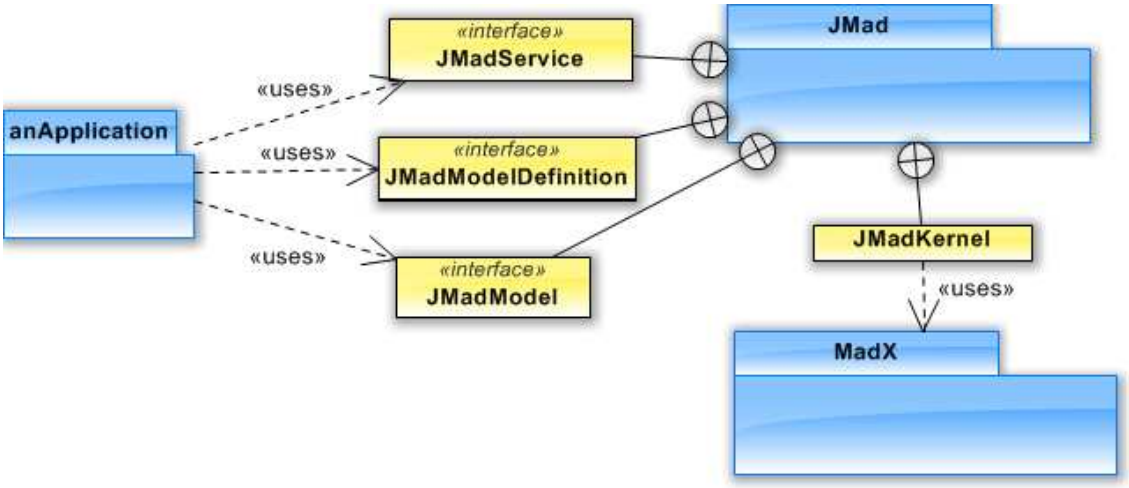


Figure 7.11.: Overview of JMad key components.

Models

A model is the key component of JMad. It is represented by the `JMadModel` interface. Each `JMadModel` instance is associated to one dedicated MadX process. The `JMadModel` interface provides Java methods to act on the model (e.g. run a twiss calculation, get/set strength values and many more) which are passed on to the MadX process.

Model Definitions

Although a model can be created simply from scratch by creating a new instance and calling certain methods one by one, the proposed (and most convenient) way is to use predefined model definitions. In the API, a model definition is represented by the `JMadModelDefinition` interface. Instances thereof contain all the information which is necessary to initialize the MadX process (e.g. all the required sequence- and strength-files) as well as available options which are possibly selectable by the user/application (like available sequences, ranges or optics).

Predefined model definitions are stored in xml files and contain only the minimum necessary information. The design goal here was to profit as much as possible from the MadX scripting language and as a consequence being able to reuse as much as possible from existing MadX-files. As a consequence these xml model definitions only act as a link between MadX-scripts and are easily understandable and maintainable by conventional MadX users.

JMad contains an auto-detection mechanism for such model definitions. This mechanism searches for model definitions contained in the JAVA class path in a distinct package. This makes it very easy to extend the available model definitions: A new model definition file must simply be placed in the correct package on the classpath, either as a simple file or inside a jar[3]. Model definition xml files can also be packaged inside a zip file, together with all required MadX files. This is very useful e.g. to archive model definitions or to pass them on to other MadX users.

Communication with MadX

Figure 7.11 shows that the object which is responsible for the direct communication with MadX is an instance of `JMadKernel`. A kernel takes care of its own MadX process and the related input-, output- and logging-files. Currently the communication is simply based on streams and files: All the commands and input data are directly written to the input stream of the MadX process by the kernel. All MadX output data is redirected to files (mostly twiss files) which are parsed by the kernel after the command has finished. Although one could imagine more sophisticated communication methods (e.g. compiling MadX as a shared library and communicating via JNA[4]) this method was chosen for the first implementation because it works with the existing MadX binaries and only depends on the MadX scripting language (which is not supposed to change very frequently) but not on MadX internals.

Each `JMadModel` is using its dedicated instance of a `JMadKernel`. The communication between the model and the kernel is done by special command objects whose responsibility it is to compose the correct command strings for MadX.

Operating System Independence

Since MadX binaries are platform dependent it is evident that a library based on executing these binaries can never be fully platform independent. The problem is circumvented the following way: Executables for different operating systems are included in every JMad release. On startup of the JMad service, the correct binary for the operating system is extracted to a temporary directory and run from there

[3] A jar is a *Java-Archive*. This is the format in which java libraries are commonly deployed.
[4] Java Native Access [Sun10]

whenever needed. Currently the operating systems Windows, Linux and Mac OS X are supported.

7.3.3. Usage Examples

To illustrate the usage of the API and introduce some of the features, some simple JAVA code examples are shown in the following.

Initializing a Model

A typical way to set up a `JMadModel` is shown in Listing 7.1: First a new service is created (Typically an application would use only one such service). This service is then used to find a model definition named "ti2" (This is the name of a predefined model definition) and create a model. After starting the model (`model.init()`), it is ready to be used. When finishing the work with the model, `model.cleanup()` should be called to close all log files and end the corresponding MadX process.

Listing 7.1: model setup example

```
/* create a new JMad service */
JMadService jmadService = JMadServiceFactory.createJMadService();

/* find a model definition */
JMadModelDefinition modelDefinition = jmadService.
   getModelDefinitionManager().getModelDefinition("ti2");

/* create the model */
JMadModel model = jmadService.createModel(modelDefinition);
model.init();

/* do something with the model */

/* finally do a cleanup */
model.cleanup();
```

Access Model Elements and Optics

JMad offers two different levels of abstraction. The top abstraction layer, which we recommend to use, is a full Java representation of the machine which is simulated. Every machine-element (as defined in a MadX sequence) is modeled as a separate Java class with appropriate properties and access methods, e.g. `Bend`, `Quadrupole` or `Monitor`. Listing 7.2 shows a simple example which retrieves all the elements of the active model range and just prints their name and element type.

Listing 7.2: retrieving elements

```
/* get all the elements */
List<Element> elements = model.getActiveRange().getElements();

/* print name and type of each element */
for (Element element : elements) {
  System.out.println("name: " + element.getName() +  "; type: " +
      element.getElementType());
}
```

The actual optics values (for the whole machine or for single elements) can be retrieved by separate methods. Some of these features are demonstrated in listing 7.3.

Listing 7.3: retrieving optics values

```
Range activeRange = model.getActiveRange();

/* retrieve an element by name (MONITOR) */
Monitor monitor = (Monitor) activeRange.getElement("BPMIH.22604");

/* retrieve the actual optics */
Optic optic = model.getOptics();

/* retrieve some optics values */
List<Double> betaxValues = optic.getValues(MadxTwissVariable.BETX,
    activeRange.getElements());

/* retrieve optics values for one element */
OpticPoint opticPoint = optic.getPoint(monitor);
double monX = opticPoint.getX();

/* increase a quad strength by 10 percent */
Quadrupole aQuad = (Quadrupole) activeRange.getElement("MQIF.20400");
aQuad.setK1(aQuad.getK1() * 1.1);

/*
 * IMPORTANT:
 * refetch optics after changing strengths,
 * it has changed!
 */
optic = model.getOptics();
```

Custom Twiss

Although it is recommended to use JMad as described in the previous sections, since this provides the cleanest abstraction to low-level MadX, it is sometimes useful or

unavoidable to have more fine-grained control over MadX. An example could be to run custom twiss commands and just reading the results for some twiss-variables without the need of creating the full `Optic` object all the time (which can become time consuming for large models). Such a use case is demonstrated in listing 7.4: The content of the result is defined by the `TfsResultRequest` object which is passed to the twiss method. This method returns a `TfsResult` from which finally the requested values can be retrieved.

Listing 7.4: custom twiss

```
TfsResultRequest request = new TfsResultRequest();

/* a regexp for the elements */
request.addElementFilter("BPM.*");

/* and the variables we want */
request.addVariable(MadxTwissVariable.NAME);
request.addVariable(MadxTwissVariable.X);
request.addVariable(MadxTwissVariable.Y);

/* run the twiss and get the results */
TfsResult result = model.twiss(request);

List<String> elementNames = result
    .getStringData(MadxTwissVariable.NAME);
List<Double> xValues = result.getDoubleData(MadxTwissVariable.X);
List<Double> yValues = result.getDoubleData(MadxTwissVariable.Y);

/* print the values */
for (String name : elementNames) {
  int index = result.getElementIndex(name);
  System.out.println(name + ": X=" + xValues.get(index) + "; Y="
      + yValues.get(index) + ";");
}
```

7.3.4. Graphical User Interface

Although the intended purpose of JMad is to be used as a library, also a graphical user interface (GUI) was created. This GUI allows to open different models, change arbitrary settings of the models and to create plots of the resulting optics functions.

The Main Frame

Figure 7.12 shows a screenshot of the main frame of this GUI. The areas in this frame have the following purpose:

Figure 7.12.: Screenshot of JMad GUI.

1 **Menus and toolbar:** The toolbar provides buttons to open a new model and to choose different ranges and optics for the actual model.

2 **List of models:** All the actually opened models are displayed in this list. In the example, two models ('sps' and 'longti8') are opened. By selecting one of the models in the list, the selected model becomes the active one and its values are displayed and can be edited in the other areas.

3 **Model-operations panel:** In this area, properties of the model can be changed. Editable values are for example twiss initial conditions, strength values or properties of individual elements. In the example, the panel for changing individual element-properties is shown: On the left, a list of all available elements is displayed. The right half of the panel contains an editable table of all the properties of a the selected element. The optics values at the position of the selected element are listed in a separate table.

4 **Data panel:** Selected output data is shown in this panel. In the current version, two tabs are available: A tune diagram and a table with the summary output of the last executed twiss command.

5 **Dataviewer-explorer:** In this area, all the currently available plots are listed and can be selected.

6 **Dataviewer:** In this area, the plots that are selected in the Dataviewer-explorer, are displayed. The example shows a plot of the beta- and dispersion-functions of the TI8 transfer line and the adjacent LHC sector 78, using two different y-axes. The example plot also demonstrates the possibility to mark certain elements in the plots (e.g. the position of the quadrupole MQIF.81000 is shown in the plot, simply by selecting the check box 'mark' in the elements-table of area 3).

7 **Plot buttons:** Below the data view two buttons are displayed: The button 'Add view' opens a separate dialog in which a new plot can be defined (see next section). The button 'Refresh all views' recalculates the optics values and updates all plots and the output data in the data panel 4. The button '>>> ref' copies the current optics values to a reference dataset, that can be used later on to produce difference plots.

Creating Views

When the button 'Add view' at the bottom of the main-frame is clicked, a dialog as shown in Fig. 7.13 appears. The new view can be configured in the following ways:

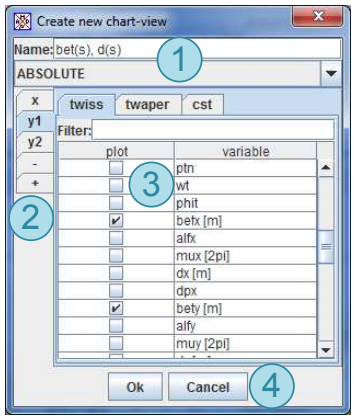


Figure 7.13.: Screenshot of dialog to create a new view in JMad.

1 **Set name and type:** An arbitrary name can be entered in the text field 'name' to identify the view later on. If no name is entered, a name will be generated automatically, based on the selected variables. The combo-box below the 'name'

field allows to set the type of the plot. The default value is *Absolute*. Alternative values are *Difference*, *Relative* or *Beating* which result in plots that relate the actual values to those of a reference data set.

2. **Manage axes:** Per default two tabs ('x' and 'y') are shown. By using the additional pseudo-tabs '+' and '-', additional vertical axes can be created and deleted again.

3. **Select variables:** Each axis-tab has the same sub-tabs to select different types of variables. For the x-axis, only one variable can be selected, whereas for any y-axis an arbitrary amount of variables can be selected. The selection of variables is done by selecting the check boxes (or radio buttons for the x-axis) in the 'plot' columns of the tables.

4. **Create the view:** When the button 'Ok' is pressed, the dialog is closed and the new view is created. Pressing the button 'Cancel' only closes the dialog, but does not create any view. Closing the dialog (e.g. in windows by pressing the 'x' at the top right corner of the window) is equivalent to pressing the button 'Cancel'.

7.3.5. Other Applications and Outlook

Besides Aloha, which served as the initial case study, other applications and systems are now using JMad. The most important client is the LHC Online Model [MPB+10]. It uses JMad functionality, especially the feature to define a machine and its optics in a `JMadModelDefinition`, as a backbone to calculate optics functions, upload them to LSA[5] and to create knobs. Additionally, the online model can extract power converter settings from LSA and directly pass them on the JAVA-level to JMad to predict aperture margins during aperture measurements or scan over generated settings of squeeze beam processes for the LHC, to verify the settings and check / predict the variation of machine parameters, like tune, chromaticity or beta beating between the matched optics points.

Currently, work is ongoing on the development of an *aperture-meter* for the LHC. This application displays the currently available aperture of the LHC in realtime and uses the model e.g. to simulate orbits from settings in the real machine. In the course of this work a JMad-online service is under development as bottom layer. This service will provide general functionalities to update models with actual settings from the real machine and will be usable as library from within other applications [Mue].

[5] LHC software architecture [ACC+05]

8. Optimization of the Injection lines

8.1. Introduction

The inconsistencies in the optics of the LHC injection lines, that motivated this work, were already outlined in Chap. 5 (Sec. 5.3). To solve these issues, an appropriate analysis method had to be found (Chap. 6) and was implemented in online tools, as described in Chap. 7.

In the following chapter, we present the most important analysis results based on these methods. We describe the measurements and analysis steps and present the measures taken to improve the models of the transfer lines. Finally, we use actual measurements to demonstrate the high beam quality at injection into the LHC, which can be considered as a result of a common effort, this work being part of it.

For first considerations, we mainly focus on the TI8 transfer line, since the described effects were first observed in this line. The effects are stronger there and, as a consequence, more measurements were taken for that line. We will later relate the results to the TI2 transfer line and compare the findings.

8.2. TI8 optics changes in 2008 and initial observations

Following several attempts to empirically correct the optics of the transfer lines, different optics settings were used at different times in the TI8 transfer line during the year 2008. All the optics changes are collected in Table 8.1. To be able to refer to the different optics settings later on, we have assigned names to these optics. These optics names are our own convention: They simply refer to the date and time (Geneva local time), when the corresponding changes were applied to the machine. Figure 8.1 displays the changes of the described quadrupole strengths throughout the year and Fig. 8.2 shows the changes of the momentum of the line, for later reference.

In the following, we describe these optics changes in more detail and discuss their influence on the measured properties of the tranfer line.

optics name (=local time)		**machine settings**
ti8-2007-09-13	matching to LHC 5.0, V. Kain.	kqid.80500 = $-0.03384\,\mathrm{m}^{-2}$, kqif.87600 = $0.03312\,\mathrm{m}^{-2}$.
ti8-2008-05-24-2125	reduced kqid.80500 by 0.65%.	kqid.80500 = $-0.03362\,\mathrm{m}^{-2}$.
ti8-2008-08-23-1957	newly matched, M. Meddahi.	initial $D_x = -0.25279\,\mathrm{m}$, initial $D'_x = 0.00334$, kqif.87600 = $0.03377\,\mathrm{m}^{-2}$.
ti8-2008-08-24-0327	kqif.87600 reduced by 9%.	kqif.87600 = $0.03098\,\mathrm{m}^{-2}$.
ti8-2008-09-18-1034	kqid.80500 and kqif.87600 back to original, strengths rematched.	kqid.80500 = $-0.03384\,\mathrm{m}^{-2}$, kqif.87600 = $0.03361\,\mathrm{m}^{-2}$.

Table 8.1.: Optics changes in TI8 during summer 2008.

8.2.1. 2007 Optics

The investigations started with a nominal optics, which was matched to the LHC V 5.0 optics in September 2007. The same values were used in the machine as well as in the model. For later reference we will denote this initial optics by "ti8-2007-09-13".

8.2.2. Reduction of kqid.80500

The strength of the defocussing quadrupole chain in the transfer line (kqid.80500[1]) was changed from $-0.033838\,\mathrm{m}^{-2}$ to $-0.033618\,\mathrm{m}^{-2}$ in the machine (reduction of about 0.65%). This was done in order to correct a phase advance error in the vertical plane, which was found in earlier investigations in 2008 and in 2004 [Wen06]. The new value was trimmed into the machine on May 25, 2008 at 21:25 and was kept until September 18, 2008 at 10:34, when finally a totally new optics was put in place. In the following, we refer to these optics settings with reduced strength kqid.80500 as "ti8-2008-05-24-2125".

This reduction was assumed to be a correction to the machine (for some unknown error). Therefore, the strength in the model ought not to be changed. And indeed, this empirical correction improves the agreement between measurement and model

[1]In the 2008 MadX files, this strength is denoted by kmqid8710m. We will use the denotation introduced in 2009, since this is consistent with the trim-editor, where this strength is represented by the knob RQID.80500/K.

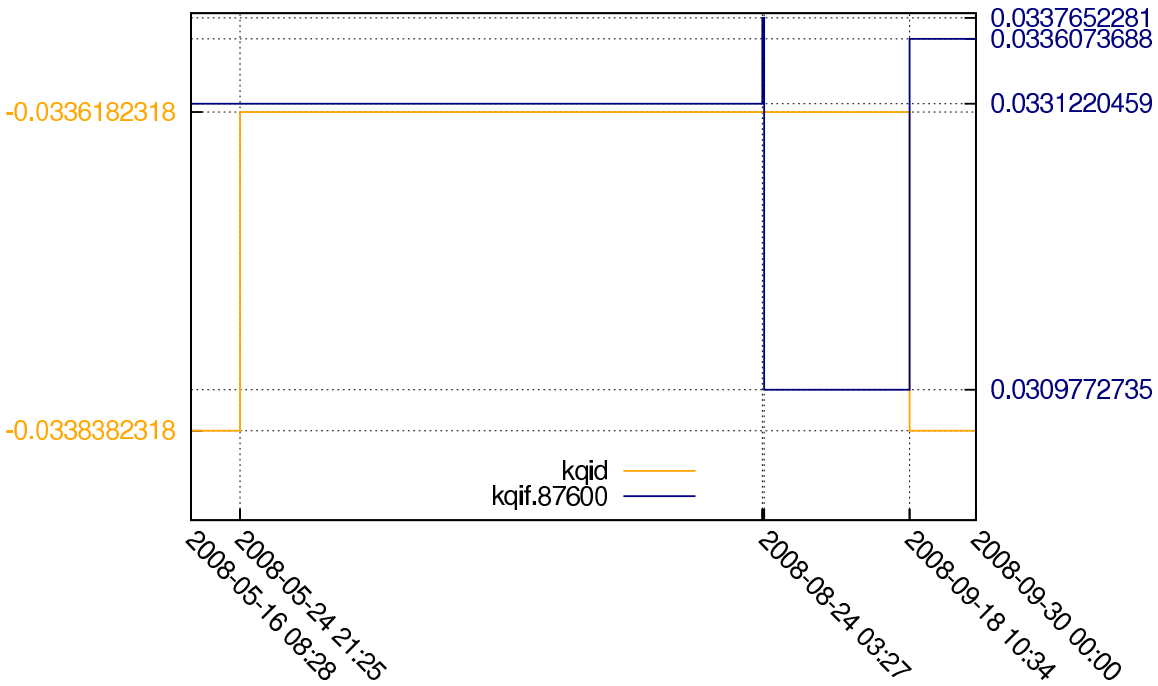


Figure 8.1.: Effect of optics changes of the TI8 transfer line during summer 2008 on two characteristic strength parameters.

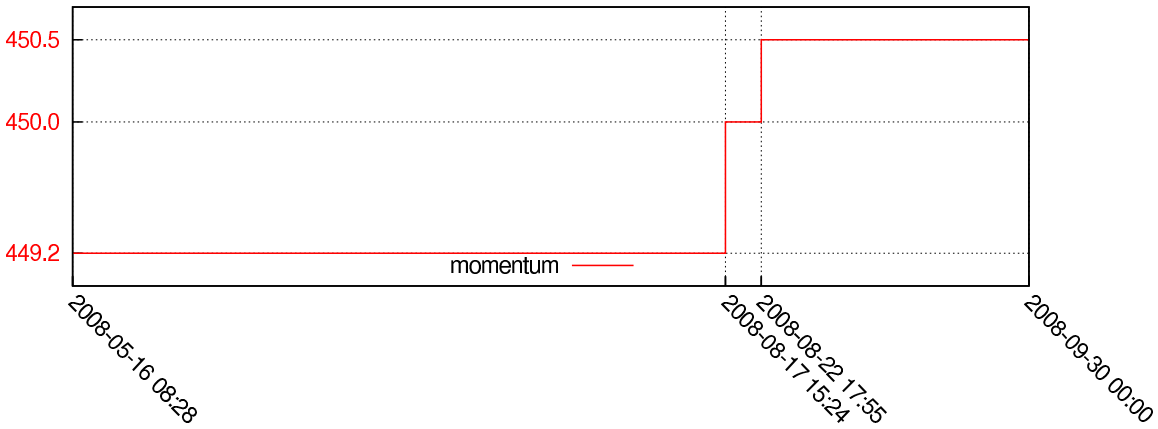
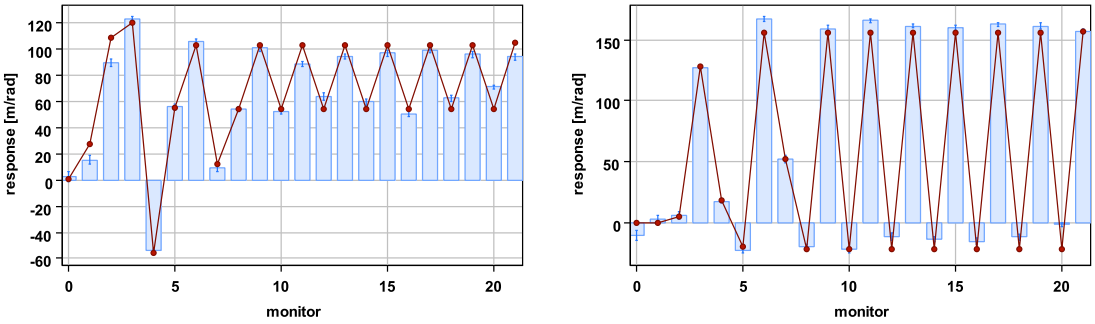


Figure 8.2.: Momentum trims of the TI8 transfer line during summer 2008.

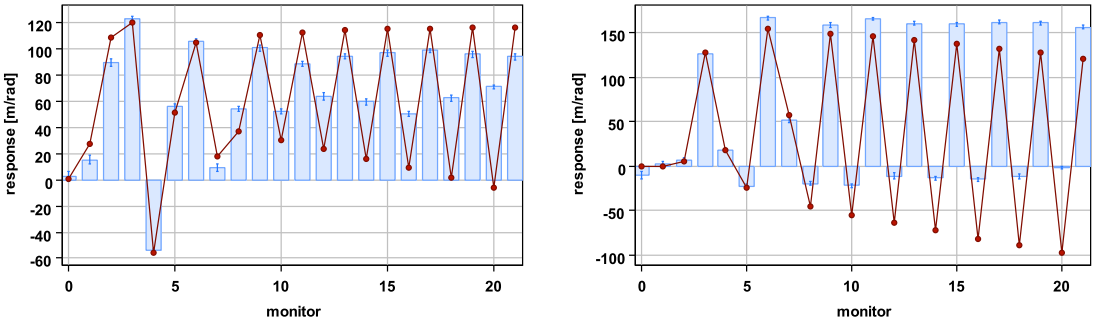
for kick-response measurements: Figures 8.3 show some example (vertical) corrector responses with reduced kqid.80500 in the line compared to the model with the original

strength[2]. These plots show a better agreement of the measurement- and model-data than those comparing the same measured data with a model that uses the same strength as in the machine (Figs. 8.4). As the change is in the defocussing quadrupole strength, the effect on the horizontal plane is minimal (and therfore not shown here).



(a) MDMV.400097 (b) MDSV.400293

Figure 8.3.: Vertical responses for 2 correctors with reduced kqid.80500 in machine, compared with model using original 2007 optics ("ti8-2007-09-13").



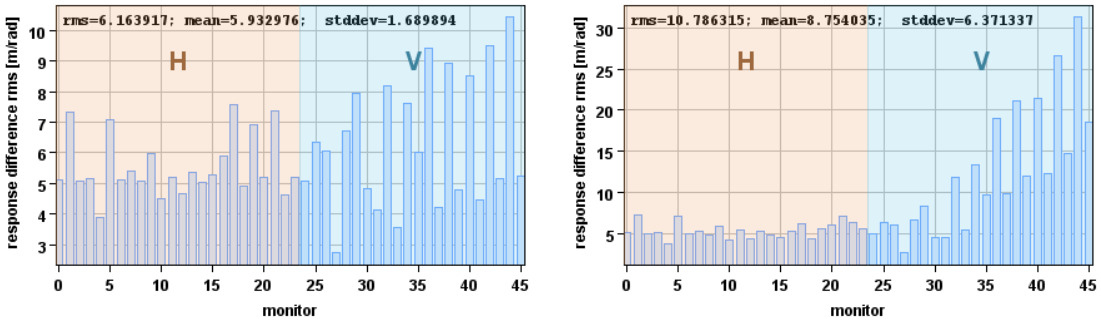
(a) MDMV.400097 (b) MDSV.400293

Figure 8.4.: Vertical responses for 2 correctors with reduced kqid.80500 in machine, compared with model using the same settings as in the machine ("ti8-2008-05-24-2125").

As illustrated in Figs. 8.5, the mean rms difference for each monitor between measurement- and model- responses, is only 5.9 $^{m}/_{rad}$ when using the original model

[2]Data taken: 2008-05-24 21:33 to 22:18

optics ("ti8-2007-09-13"), compared to 8.8 $^{m}/_{rad}$ when using model settings equal to the real machine settings ("ti8-2008-05-24-2125"). So the reduction of kqid.80500 definitely improves the agreement of the measurement with the (original) model.



(a) Model optics "ti8-2007-09-13"

(b) Model optics "ti8-2008-05-24-2125"

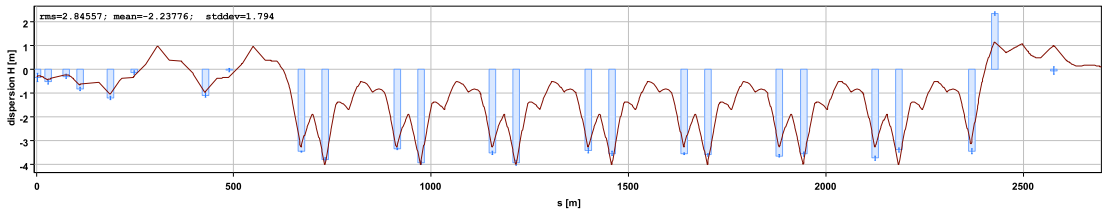
Figure 8.5.: RMS differences per monitor between measurement and model; Same measurement, different model settings. Note the different vertical scales: The maximum difference for optics "ti8-2007-09-13" is about $\frac{1}{3}$ of the maximum for optics "ti8-2008-05-24-2125".

Nevertheless, as mentioned before, this trim was seen as an empirical correction and no physical explanation had been found at that time. Furthermore, since this empirical correction was deduced from kick-response measurements only, it is advisable to crosscheck its effect on the dispersion. To do so, we use a dispersion measurement which was taken in August 2008 using still the same optics in the transfer line[3]: Figures 8.6 compare the measured horizontal dispersion of TI8 with reduced strength kqid.80500 in the machine and the original strength in the model (optics "ti8-2007-09-13"). As visible in Fig. 8.6(b), the rms of the difference between measurement and model, normalized by $\sqrt{\beta_x}$, is 0.0354 $\sqrt{m}$ in this case. The error is increasing towards the end of the line.

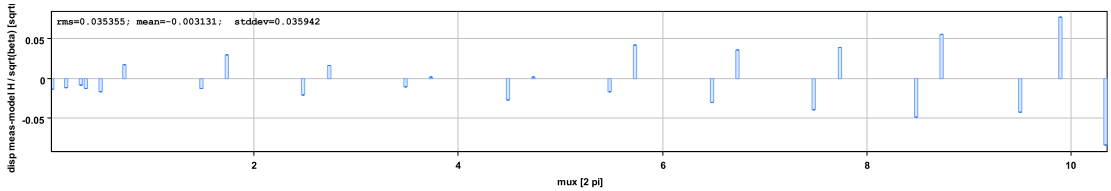
On the other hand, using the same optics in the model as it was trimmed in the machine results in a rms difference of 0.0293 $\sqrt{m}$, as shown in Fig. 8.7(b). So this empirical correction is doubtable, since it does not improve the dispersion agreement between measurement and model. On the contrary, the agreement of the dispersion is slightly better, when using the correct model.

A second reason, why this empirical correction is problematic, is that there is no meaningful physical explanation for such an assymmetry between the errors on the two main quadrupole strengths: The focussing strength seems to be rather correct, while the defocussing one shows the described error of 0.65%.

[3]Data taken: 2008-08-23 18:53

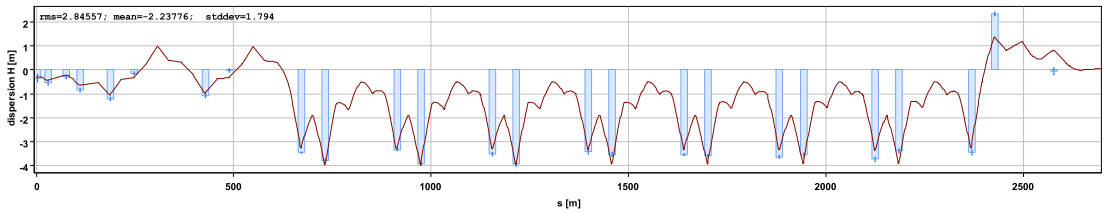


(a) Measured dispersion (blue bars) compared with model dispersion (red line).

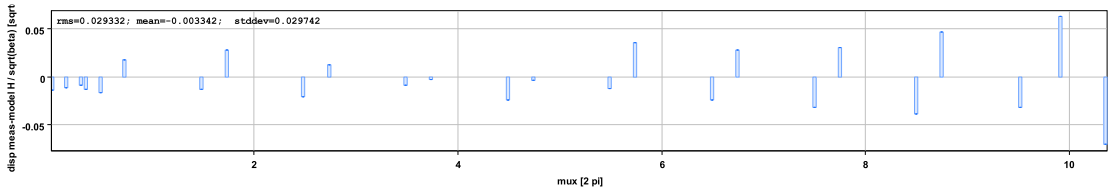


(b) Dispersion difference between measurement and model, normalized by $\sqrt{\beta_x}$.

Figure 8.6.: Comparison between measured dispersion with reduced kqid.80500 in machine, model with original 2007 optics "ti8-2007-09-13".



(a) Measured dispersion (blue bars) compared with model dispersion (red line).



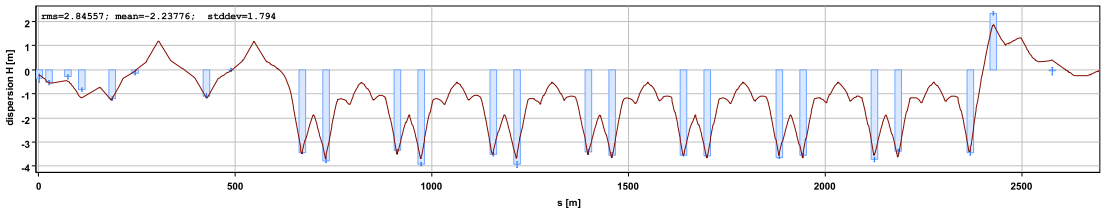
(b) Dispersion difference between measurement and model, normalized by $\sqrt{\beta_x}$.

Figure 8.7.: Comparison of horizontal dispersion with reduced kqid.80500 both, in machine and model. Model optics "ti8-2008-05-24-2125".

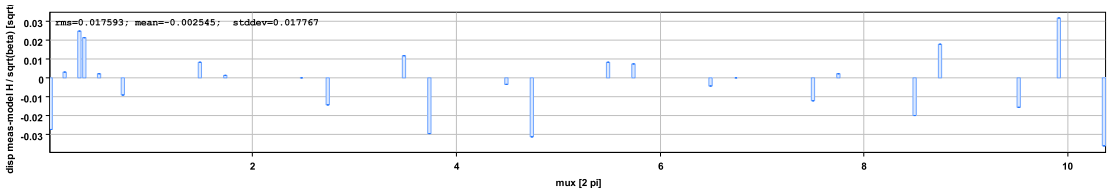
8.2.3. New Initial Conditions Deduced from Measurements in May 2008

From measurements in May 2008, new initial conditions (resulting from the SPS optics) for the horizontal dispersion and its derivative were estimated to about $D_x =$

$-0.253\,\mathrm{m}$ and $D'_x = 0.003$ instead of the previous $D_x = -0.329\,\mathrm{m}$ and $D'_x = 0.0123$ [MAF+08]. The use of these initial conditions indeed reduces the error of the normalized horizontal dispersion to $0.0176\,\sqrt{m}$ as illustrated in Figs. 8.8



(a) Measured dispersion (blue bars) compared with model dispersion (red lines).



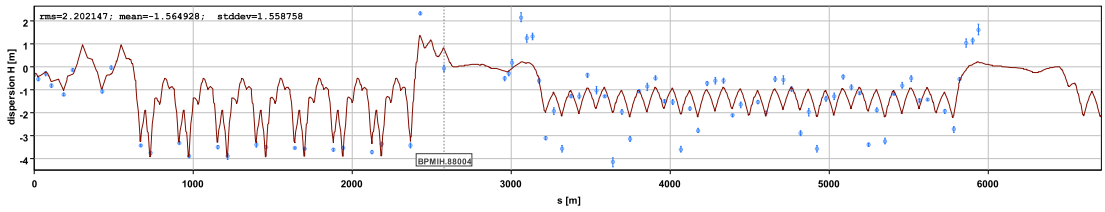
(b) Dispersion difference between measurement and model, normalized by $\sqrt{\beta_x}$.

Figure 8.8.: Comparison of horizontal dispersion with reduced kqid.80500 both, in machine and model (optics "ti8-2008-05-24-2125") with new initial conditions $D_x = -0.253\,\mathrm{m}$ and $D'_x = 0.003$.

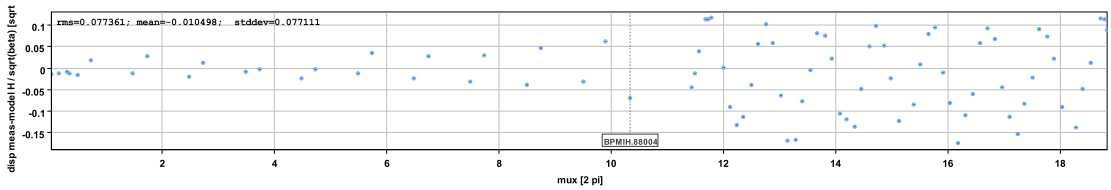
On August 23, 2008 protons were injected for the first time through the TI8 transfer line into the LHC throughout to point 7. As already mentioned in Sec. 5.3.2, a dispersion mismatch between measurement and model was observed, starting around the injection point. By using the new initial conditions, the rms of the normalized difference for the transfer line plus LHC sector 78 is reduced from $0.077\,\sqrt{m}$ (see Fig. 8.9(b)) to $0.063\,\sqrt{m}$ (see Fig. 8.10(b)).

Despite this improvement, simply changing the initial conditions results in a clearly mismatched model optics between TI8 and LHC as illustrated in Fig. 8.10(a).

To correct this mismatch, a newly matched optics was calculated by M. Meddahi, which was trimmed in the machine on August 23, 2008 at 19:57. This new optics contains the reduced strength of kqid.80500, as described in the previous section, and the new initial conditions for the initial horizontal dispersion and horizontal dispersion derivative ($D_x = -0.2527900995\,\mathrm{m}$, $D'_x = 0.0033414458$). We will refer to this optics later as "ti8-2008-08-23-1957". The resulting dispersion is shown in Figs. 8.11. Although the normalized rms difference is a slightly reduced to $0.051\,\sqrt{m}$, compared to $0.063\,\sqrt{m}$ in Fig. 8.10(b), the dispersion wave starting around the junction of the transfer line and the LHC remains unexplained (see Fig. 8.11(b)).

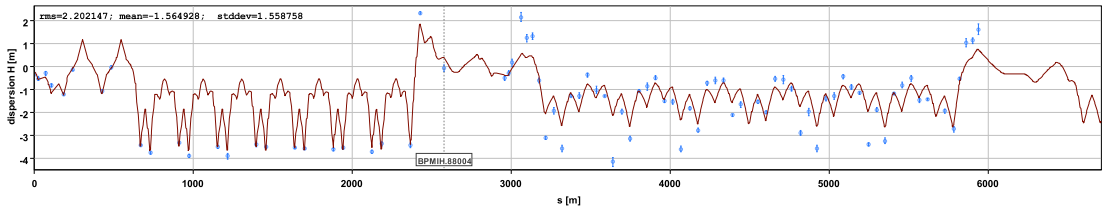


(a) Measured dispersion (blue bars) compared with model dispersion (red lines).

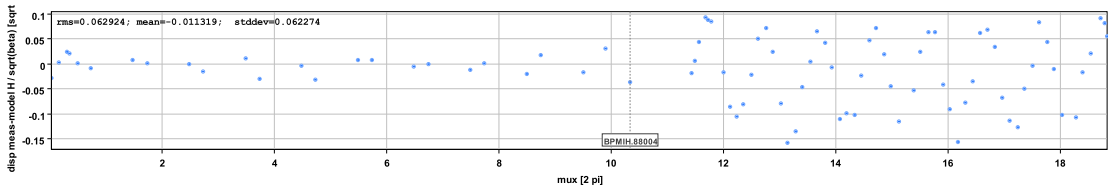


(b) Dispersion difference between measurement and model, normalized by $\sqrt{\beta_x}$.

Figure 8.9.: Comparison between measured dispersion with reduced kqid.80500 both, in machine and model. TI8 plus LHC sector 78. Model optics "ti8-2008-05-24-2125".

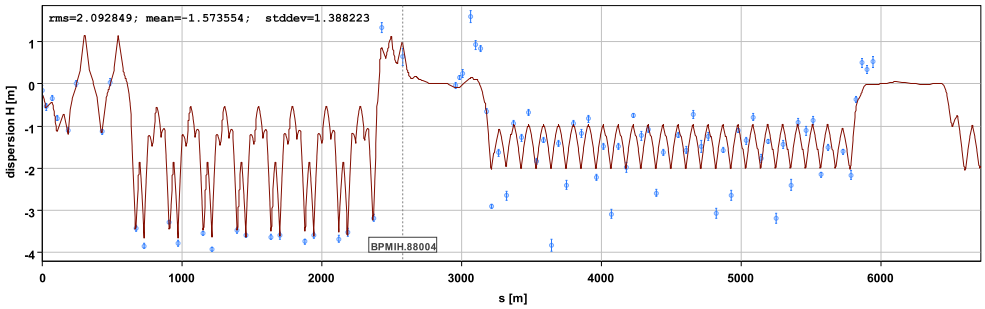


(a) Measured dispersion (blue bars) compared with model dispersion (red lines).

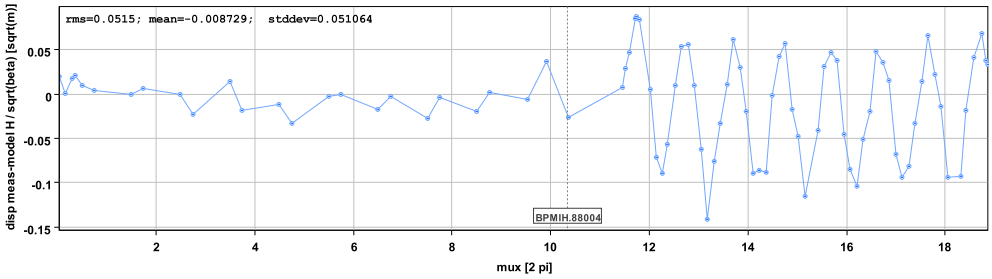


(b) Dispersion difference between measurement and model, normalized by $\sqrt{\beta_x}$.

Figure 8.10.: Comparison between measured dispersion with reduced kqid.80500 both, in machine and model (optics "ti8-2008-05-24-2125") with new initial conditions $D_x = -0.253\,\mathrm{m}$ and $D'_x = 0.003$. TI8 plus LHC sector 78.



(a) Measured dispersion compared with calculated.



(b) Normalized dispersion difference between measurement and model.

Figure 8.11.: Comparison between measured dispersion with newly rematched optics "ti8-2008-08-23-1957" in both, machine and model.

8.2.4. Reduction of kqif.87600

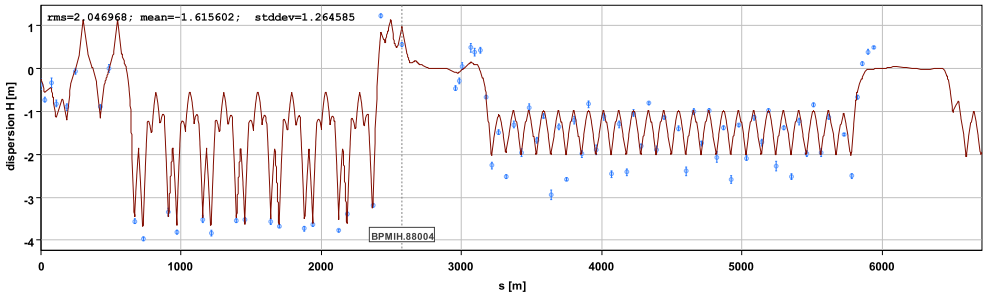
Several options were evaluated to empirically correct this dispersion mismatch at the TI8-LHC junction. One is the reduction of the strength kqif.87600[4] by 9% (from 0.033765 m^{-2} to 0.030977 m^{-2}) as discussed in [MAF+08]. This is the strength of the individually powered quadrupole MQIF.87600 towards the end of the TI8 transfer line. The new setting was trimmed into the machine on August 24, 2008 at 03:27 and left in until September 18, 2008 at 10:34 when totally new optics settings were put in place. The rest of the optics settings is the same as described in the previous section. We will refer to these settings as "ti8-2008-08-24-0327".

Again, this trim was assumed to be a correction, so the model-optics ought not to be changed. The resulting dispersion[5] is shown in Figs. 8.12. Indeed the rms of the normalized difference between measurement and model is reduced to $0.033\sqrt{m}$ for the horizontal dispersion.

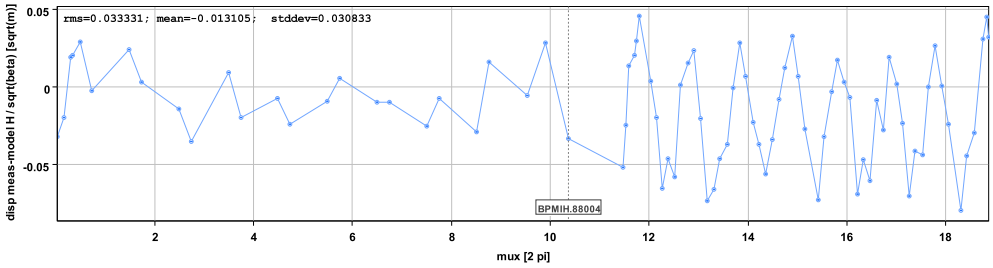
Nevertheless, the problem in this case becomes visible in the vertical dispersion:

[4]In the 2008 MadX models, this strength was denoted by kmqif8760. Again, we use the new denotation throughout this text to avoid confusion.

[5]Data taken: 2008-08-24 03:59



(a) Measured dispersion compared with calculated.



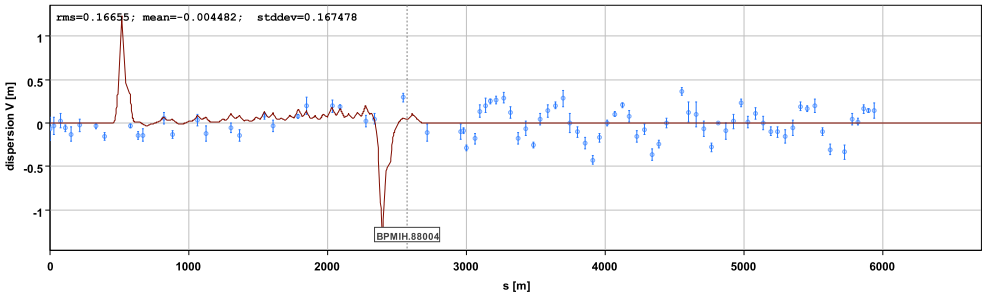
(b) Normalized dispersion difference between measurement and model.

Figure 8.12.: Comparison between measured dispersion with newly rematched optics "ti8-2008-08-23-1957". kqif.87600 reduced by 9% in the machine, not in the model.

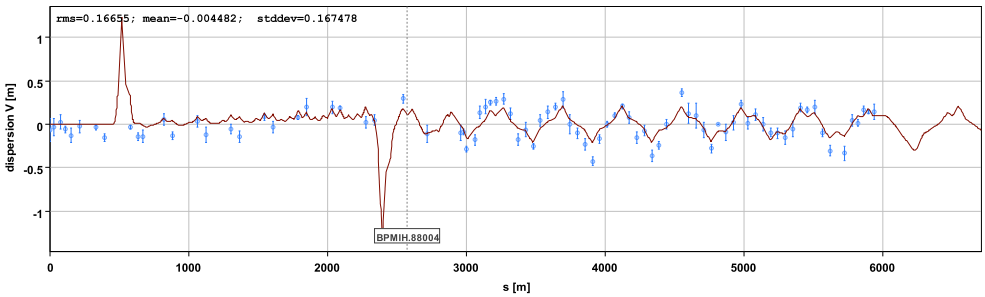
The measurement clearly shows a dispersion wave starting around the TI8-LHC junction which is not reproduced by the model with the nominal value of kqif.87600 (optics "2008-08-23-1957"). On the contrary, the vertical dispersion is better reproduced by a model with the same value for kqif.87600 in the model as in the machine. This is shown in Figs. 8.13.

The second clear indication, why this reduction is not a good correction, is that kick response measurements also indicate a mismatch between measurent and model as demonstrated in Figs. 8.14 for a horizontal corrector[6]. The mismatch between measurement and model starts at the TI8-LHC junction when using the model without reduced kqif.87600 (Fig. 8.14(a)), while the model prediction matches very well the measurement when using the same strength in the model as in the machine (Fig. 8.14(b)).

[6]Data taken: 2008-08-24 23:41 to 2008-08-25 02:52



(a) Optics "ti8-2008-08-23-1957" in model (nominal value of kqif.87600).



(b) Optics "ti8-2008-08-24-0327" in model (kqif.87600 reduced by 9% also in model).

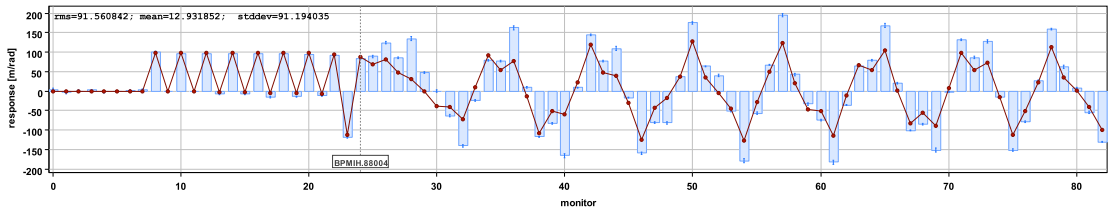
Figure 8.13.: Comparison between measured vertical dispersion with newly rematched optics "ti8-2008-08-23-1957". kqif.87600 reduced by 9% in the machine.

8.2.5. Rematched Optics with kqid.80500 reset to Nominal Values

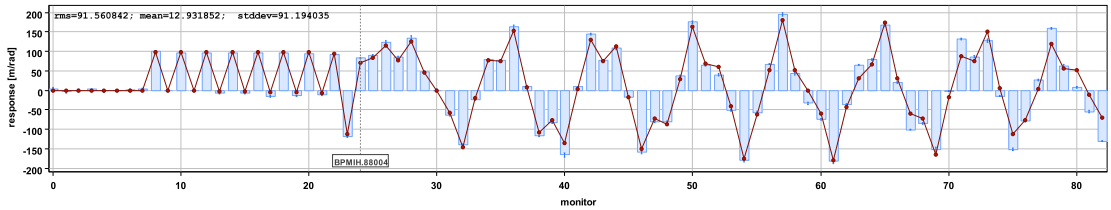
On September 18, 2008 at 10:34 a newly calculated optics was applied to the machine. We will refer to this optics as "ti8-2008-09-1034". The initial conditions are the same as in the optics described above and kqid.80500 was reset to its nominal value and kqif.87600 was not reduced in the machine w.r.t. the model.

8.2.6. Summary

Several attempts were made during 2008 to empirically correct the mismatch between measurement and model for the TI8 transfer line and the adjacent LHC sector 78. The reduction by 0.65% of the main defocussing quadrupole strength in the transferline was promising to reproduce the phase error in the transfer line observed in the vertical kick response data, but could not explain the dispersion mismatch. The reduction of the quadrupole strength kqif.87600 towards the end of the line improved



(a) Optics "ti8-2008-08-23-1957" in model (nominal value of kqif.87600).



(b) Optics "ti8-2008-08-24-0327" in model (kqif.87600 reduced by 9% also in model).

Figure 8.14.: Comparison between measured horizontal corrector response (corrector MCIAH.81604) with newly rematched optics "ti8-2008-08-23-1957". kqif.87600 reduced by 9% in the machine.

the horizontal dispersion, but clearly was in disagreement with both, vertical dispersion and kick-response measurement. So none of these empirical corrections was successful.

8.3. Systematic Fit Analysis

Since no obvious empirical correction could be found, as described in the previous section, we launched a systematic analysis of the available data using the tools described in Chap. 7. Different combinations of model parameters were used to run fits to the measured data. Examples of results shown in the following use a set of kick-response measurement[7] and a corresponding dispersion measurements[8] which were taken with the same optics ("ti8-2008-08-24-0327") in the machine. The model qualities (Eq. (6.33) and Eq. (6.40)) for kick-response and dispersion for the used set of monitors were

$$\Delta^K_{\mathrm{rms}} = 0.61\,[1] \qquad \text{and} \qquad \Delta^D_{\mathrm{rms}} = 0.69\,[1]. \tag{8.1}$$

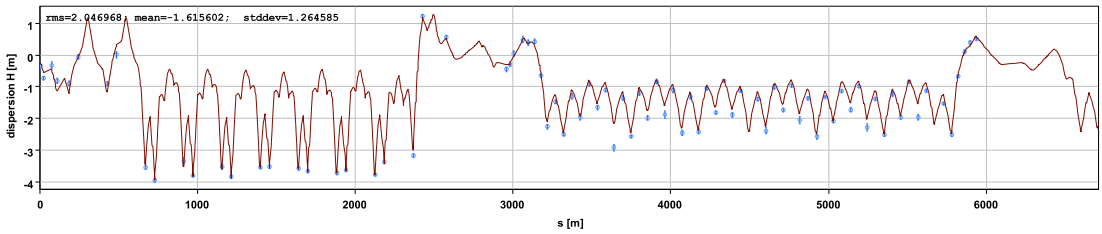
[7]Data taken: 2008-08-24 23:41 to 2008-08-25 02:52
[8]Data taken: 2008-08-24 03:59

8.3.1. All Quadrupoles in the Transfer Line

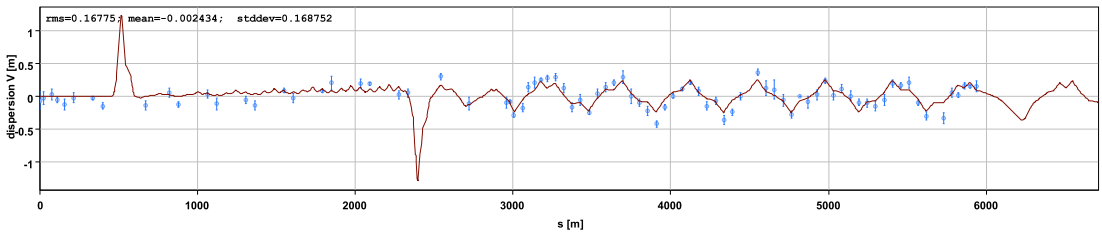
To get a feeling, if and how the measured data can be reproduced at all by the model, we used the strengths of almost all quadrupoles in the TI8 transfer line as individual fit parameters (only quadrupoles which were not constrained by any kick-response measurement, i.e. quadrupoles upstream of MCIAH.80204, were not used). The SVD fit (after one iteration) improves the agreement between measurement and model for both, dispersion and kick-response: The values for the model qualities, Eq. (6.33) and Eq. (6.40), are reduced to

$$\Delta^K_{\mathrm{rms}} = 0.40\,[1] \qquad \text{and} \qquad \Delta^D_{\mathrm{rms}} = 0.45\,[1]. \tag{8.2}$$

This big improvement is very well visible in Figs. 8.15, which show the horizontal and vertical dispersion functions, and Figs. 8.16, which show an example horizontal and vertical corrector response after this fit.



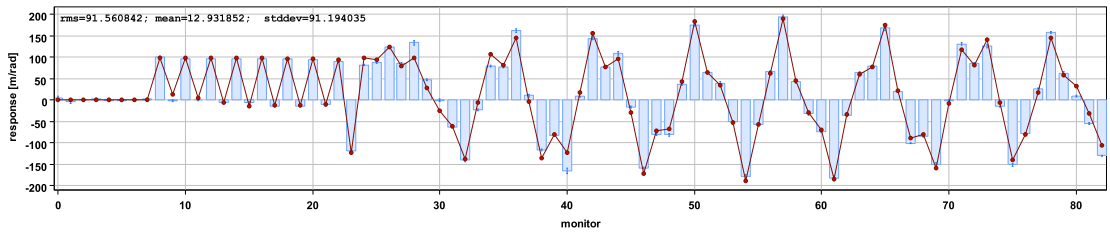
(a) Horizontal dispersion.



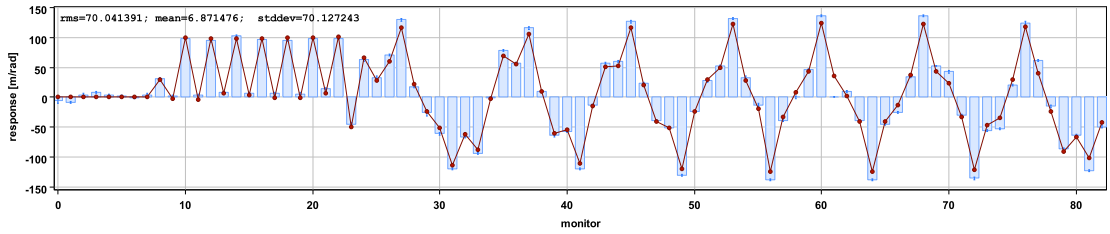
(b) Vertical dispersion.

Figure 8.15.: Comparison between measured and model dispersion after SVD fit with all quadrupole strengths in TI8. Optics "ti8-2008-08-24-0327" in both, machine and model.

The relative changes of the quadrupole strengths during this fit are shown in Fig. 8.17. It is notable that the mean of this changes is clearly positive (about +0.7%). So almost all strengths are increased by this fit. The few strengths which are reduced are mainly strenths of focussing quadrupoles. The largest strength changes are that of the quadrupoles MQID.80500 and MQIF.80600. Nevertheless, fits using only these



(a) Horizontal orbit response of corrrector MCIAH.81604.



(b) Vertical orbit response of corrector MCIAV.81504.

Figure 8.16.: Comparison between measured and model example corrector responses after SVD fit with all quadrupolte strengths in TI8. Optics "ti8-2008-08-24-0327" in both, machine and model.

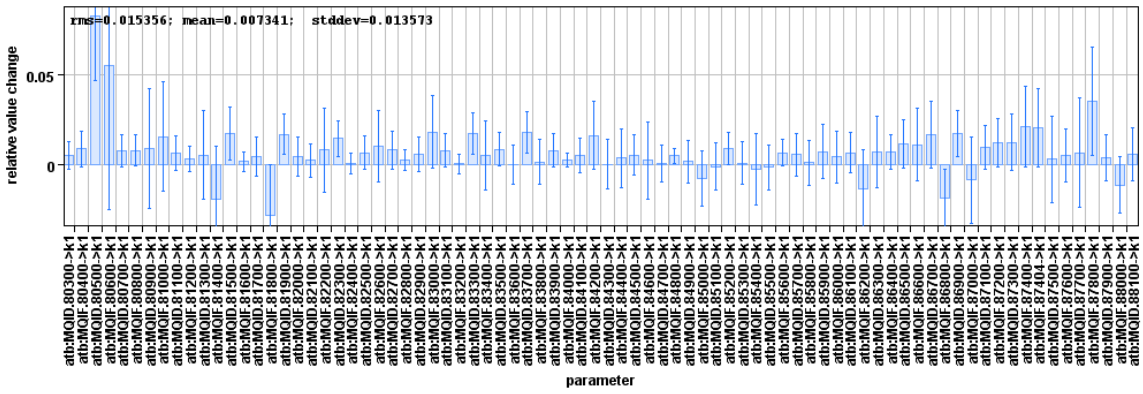


Figure 8.17.: Relative parameter changes for first fit iteration using all quadrupole strengths in TI8 as free parameters.

strengths as parameters cannot reproduce at all the measured behaviour.

8.3.2. Individually Powered Quadrupoles

The corrections as obtained in the previous section are of limited practical use, because not all of the quadrupoles in the transfer line are individually powered. On the contrary, the corrections proposed in the following could (at least in principle) be applied in the real machine, since we will only use separately accessable circuits in TI8.

Assuming a fully linear optics, a corrected value c_{corr} of a machine parameter c could be calculated by

$$c_{\mathrm{corr}} = c_{\mathrm{nom}} - \Delta c, \tag{8.3}$$

where c_{nom} is the nominal value of the parameter and Δc the change of the parameter during the fit as given by

$$\Delta c = c_{\mathrm{fit}} - c_{\mathrm{nom}}. \tag{8.4}$$

Here c_{fit} is the value of the parameter after the fit.

For example, using the main quadrupole chains in TI8 and all individually powered quadrupoles downstream of MCIAH.80204 as fit parameters, one obtains after one SVD iteration for the model qualities, Eq. (6.33) and Eq. (6.40):

$$\Delta^K_{\mathrm{rms}} = 0.39\,[1] \qquad \text{and} \qquad \Delta^D_{\mathrm{rms}} = 0.46\,[1]. \tag{8.5}$$

So also this fit reproduces very well both kick-response and dispersion measurement. The relative strength changes are shown in Fig. 8.18. This fit clearly reproduces the results already discovered earlier [Wen06], namely that the defocussing quadrupole strength in the transfer line (denoted by kmqid in the plot) appears to be too large (This fit gave $0.59 \pm 0.16\%$). The rest of the resulting strengths help to reproduce the dispersion, but are useless because of the large errors, as visible in the plot (errorbars of 100% or larger).

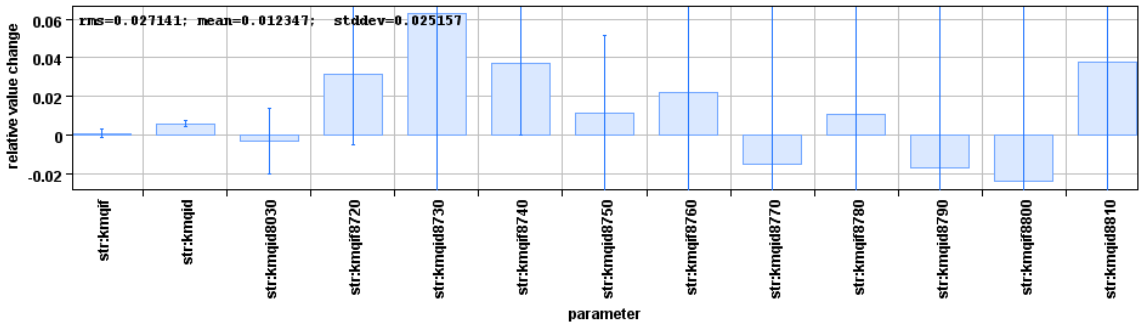


Figure 8.18.: Relative parameter changes after SVD fit using all individually powered quadrupoles in TI8 downstream of MCIAH.80204.

8.3.3. Corrections with Few Quadrupoles

As already mentioned in Sec. 6.3, a simple way to determine the most effective parameters is to use the MICADO algorithm instead of SVD. An example result of such a fit is shown in Table 8.2. The table shows the 3 parameters (ordered by their sensitivity), that were changed by a MICADO fit with 3 internal iterations, using the main quadrupole strengths and all individually powered quadrupoles in TI8 as free parameters. As expected, the most sensitive parameter is again the main defo-

param	initial value [m^{-2}]	sensitivity [1]	fitted value [m^{-2}]	rel difference [%]
kqid.80500	-0.0336	0.6363	-0.0339	0.90
kqif.87200	0.0350	0.0609	0.0364	3.87
kqif.87400	0.0211	0.3028	0.0223	5.55

Table 8.2.: Parameters resulting from a fit using 3 Micado iterations.

cusing quadrupole strength kqid.80500, which compensates for the phase error in the vertical plane. The other two strengths picked by the algorithm, are two focussing quadrupoles with a phase difference of about 90 degrees between them. These two together correct the horizontal dispersion wave starting at the TI8-LHC junction.

Since error propagation is not implemented for the MICADO algorithm in Aloha, we use an SVD fit with only these three strengths as free parameters, to demonstrate the resulting properties. The results for the three parameters are listed in Table 8.3.

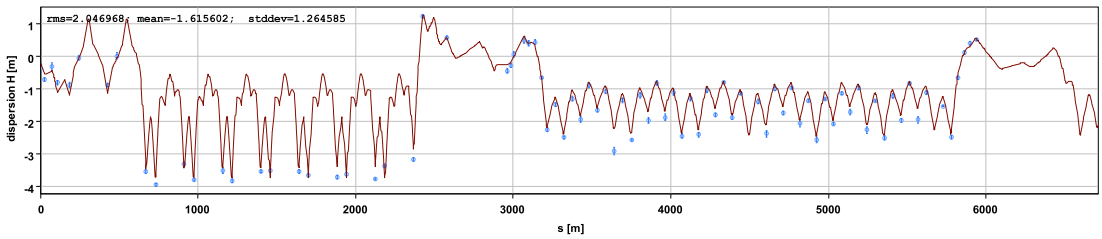
param	initial value [m^{-2}]	fitted value [m^{-2}]	rel difference [%]
kqid.80500	-0.0336	-0.0339	0.91 ± 0.04
kqif.87200	0.0350	0.0366	4.38 ± 1.17
kqif.87400	0.0211	0.0223	5.74 ± 0.75

Table 8.3.: Resulting parameter values from SVD fit with only three parameters.

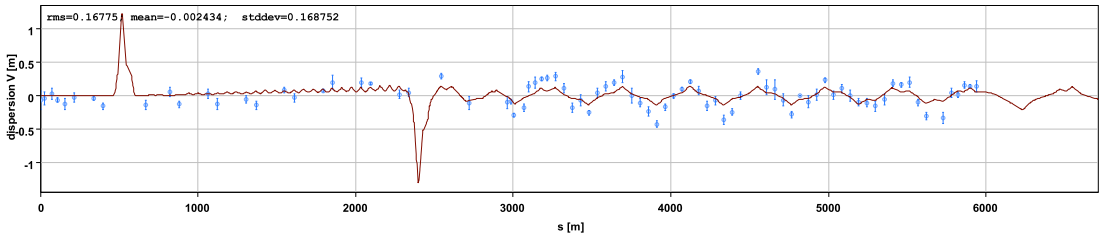
The resulting model qualities (Eq. (6.33) and Eq. (6.40)) are similar to the results of the previous fits (which had many more parameters):

$$\Delta^K_{\mathrm{rms}} = 0.39\,[1] \qquad \text{and} \qquad \Delta^D_{\mathrm{rms}} = 0.49\,[1]. \tag{8.6}$$

Figures 8.19 and Figs. 8.20 illustrate the good agreement between measurements and the resulting model.



(a) Horizontal dispersion.



(b) Vertical dispersion.

Figure 8.19.: Comparison between measured and model dispersion after SVD fit with three parameters in TI8 (see Table 8.3). Optics "ti8-2008-08-23-1957" in both, machine and model.

8.3.4. Summary

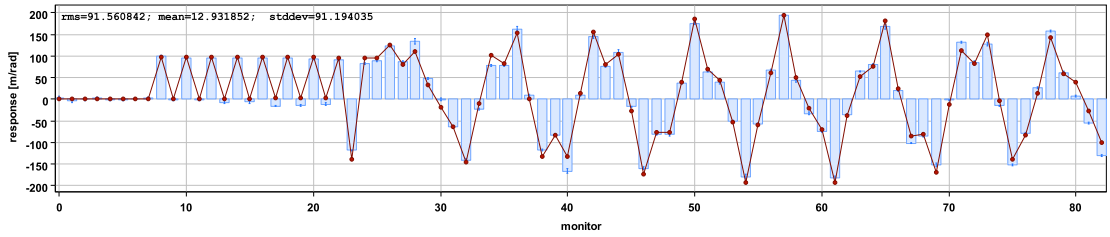
In this section we showed, that we are able to reproduce the measured dispersion- and kick-response data with a model using different sets of parameters. Most notable, even with only three parameters (the main defocussing quadrupole strength and two individually powered quadrupoles towards the end of TI8) the measurements can be nicely reproduced and thus could also be corrected.

Nevertheless, still no isolated sources could be identified by this method and the physical explanations for the observed effects are still missing.

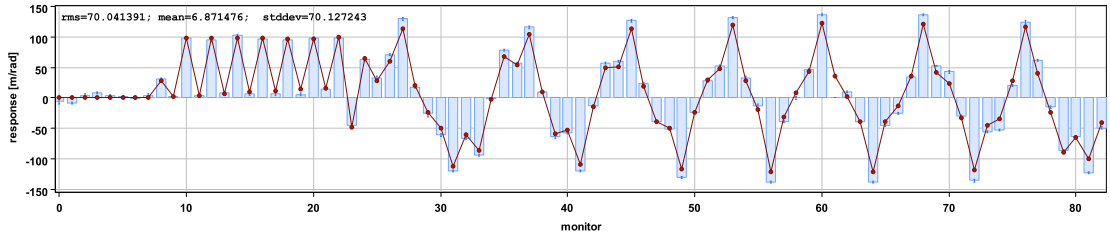
8.4. BPM Gains and Nonlinear Dispersion

The dispersion measurements used in the previous sections were dispersion measurements that were calculated online by the steering tool YASP. Internally, YASP uses linear fits to the position data for several momentum offsets $\frac{\Delta p}{p}$ (which were trimmed in the SPS during the measurement procedure) to estimate the linear dispersion D_u (see Eq. (3.8)). This is sufficient in most cases.

To illustrate the considerations in this section, we will use a dataset of 229 TI8-



(a) Horizontal orbit response of corrrector MCIAH.81604.



(b) Vertical orbit response of corrector MCIAV.81504.

Figure 8.20.: Comparison between measured and model example corrector responses after SVD fit with three parameters in TI8 (see Table 8.3). Optics "ti8-2008-08-23-1957" in both, machine and model.

trajectories[9] recorded at five different values of $\frac{\Delta p}{p}$. Figure 8.21(a) shows a linear fit to this set of data at one BPM. It is well visible that the linear fit can not reproduce the real momentum dependent behaviour of the transverse position, since even in the model, the position clearly shows a nonlinear behaviour. The behaviour can be better reproduced for example by a quadratic fit, as shown in Fig. 8.21(b) for the same BPM and measurement data.

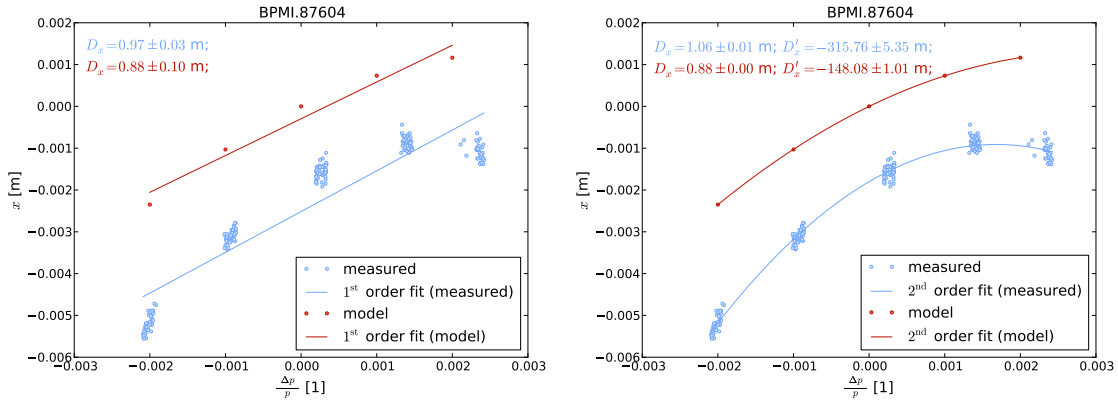
Having this in mind, we define the second order dispersion D'_u as

$$D'_u = \frac{1}{2} \frac{\partial^2 u}{\partial \left(\frac{\Delta p}{p}\right)^2}. \tag{8.7}$$

It shall be noted, that the factor $\frac{1}{2}$ is uncommon compared to the usual definition. Nevertheless, this definition is consistent e.g. with the treatment in MadX and is convenient, since this way D'_u directly represents the second order coefficient for the transverse position,

$$u = u_\beta + D_u \frac{\Delta p}{p} + D'_u \left(\frac{\Delta p}{p}\right)^2, \tag{8.8}$$

[9]Data taken: 2009-06-06 13:11 to 16:24



(a) Linear fit.

(b) Quadratic fit.

Figure 8.21.: Model and measured positions for an example BPM (BPMI.87604) in the TI8 transfer line.

and therefore can be read off directly from polynomial fits.

8.4.1. Correction of Measurement Data

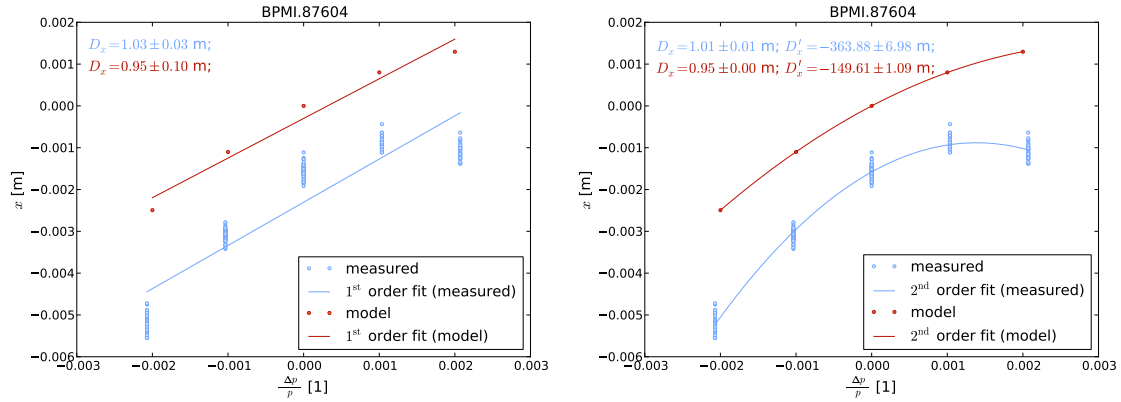
When we started looking into nonlinear dispersion, it turned out that there were other effects, that contributed to the measurements, which had to be investigated beforehand. These will be sketched in the following.

Adjustments of Momentum Trims

The values for $\frac{\Delta p}{p}$ used e.g. in Figs. 8.21 were the values which were calculated by YASP from the measured orbits. To estimate the $\frac{\Delta p}{p}$, YASP uses the dispersion values at the monitors from the nominal model (i i denotes the monitor index):

$$\frac{\Delta p}{p} = \frac{\sum_i x_i {D_x}^i}{\sum_i ({D_x}^i)^2}. \tag{8.9}$$

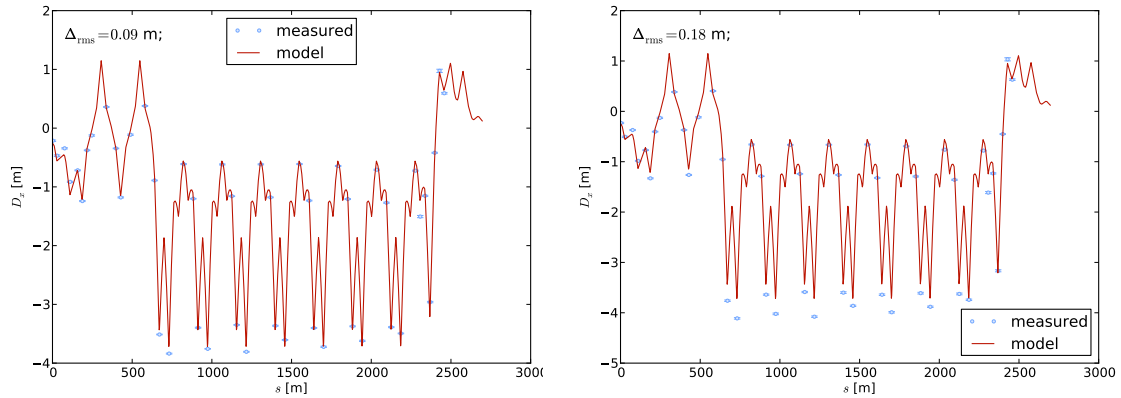
For that reason, this estimate is already biased and forces the measured dispersion towards the model one. To obtain a measurement which is independet of the nominal model dispersion it was necessary to use the 'real' values for $\frac{\Delta p}{p}$. As the real values we consider the $\frac{\Delta p}{p}$ values which can be calculated the frequency trims in the SPS by the use of Eq. (4.8). The calculated values were forced either by entering them manually in YASP or forcing them by the analysis scripts to certain values. Figures 8.22 show the effect of this forcing of the $\frac{\Delta p}{p}$ values on the data for the same BPM as Figs. 8.21.



(a) Linear fit. (b) Quadratic fit.

Figure 8.22.: Model and measured positions for an example BPM (BPMI.87604) in the TI8 transfer line with $\frac{\Delta p}{p}$ values forced to trimmed ones.

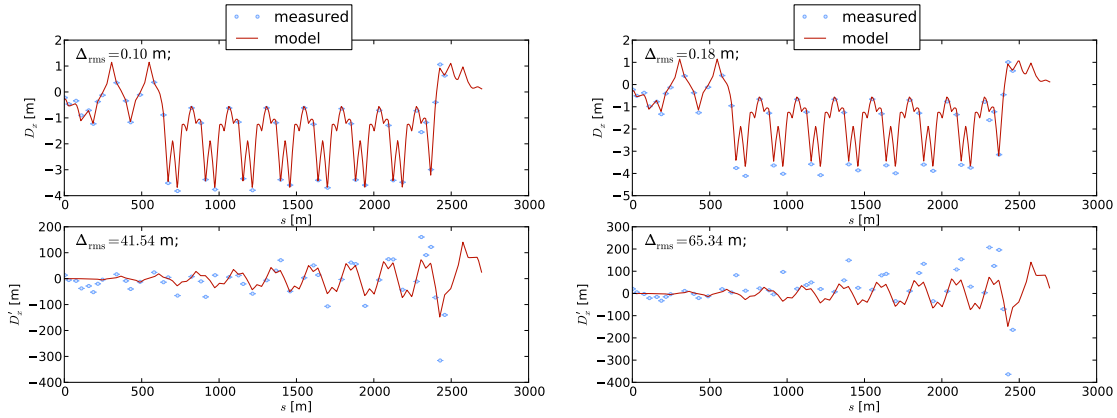
Figure 8.23(a) shows the dispersion in TI8 calculated by linear fits using the $\frac{\Delta p}{p}$ values estimated by YASP, while Fig. 8.23(b) shows the same dataset with the $\frac{\Delta p}{p}$ values forced to the values calculated from the frequency trims. It turned out, that this (in principle more accurate) calculation of the dispersion worsened the agreement between measurement and model. The model quality changes from $\Delta_{\mathrm{rms}} = 0.09\,\mathrm{m}$ to $\Delta_{\mathrm{rms}} = 0.18\,\mathrm{m}$.



(a) $\frac{\Delta p}{p}$ values estimated by YASP (b) $\frac{\Delta p}{p}$ values forced to trimmed ones.

Figure 8.23.: Horizontal dispersion in TI8 calculated from linear fits to position data.

Figures 8.24 show the same comparison for quadratic fits. Here also the agreement for the second order dispersion gets visibly worse.



(a) $\frac{\Delta p}{p}$ values estimated by YASP (b) $\frac{\Delta p}{p}$ values forced to trimmed ones.

Figure 8.24.: Horizontal dispersion in TI8 calculated from linear fits to position data.

Monitor Gains

When looking closer into nonlinear effects, it becomes more and more important to have reliable measurement data. Clearly, for dispersion and kick-response data which we use in our discussions we rely on the quality of the BPMs. To check the BPM gains we used fits in Aloha with the BPM gains as additional free parameters.

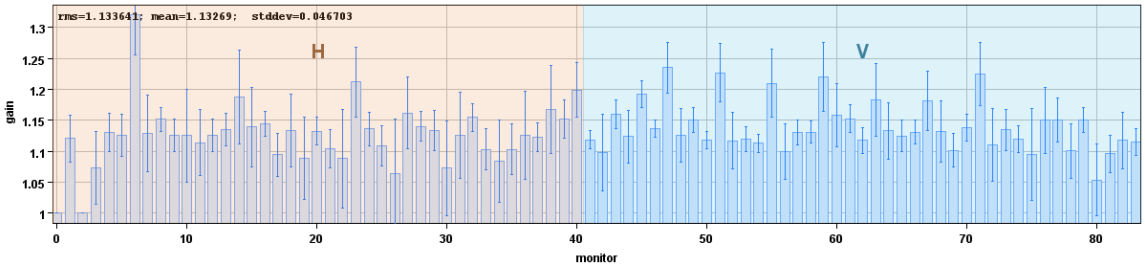


Figure 8.25.: BPM gains for TI8, resulting from a fit using BPM gains and main quadrupole strengths as free parameters.

Figure 8.25 shows the result of such a fit. This plot shows the monitor gains obtained from fits to kick-response data with 25 correctors for TI8 data[10] taken at the day

[10]Data taken: 2009-06-07 08:50 to 09:40

as the previously discussed dispersion data. The corrector gains were assumed to be perfect and were therefore not varied during this fit. The main quadrupole strengths of the line were used to compensate for the phase errors. The mean of the corrector gains h_i obtained from this fit is about

$$\langle h \rangle = 1.13, \tag{8.10}$$

i.e. systematically too high.

Detailed investigations of this issue revealed that the observed errors were due to the fact that the BPM frontends did not take into account nonlinear errors, which are resulting from the BPM geometry and electronics nonlinearities. During the measurement, only one global scaling factor was applied. This setup resulted in an average scaling factor of approximately 1.1 for beam excitations within the range of $\pm 10\,\mathrm{mm}$ [Jon09].

The electronics linearization correction, which was not implemented in the frontends at that time, is given by

$$u_3 = K_f \left((1 - A)\, u_2 + A\, u_2^3 \right). \tag{8.11}$$

Here u_3 denotes the corrected position for one transverse plane (x or y) and u_2 denotes the raw position, as delivered by the BPM electronics. K_f denotes a global linear factor which for the transfer line BPMs is given by $K_f = 15.35$. A is the electronics linearization coefficient which for TI8 is 0.139 for high sensitivity (low beam intensity) or 0.159 for low sensitivity (high beam intensity).

When recalibrating old measurement data, u_0 can be calculated by

$$u_2 = \frac{u_1}{K_f^1}, \tag{8.12}$$

where u_1 is the (wrongly) measured position and K_f^1 denotes the global correction factor which was in place at the time of measurement. Its value is given by $K_f^1 = 15.77$ for all the data used in this section.

The geometrical correction, which also was not applied, is given by

$$x = 1.08 \cdot 10^{-5} x_3^5 + 8.0 \cdot 10^{-6} x_3^3 + 1.033 x_3 + 3.8 \cdot 10^{-6} x_3^3 y_3^2 + 6.9 \cdot 10^{-6} x_3 y_3^4 \tag{8.13a}$$
$$y = 1.08 \cdot 10^{-5} y_3^5 + 8.0 \cdot 10^{-6} y_3^3 + 1.033 y_3 + 3.8 \cdot 10^{-6} y_3^3 x_3^2 + 6.9 \cdot 10^{-6} y_3 x_3^4. \tag{8.13b}$$

Here x and y are the real positions, while x_3 and y_3 are the intermediate results as calculated by Eq. (8.11). The dependency of the corrected (real) positions on the uncorrected (wrongly measured) positions is plotted for selected x_1-y_1 combinations in Fig. 8.26. Figures 8.27 illustrate the difference between the corrected and the uncorrected positions.

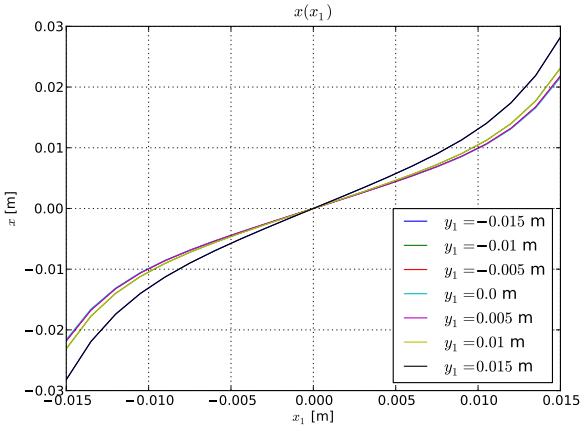
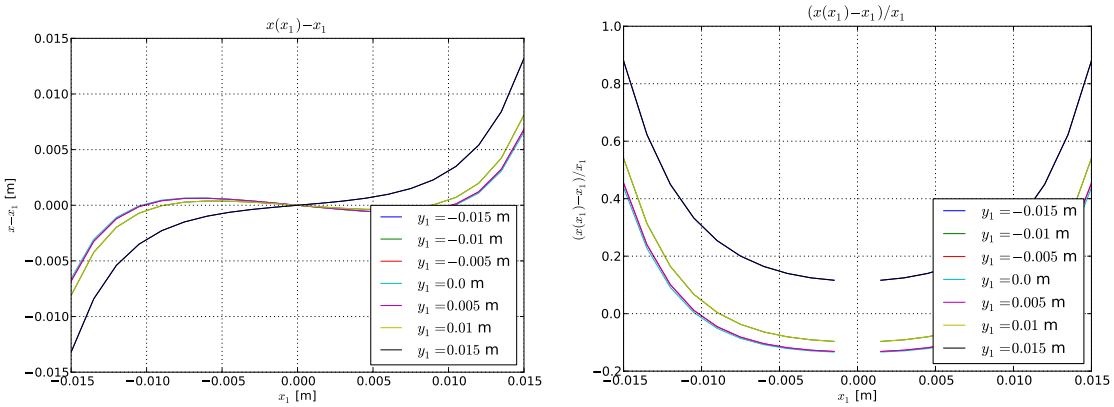


Figure 8.26.: Dependence of real beam position x on uncorrected position x_1.



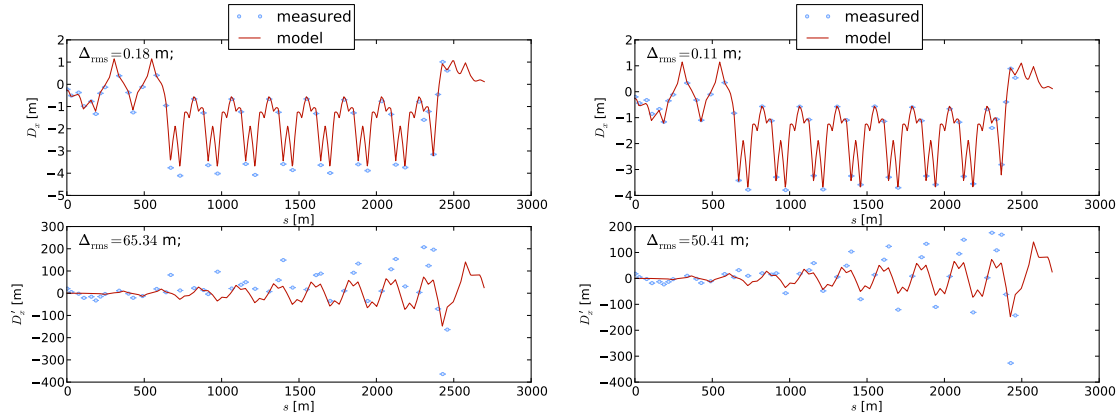
(a) Difference.

(b) Relative difference.

Figure 8.27.: Differences between corrected and uncorrected positions.

To demonstrate the influence of these corrections we use the same dispersion data than in the previous section. Figures 8.28 show a comparison between quadratic fits to the dispersion raw data, once only with the forced $\frac{\Delta p}{p}$ values to the trimmed ones, without BPM calibration (Fig. 8.28(a), which is the same plot as Fig. 8.24(b)) and once BPM data (Fig. 8.28(b)), posprocessed using Eqs. (8.13). The model quality is improved from $\Delta_{\text{rms}} = 0.18\,\text{m}$ to $\Delta_{\text{rms}} = 0.11\,\text{m}$ for the linear dispersion and from $\Delta_{\text{rms}} = 65\,\text{m}$ to $\Delta_{\text{rms}} = 50\,\text{m}$ for the second order dispersion, which additionaly is better centered in the latter case.

Inserting the values obtained from the magnet simulations and fits (Eq. (8.39)) into the model results in a very good model-measurement agreement of Δ_{rms}=0.10 m for



(a) Measurement with uncalibrated BPMs. (b) Data calibrated after measurement.

Figure 8.28.: Influence of BPM calibration on horizontal dispersion in TI8.

the linear dispersion and Δ_{rms}=22 m for the second order dispersion, as illustrated in Fig. 8.29.

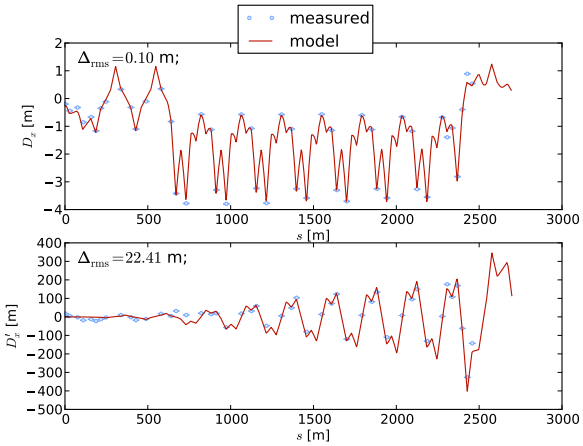


Figure 8.29.: Horizontal dispersion in TI8. BPM data calibrated, model with values from Eq. (8.22).

8.5. Higher Order Field Components in the Injection Main Bends

8.5.1. Introduction

Since no isolated sources could be identified using the systematic analysis described in Sec. 8.3, other sources for the errors had to be considered. It was suggested [Far08] that higher order field components in the transfer line magnets could explain some of the observed effects. This could be checked by measuring the chromaticity of the transfer line. This proposal lead to the measurements described in this section.

8.5.2. Conventions

In the following we will only refer to the field components of the vertical (i.e. horizontally bending) field. We define the relations for the bending of the beams as in [Hin08, p.120], so a positive vertical field bends a positive particle to the right. For a particle with the horizontal position x the field B_y can be expressed by the nominal field B_1 and higher order field errors b_n as follows (up to second order):

$$B_y = B_1 \left(1 + b_2 \frac{x}{{R_{\text{ref}}}} + b_3 \frac{x^2}{{R_{\text{ref}}}^2}\right), \tag{8.14}$$

where R_{ref} is a reference radius which is defined for the main bends in the transfer lines as $R_{\text{ref}} = 0.025\,\text{m}$.

For further reference we calculate also the field seen by a trajectory with a systematic offset in the main bends. For a trajectory offset ($x \to x + \Delta x$) this becomes

$$\begin{aligned} B_y &= B_1 \left(1 + b_2 \frac{x + \Delta x}{{R_{\text{ref}}}} + b_3 \frac{(x + \Delta x)^2}{{R_{\text{ref}}}^2}\right) = \\ &= B_1 \left(1 + \tilde{b}_1 + \tilde{b}_2 \frac{x}{{R_{\text{ref}}}} + \tilde{b}_3 \frac{x^2}{{R_{\text{ref}}}^2}\right) \end{aligned} \tag{8.15}$$

with the off-center higher order field components

$$\tilde{b}_1 = b_2 \frac{\Delta x}{{R_{\text{ref}}}} + b_3 \frac{\Delta x^2}{{R_{\text{ref}}}^2} \tag{8.16a}$$

$$\tilde{b}_2 = b_2 + b_3 \frac{2\Delta x}{{R_{\text{ref}}}} \tag{8.16b}$$

$$\tilde{b}_3 = b_3. \tag{8.16c}$$

While the magnets in TI2 bend the proton beam to the right, the beam in TI8

Transfer line	Field	radius	angle	bend in beam direction
TI2	$B_1 > 0$	$\rho_0 > 0$	$\alpha > 0$	to the right
TI8	$B_1 < 0$	$\rho_0 < 0$	$\alpha < 0$	to the left

Table 8.4.: Convention for bending magnets. B_1 denotes the dipole field component in y-direction.

is bent to the left. Therefore, the nominal dipole field in TI2 is positive, while it is negative in TI8. All these conventions are summarized in Table 8.4 for the two transfer lines.

8.5.3. Chromaticity Measurement

The natural chromaticity for a ring is defined as

$$\Delta Q_u = {Q'_u}^{\text{ring}} \frac{\Delta p}{p}, \tag{8.17}$$

where u indicates one of the planes x or y, ΔQ_u is the tune change in the respective plane and ${Q'_u}^{\text{ring}}$ is the natural chromaticity. Correspondig to this we define the natural chromaticity of a transfer line as

$$\Delta \mu_u = {Q'_u}^{\text{tl}} \frac{\Delta p}{p}, \tag{8.18}$$

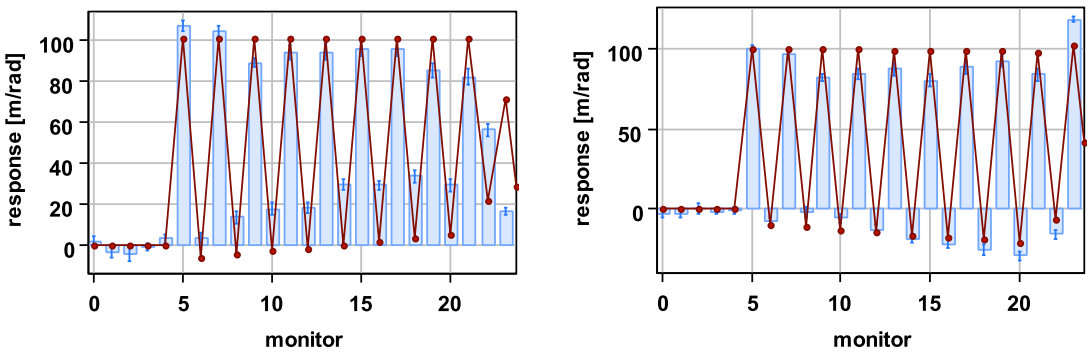
where $\Delta\mu_u$ denotes the change of the phase advance at the end of the tranfer line in the plane u.

As visible from Eq. (8.17), measuring the chromaticity in a ring can be accomplished by changing the beam energy w.r.t. the magnetic fields and recording the corresponding tune change. In a transfer line, where the parameter corresponding to the tune in a ring is the betatron phase advance, no direct measurement is possible. Therefore a different approach had to be developed. We will derive the chromaticity indirectly, by measuring corrector responses at different beam momenta and fit the model to this measured data with phyiscally meaningful model strengths as free parameters. The phase advance at the end of the line (and thus the chromaticity) can then be read off directly from the resulting model.

The data used in the following, consists of data sets of 4 corrector responses for each plane. The correctors were excited by $\pm 40\,\mu$rad with respect to their nominal setting. The measurement was repeated for 7 different values of the initial energy

offset $\frac{\delta p}{p}$ at the exit of the SPS (-2, -1, -0.5, 0.0, +0.5, +1 and +2 permill)[11].

Already from the raw measurement data it is clear that the horizontal response depends strongly on $\frac{\delta p}{p}$ (Fig. 8.30), while the vertical response does not (Fig. 8.31). The model predicts dependencies on $\frac{\delta p}{p}$ in both planes (red lines in Fig. 8.30 and Fig. 8.31).



(a) $\frac{\delta p}{p} = -0.002$ (b) $\frac{\delta p}{p} = +0.002$

Figure 8.30.: Horizontal response for one horizontal corrector. Bars represent the measured data, dots the nominal model. A clear dependence on $\frac{\delta p}{p}$ is visible, which is only partly reproduced by the model.

Model of Momentum Dependence

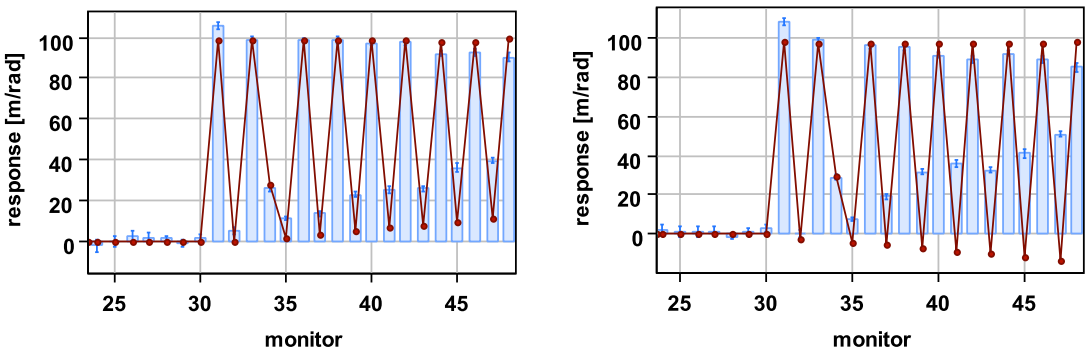
The influence of systematic quadrupolar and sextupolar field errors in the main bends of the transfer line on the response matrix can be expressed by

$$\Delta R_{ij} = A_{ij}\left(\frac{\Delta K}{K} - \frac{\Delta p}{p}\right) + B_{ij}b_2 + C_{ij}b_3\frac{\Delta p}{p}. \tag{8.19}$$

$\frac{\Delta K}{K}$ denotes a systematic error of the main quadrupole strengths with respect to the nominal settings. b_2 and b_3 denote the systematic relative quadrupolar and sextupolar field errors in units of 10^{-4} with respect to the main field of the bend. These were implemented directly in the MadX model of the transfer line by the use of

$$k_{n-1} = \frac{b_n}{R_{\text{ref}}^{n-1}}\frac{\alpha}{l}10^{-4}(n-1)!, \tag{8.20}$$

[11]Data taken: 2008-09-05 about 21:00 to 24:00



(a) $\frac{\delta p}{p} = -0.002$ (b) $\frac{\delta p}{p} = +0.002$

Figure 8.31.: Vertical response for one vertical corrector. Bars represent the measured data, dots the nominal model. Almost no difference in the measured data is visible, while there should be a dependence on $\frac{\delta p}{p}$ according to the model.

where k_n denote the multipole strengths applied to the main bends in MadX, α is the bending angle of the magnet and l denotes its length. The "true" momentum mismatch $\frac{\Delta p}{p}$ is given by

$$\frac{\Delta p}{p} = \frac{\delta p}{p} + \frac{\Delta p_0}{p}, \tag{8.21}$$

where $\frac{\delta p}{p}$ denotes the trimmed momentum offset and $\frac{\Delta p_0}{p}$ an a priori unknown initial momentum error.

There are four degrees of freedom which have to be determined by the fit algorithm: $\frac{\Delta p_0}{p}$, $\frac{\Delta K}{K}$, b_2 and b_3. The factors A_{ij}, B_{ij} and C_{ij} in Eq. (8.19) correspond to the fit gradients which are calculated implicitely from Eq. (6.23). B_{ij} and C_{ij} act with the same sign as A_{ij} in the horizontal plane and with the opposite sign than A_{ij} in the vertical one.

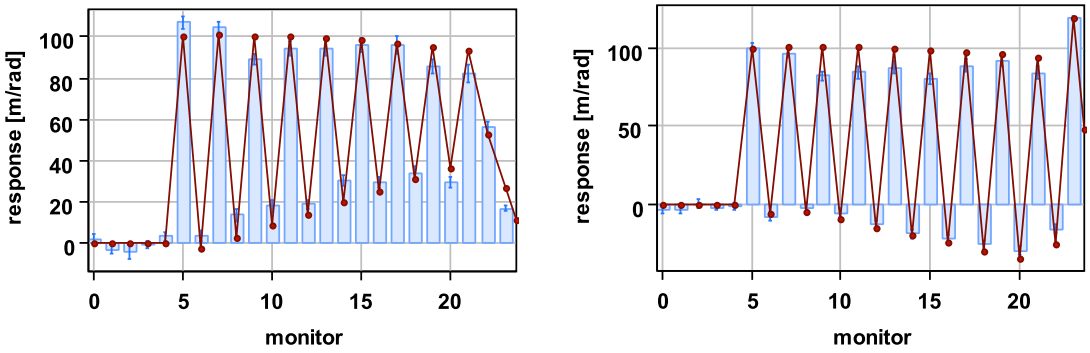
Fit Analysis

The measured data was analysed using Aloha, heavily profiting from the feature to combine measurement data taken at different machine conditions into one large sensitivity matrix. In this case different models were used for different momentum offsets $\frac{\delta p}{p}$ in the machine. This allowed to fit for all four parameters simultaneously.

First fits lead to the result, as published in [FFG+09] of

$$\begin{aligned} b_2 &= 0.85, & b_3 &= -5.06, \\ \frac{\Delta p_0}{p} &= 7.51 \times 10^{-4}, & \frac{\Delta K}{K} &= 5.61 \times 10^{-3}. \end{aligned} \tag{8.22}$$

The fit error as defined by Eq. (6.33) results in $\Delta_{\mathrm{rms}} = 1.99$, which is small compared to the initial values between 5 and 8 for the different measurements. The b_3 was very well reproduced by various fit variants while the other three parameters can only be determined up to a constant as will be discussed in Sec. 8.5.5. Applying the values of Eq. (8.22) to the model results in a very good agreement between model and measurement as demonstrated in Fig. 8.32 and Fig. 8.33.



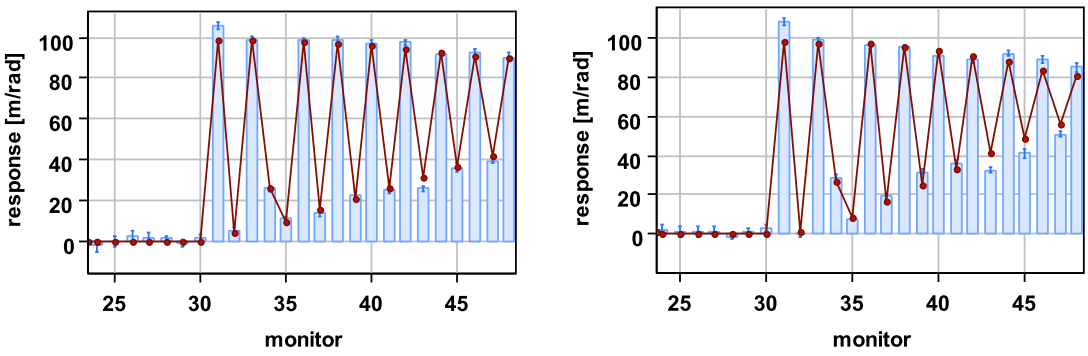
(a) $\frac{\delta p}{p} = -0.002$ (b) $\frac{\delta p}{p} = +0.002$

Figure 8.32.: Horizontal response for the same corrector as in Fig. 8.30 with the values of Eq. (8.22) applied to the model. Bars represent the measured data, dots the updated model.

Chromatic Behaviour

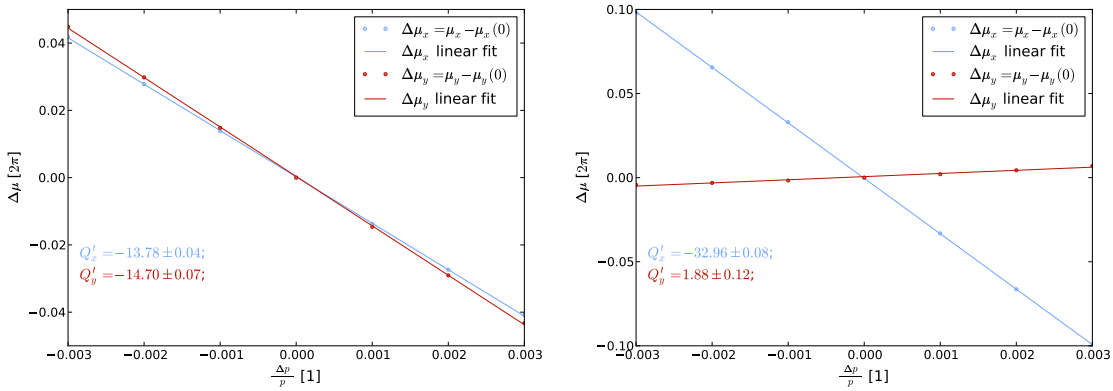
After performing magnetic simulations (see Sec. 8.5.4) the values for b_2 and b_3 were set to the values given in Eq. (8.39) for the new models. Figures 8.34 and Figure 8.35 show comparison of the chromaticities of the original models and the models with these new values. In these figures the slopes represent the natural chromaticities of the lines which in the original model is similar for both planes ($Q'_x = -13.8$, $Q'_y = -14.7$ for TI8 and $Q'_x = -16.1$, $Q'_y = -15.7$ for TI2).

The influence of the b_2 and b_3 on the horizontal and vertical phases μ_x and μ_y is



(a) $\frac{\delta p}{p} = -0.002$ (b) $\frac{\delta p}{p} = +0.002$

Figure 8.33.: Vertical response for the same corrector as in Fig. 8.31 with the values of Eq. (8.22) applied to the model. Bars represent the measured data, dots the updated model.



(a) Original model. (b) Model with values from Eq. (8.39).

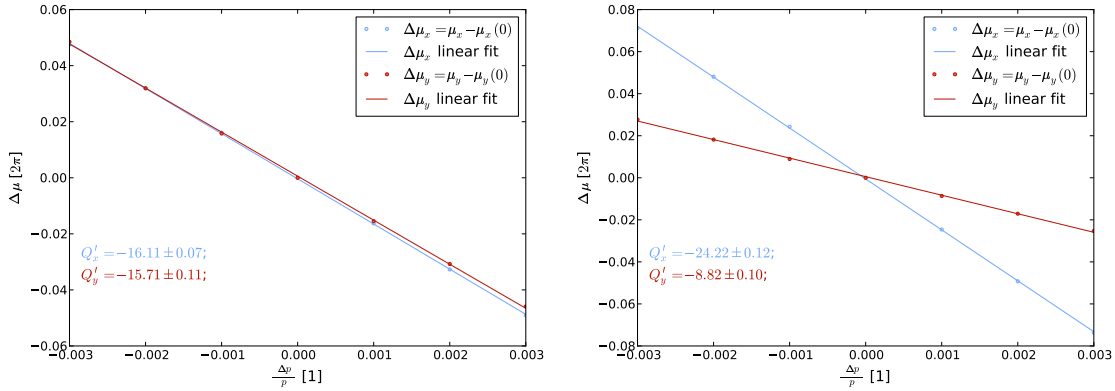
Figure 8.34.: Comparison of natural chromaticity for TI8 (Note: The scales of the vertical axes are different in the two plots.).

given by

$$\Delta\mu_x = Q'_x \left(\frac{\Delta p}{p} - \frac{\Delta K}{K} \right) + A_{2x} b_2 + A_{3x} b_3 \frac{\Delta p}{p}, \tag{8.23a}$$

$$\Delta\mu_y = Q'_y \left(\frac{\Delta p}{p} - \frac{\Delta K}{K} \right) - A_{2y} b_2 - A_{3y} b_3 \frac{\Delta p}{p}. \tag{8.23b}$$

A_{2x}, A_{2y}, A_{3x} and A_{3y} are positive factors which are calculated from the model. Therefore b_2 and b_3 act on the two planes with the opposite signs while the natural



(a) OriginalModel (b) Model with values from Eq. (8.39).

Figure 8.35.: Comparison of natural chromaticity for TI2 (Note: The scales of the vertical axes are different in the two plots.).

chromaticities (Q'_x, Q'_y), which are negative quantities, and $\frac{\Delta K}{K}$ act on both planes with the same sign.

In TI8, the b_3 approximately compensates the vertical natural chromaticity and results in a total vertical chromaticity of $Q'_y = +1.9$ and total horizontal chromaticity of $Q'_x = -33.0$ which is about twice the natural one (Fig. 8.34(b)). This explains the phase behaviour already qualitatively demonstrated by Fig. 8.30 and Fig. 8.31. For TI2 the effect is less drastic (as expected, because of fewer main bends in the line): The resulting chromaticities are $Q'_x = -24.2$ and $Q'_y = -8.8$ (Fig. 8.35(b)).

Flat Top Dependence

One of the remaining questions was the source of these higher order components, derived in the previous sections. Since the transfer line magnets are cycled in sync with the SPS cycle, one suspicion was, that these field errors might come from eddy currents, resulting from reaching the flat top of the cycle. To test this hypothesis, we performed similar measurements for different settings for the delay time of the extraction kicker after reaching top energy in the SPS.

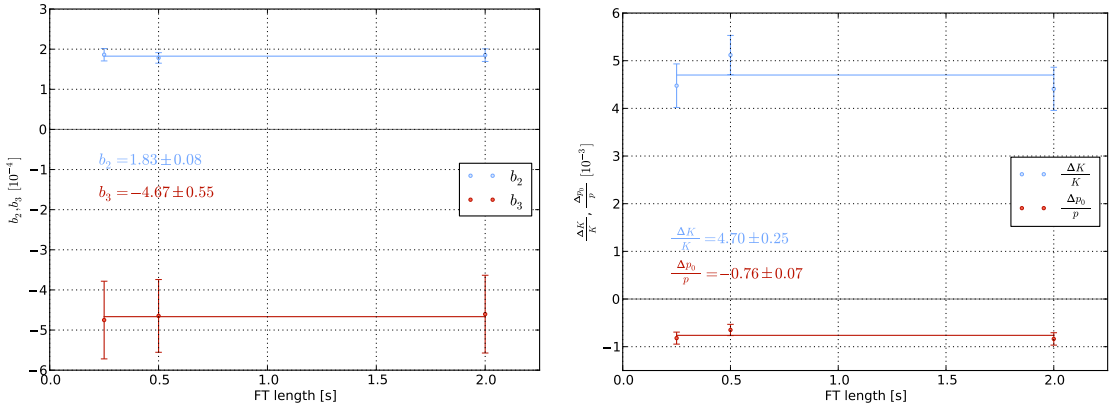
We took data with three different datasets for the delay times of 0.25, 0.5 and 2.0 seconds. For each of the three measurements two correctors per plane were used for which the response was measured at five different values of $\frac{\delta p}{p}$ (-2.2, -1.1, 0.0, +1.1 and +2.2 permille).

Each of these datasets was analysed separately, using the same procedures as described in Sec. 8.5.3. The results of these fits are collected in Table 8.5 and plotted in Figs. 8.36. No significant dependence of b_2 or b_3 on the length of the flat top can

FT-length [s]	b_2	b_3	$\frac{\Delta K}{K}$	$\frac{\Delta p_0}{p}$
0.25	1.86	-4.75	$4.48 \cdot 10^{-3}$	$-8.20 \cdot 10^{-4}$
0.5	1.78	-4.65	$5.12 \cdot 10^{-3}$	$-6.49 \cdot 10^{-4}$
2.0	1.85	-4.60	$4.41 \cdot 10^{-3}$	$-8.37 \cdot 10^{-4}$

Table 8.5.: Results of off-momentum kick-response fits for different flat top lengths.

be deduced from these measurements. From this it was clear that a different source had to be responsible for the systematic b_2 and b_3 errors.



(a) b_2, b_3 (b) $\frac{\Delta p_0}{p}$, $\frac{\Delta K}{K}$

Figure 8.36.: Dependence of fit results on flat top length.

Cross-Check with LHC Main Bends

As a test of the methods used for the transfer lines, we analyzed LHC data in a similar way. At that time no b_2 correction was in place in the LHC. No off-momentum kick response measurements were done, so only a fit for b_2 is possible and not for b_3. Since the b_2 in the LHC have different signs for the different apertures the fit was done with one b_2 parameter per sector. Example results for such a fit on beam 1 data are shown in Table 8.6.

The model data quoted in the table is field error data as produced by the WISE simulation tool which uses a combination of measured magnet data and statistical simulations as replacement for missing information [HGK+04].

The fit was done with the parameters listed in the table, plus a systematic detuning of the quadrupoles as parameter which resulted in $\frac{\Delta K}{K} = 2.31 \cdot 10^{-4}$. The sign and

	b_2 model	b_2 fitted	diff	rel diff
Sector 12	-1.49	-1.68	-0.20	13.23%
Sector 23	1.46	1.40	-0.06	-4.08%
Sector 34	1.35	1.67	0.32	23.46%
Sector 45	1.31	1.44	0.13	10.12%
Sector 56	-1.05	-1.75	-0.70	66.57%
Sector 67	-1.23	-1.57	-0.33	27.16%
Sector 78	-1.06	-1.20	-0.15	13.71%
Sector 81	1.30	1.30	0.00	-0.01%

Table 8.6.: Comparison of fitted and model b_2 in LHC main bends (Model data courtesy of M. Alabau).

the order of magnitude of the b_2 is nicely reproduced by the fit, although some big deviations from the model are visible (e.g. sector 56, almost 67 %).

8.5.4. Magnet Simulations

So far, the values for the higher order field components (b_2 and b_3) were only determined by fits from beam measurements. To cross-check these values, a 2D simulation of the magnets was made by the magnet group in summer 2009 [Bau09]. In the following we show the results of these simulations for the MBIs in TI8, assuming a nominal current of 5270 A, which corresponds to a nominal field of $B_1 = -1.8025\,\mathrm{T}$.

The vertical field along the horizontal axis of the magnet can be fitted by a polynomial fit up to 15^{th} order

$$B_y = \sum_{n=1}^{15} B_n \frac{x^n}{{R_{\mathrm{ref}}}^{n-1}}, \tag{8.24}$$

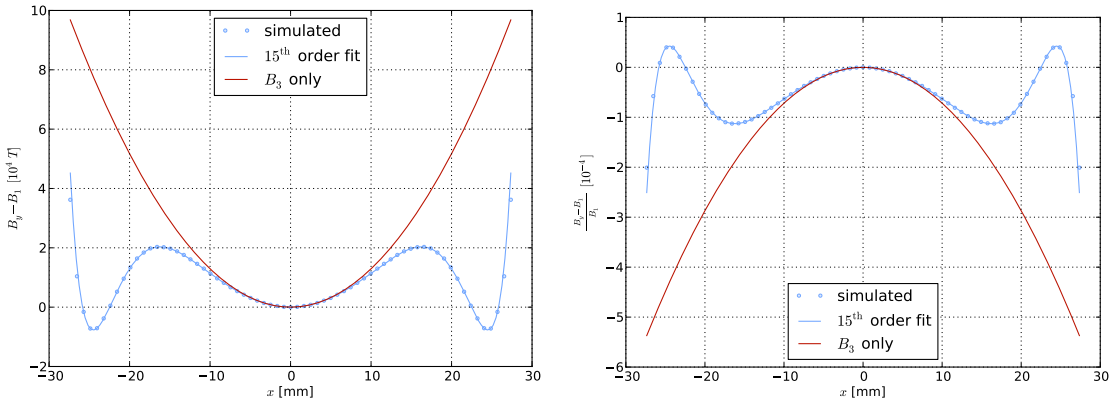
where B_n denotes the absolute higher order field given by $B_n = B_1 b_n$ and R_{ref} the reference radius. The values for B_n from the polynomial fit are listed in Table 8.7.

The simulated data is plotted together with the polynomial fit in Fig. 8.37(a). The same plot also shows the polynomial only taking into account the sextupolar field (B_3). It is nicely visible that in the horizontal range of $\pm 10\,\mathrm{mm}$ the field error is clearly dominated by the sextupolar field. This the range where practically all our measurements took place (kicks of $\pm 40\,\mu\mathrm{rad}$ result in beam oscillations with amplitudes of about $\pm 4\,\mathrm{mm}$ in both transfer lines.). Figure 8.37(b) shows the relative difference w.r.t. the nominal field B_1 in units of 10^{-4}.

From Fig. 8.37(b) we also can read off a value for b_3 at 25 mm, which we used as

B_n	value	B_n	value
B_1	-18025.16	B_2	0.0
B_3	1.29	B_4	0.0
B_5	-0.17	B_6	0.0
B_7	$53.07 \cdot 10^{-3}$	B_8	0.0
B_9	$-42.41 \cdot 10^{-3}$	B_{10}	0.0
B_{11}	$9.26 \cdot 10^{-3}$	B_{12}	0.0
B_{13}	$-0.98 \cdot 10^{-3}$	B_{14}	0.0
B_{15}	$53.72 \cdot 10^{-6}$		

Table 8.7.: Coefficients of of B_y in MBIs from polynomial fit to 2D simulated data in units of 10^{-4} T for $R_{\text{ref}} = 10\,\text{mm}$.



(a) difference (b) relative difference

Figure 8.37.: Differences between simulated and nominal vertical field in TI8 MBIs.

reference radius in all our fits in Sec. 8.5.3, of

$$b_3 = -4.7 \cdot 10^{-4} \quad \text{for } R_{\text{ref}}{=}25\,\text{mm}. \tag{8.25}$$

This corresponds very nicely to our findings in Sec. 8.5.3.

8.5.5. Fit Dependence on Momentum Offset

The four parameters used in Sec. 8.5.3 can only be determined up to a constant ε since they are not completely decoupled. The fit procedure is insensitive to the

transformation

$$\begin{aligned} \frac{\Delta p_0}{p} &\to \frac{\Delta p_0}{p} + \varepsilon, & \frac{\Delta K}{K} &\to \frac{\Delta K}{K} + \varepsilon, \\ b_2 &\to b_2 - \frac{2b_3}{R_{\text{ref}}} \langle D_x \rangle \varepsilon, & b_3 &\to b_3. \end{aligned} \tag{8.26}$$

Here $\langle D_x \rangle$ denotes the average horizontal dispersion over all the bends with errors.

To investigate this behaviour, we will examine various fits where we force the model momentum-offset $\frac{\Delta p_0}{p}$ to distinct values. We assume that the momentum offset in the fit results in a systematic (average) shift of the trajectory in the simulation:

$$x^{\text{M}} = x^{\text{F}} - \Delta x^{\text{F}}, \tag{8.27}$$

where x^{F} denotes the (simulated) horizontal position during the fitting, x^{M} the (unknown) horizontal position in the real machine and Δx^{F} is the (artificially) introduced offset (In this case by changing $\frac{\Delta p_0}{p}$). For b_2 and b_3, Eqs. (8.16c) read in this case:

$$b_2{}^{\text{F}} = b_2{}^{\text{M}} - b_3{}^{\text{M}} \frac{2\Delta x}{R_{\text{ref}}} \tag{8.28a}$$

$$b_3{}^{\text{F}} = b_3{}^{\text{M}}. \tag{8.28b}$$

Here b_2^{F} and b_3^{F} denote the quadrupolar and sextupolar field errors, respectively, resulting from the fit and b_2^{M} and b_3^{M} denote the field errors in the machine. In the case of a forced $\frac{\Delta p_0}{p}$ we have

$$\Delta x^{\text{F}} = \langle D_x \rangle \frac{\Delta p_0}{p}, \tag{8.29}$$

where $\langle D_x \rangle$ denotes the average horizontal dispersion at the concerned MBIs. These average dispersions (calculated from the nominal models) are

$$\langle D_x \rangle = \begin{cases} -1.7\,\text{m} & \text{for TI8 and} \\ 1.5\,\text{m} & \text{for TI2.} \end{cases} \tag{8.30}$$

Therefore, assuming $b_3 = -4.7 \cdot 10^{-4}$ (from simulations, Eq. (8.25)), we expect from Eqs. (8.28b)

$$b_2 = b_2{}^0 - 639.2 \frac{\Delta p_0}{p} \tag{8.31}$$

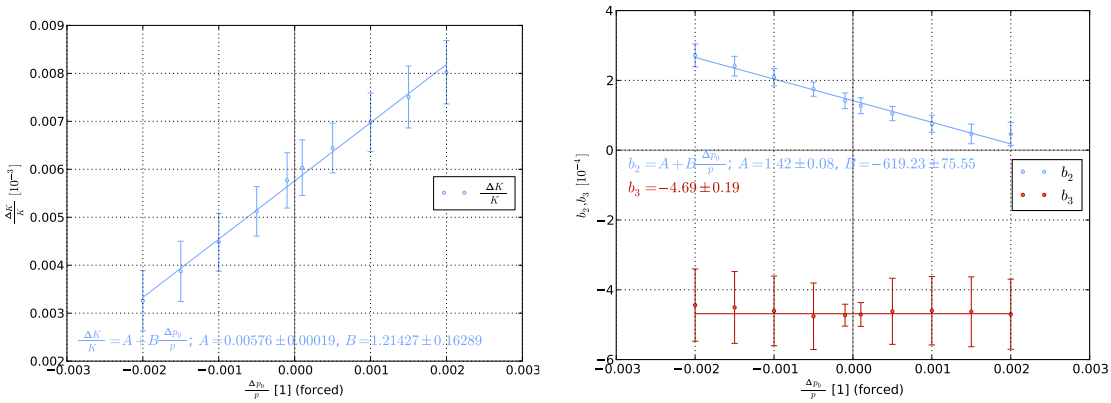
for TI8 and

$$b_2 = b_2{}^0 + 564.0 \frac{\Delta p_0}{p} \tag{8.32}$$

for TI2.

The results of such fits for TI8 are plotted in Figs. 8.38. The b_3 is almost constant

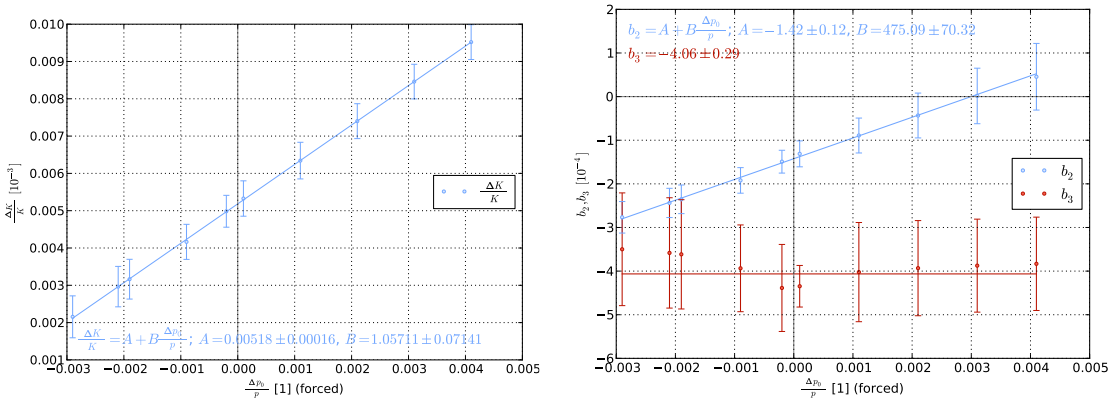
for all the fits, as expected. The average $(-4.69 \cdot 10^{-4})$ is in perfect agreement with the b_3 from the 2D magnet simulations $(-4.7 \cdot 10^{-4})$. Also the dependence of b_2 on $\frac{\Delta p_0}{p}$ meets very well the expectations (Eq. (8.31)).



(a) $\frac{\Delta K}{K}$ (b) b_2, b_3

Figure 8.38.: Dependence of fit parameters on forced values of $\frac{\Delta p_0}{p}$ for TI8.

For TI2 (Figs. 8.39) the slope of b_2 is slightly smaller than expected (475 cf. to 564 in Eq. (8.32)), but if the slope is calculated from the average b_3 resulting from the same fits $(4.06 \cdot 10^{-4})$, one obtains a slope of 487.2 which is in very good agreement with the slope in the plot.



(a) $\frac{\Delta K}{K}$ (b) b_2, b_3

Figure 8.39.: Dependence of fit parameters on forced values of $\frac{\Delta p_0}{p}$ for TI2.

For $\frac{\Delta K}{K}$ we expect

$$\frac{\Delta K}{K} = \left(\frac{\Delta K}{K}\right)_0 + \frac{\Delta p_0}{p}. \tag{8.33}$$

This expectation is in good agreement with the results for TI2 (Fig. 8.39(a)), but slightly worse for TI8 (Fig. 8.38(a)).

For further reference, we also note down the b_2 values at $\frac{\Delta p_0}{p} = 0$, which can be read off from the linear fits. They are (in units of 10^{-4}):

$$b_2 = \begin{cases} -1.42 \pm 0.12 & \text{for TI2,} \\ 1.42 \pm 0.08 & \text{for TI8.} \end{cases} \tag{8.34}$$

8.5.6. Quadrupolar Field Error

We have shown in the previous section that the sextupolar field error b_3 can be very nicely explained by the 2D simulations. This is not the case for the quadrupolar field error b_2, since the model (as the magnet) is perfectly symmetric. Instead, the systematic b_2 components are an artifact of the numerical model:

MadX, on which all the models are based, treats rectangular bending magnets (as in the transfer lines) as sector bending magnets with tilted ends. This means that the simulation code treats the beam within the magnet always centered. Therefore no feeddown effects from the sagitta of the beam are taken into account. One can compare this with the real situation as follows:

From the length of the bending magnet $L = 6.33\,\text{m}$ and the bending angle of the magnet $\alpha = 0.007611\,\text{rad}$, the local bending radius ρ is given by

$$\rho = \frac{L}{2\sin(\frac{\alpha}{2})} = 831.693\,\text{m}. \tag{8.35}$$

From these values we get the mean horizontal distance of the beam $\bar{x}$ from the center of the magnet as

$$\bar{x} = \frac{\rho^2}{2L}\left(\alpha - \sin\alpha\right). \tag{8.36}$$

This value is positive for TI2 and negative for TI8:

$$\bar{x} = \begin{cases} 4.015\,\text{mm} & \text{for TI2,} \\ -4.015\,\text{mm} & \text{for TI8.} \end{cases} \tag{8.37}$$

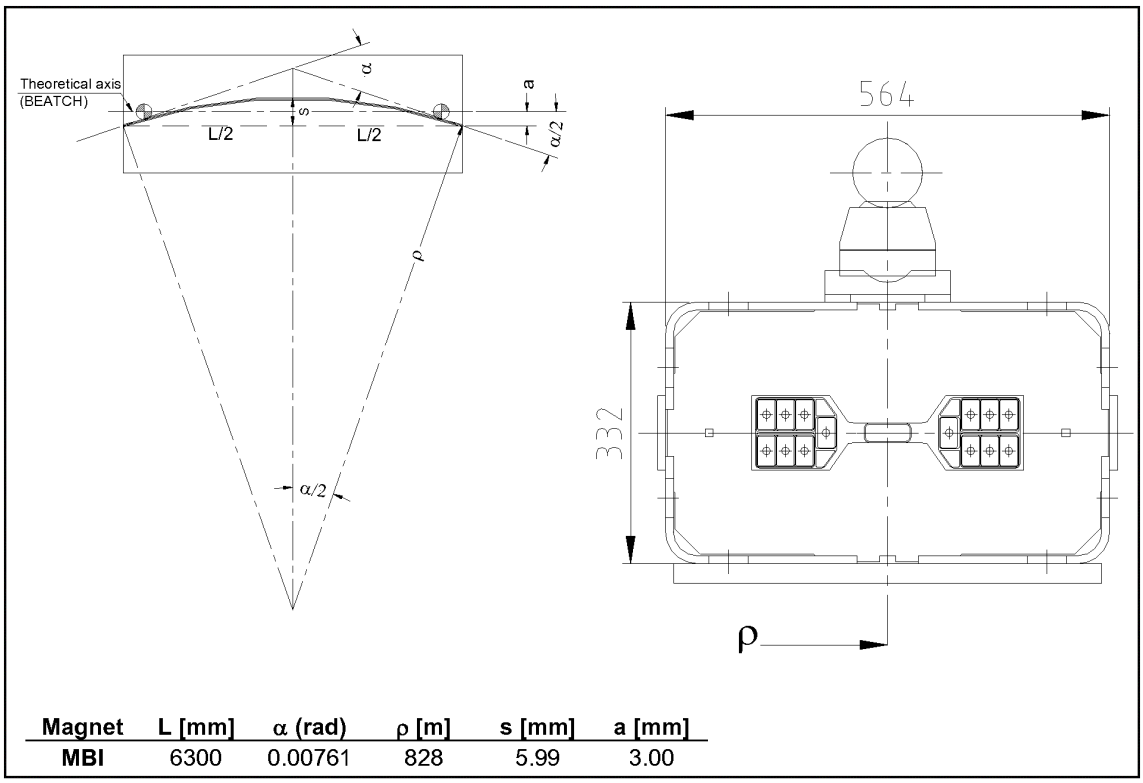
From this we can calculate the expected feeddown from Eq. (8.16c), using $b_3 = -4.7\cdot$

10^{-4} as calculated from the magnet simulations (Sec. 8.5.4). Finally we get

$$\tilde{b}_2 = \begin{cases} -1.5 \cdot 10^{-4} & \text{for TI2,} \\ 1.5 \cdot 10^{-4} & \text{for TI8.} \end{cases} \tag{8.38}$$

These numbers have the correct signs and are perfectly within the error limits of the numbers calculated from the fits in Sec. 8.5.5, Eq. (8.34).

One issue remains unresolved within this context: To maximise the aperture in the magnets, they were shifted outwards by 3 mm during installation (the value a in Fig. 8.40) [WWDB03] [Wet04]. Therefore, the resulting mean horizontal distance from the center would only be about 1 mm and from this one would only expect an resulting $|b_2|$ of about $0.38 \cdot 10^{-4}$. One possible explanation for the higher measured b_2 could be that the energy of the line was not perfectly matched. Nevertheless, to investigate this in more detail, further dedicated measurements would be necessary.



Magnet	L [mm]	α (rad)	ρ [m]	s [mm]	a [mm]
MBI	6300	0.00761	828	5.99	3.00

Figure 8.40.: Sketch of beam sagitta within an MBI magnet. (Taken from [WWDB03]).

8.6. Correction Measures

8.6.1. Model Corrections

As described in the previous sections, the investigations clearly pointed in the direction of field errors in the MBIs and a detuning of the quadrupole chains. To improve the situation we took the following measures:

- The values for higher order components were set in the models to

$$b_3 = -4.7 \quad \text{for both transfer lines,} \tag{8.39a}$$

$$b_2 = \begin{cases} 1.35 & \text{for TI8,} \\ -1.35 & \text{for TI2.} \end{cases} \tag{8.39b}$$

  These were the best estimates for these values at that time. At the time of writing, we would consider a value in the range of $1.4 \leq |b_2| \leq 1.5$ as more consistent (see Eq. (8.34), Eq. (8.38)).

- To take into account the found value for $\frac{\Delta K}{K}$, the calibration curve of the main quadrupole chains of the lines was decreased by 0.6% in the machine [MFF+10].

- A new version of the BPM firmware was deployed, which then contained the correct calibration curves for the BPMs. This was done in July 2009 for the transfer line BPMs and in October 2009 for the LHC ring BPMs [Jen10].

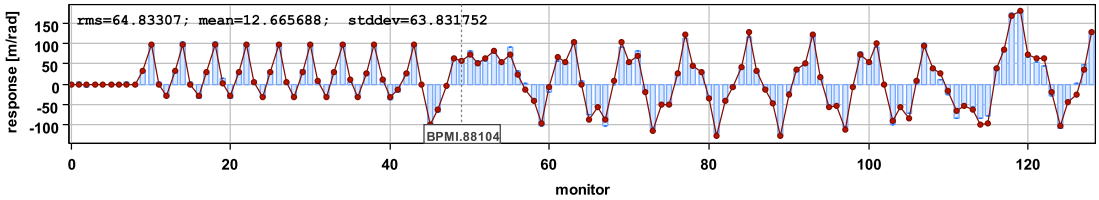
The transfer line optics were finally rematched to the optics of the LHC, using the models with the magnetic errors. After this rematching, once again we took kick-response data by changing corrector strengths in the transfer line and measuring the orbit response in the line and the adjacent sector of the LHC. This was done for TI8 during the injection test in November 2009[12].

Figures 8.41 demonstrate the very good model-measurement agreement after these improvements by showing one example response for each plane. Also the coupling is in very good agreement with the model, because of the correctly implemented calibration correction for the BPMs. This is visualized in Figs. 8.42 which show the out-of-plane responses for the same correctors (cf. Fig. 5.11(b) for the initial situation).

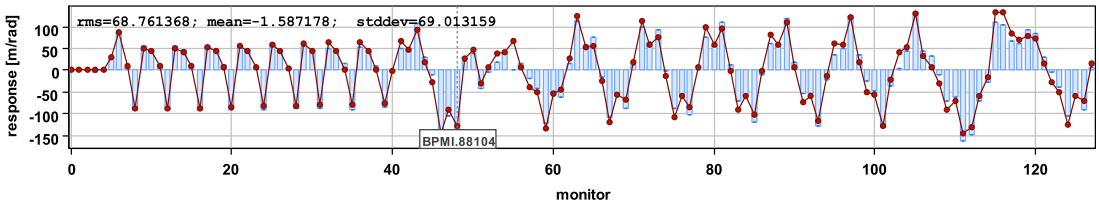
8.6.2. Dispersion Matching

The measures described in the last section still do not improve the dispersion mismatch at the injection point between the transfer lines and the LHC. We already

[12]Data taken: 2009-11-07, 22:38 to 2248. Accidentally taken at $\frac{\Delta p_0}{p} = -0.5$ permille.

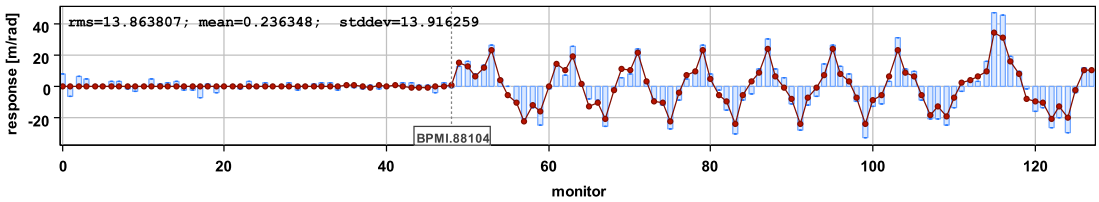


(a) Horizontal response for horizontal corrector MCIAH.80804.

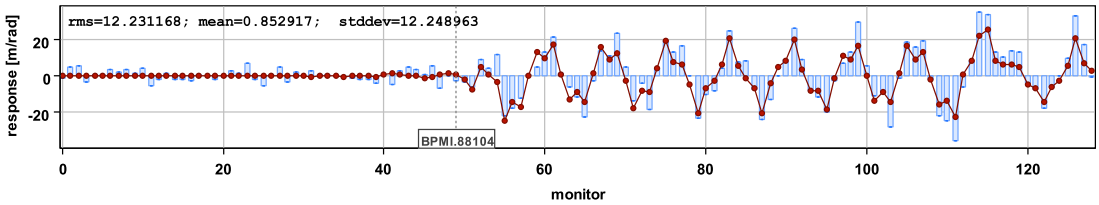


(b) Vertical response for vertical corrector MCIAV.80104.

Figure 8.41.: Example in-plane responses of corrector kicks in TI8 after model improvements and rematching. The marker 'BPMI.88104' indicates the last BPM in the transfer line.



(a) Vertical response for horizontal corrector MCIAH.80804.



(b) Horizontal response for vertical corrector MCIAV.80104.

Figure 8.42.: Example out-of-plane responses of corrector kicks in TI8 after model improvements and rematching. The marker 'BPMI.88104' indicates the last BPM in the transfer line. Note the different scale of the vertical axes w.r.t. Figs. 8.41.

demonstrated in Sec. 8.3.2 that model settings can be found, which are in good agreement with both, kick response and dispersion measurements and could be used to correct the mismatch. To validate the effectiveness of such a correction, we performed dedicated measurements in the context of LHC recommissioning in 2011: We first measured kick response and dispersion of the TI8 transfer line and the adjacent LHC sector 78 with the original settings. The result of this measurement[13] for the horizontal dispersion is shown in Fig. 8.43.

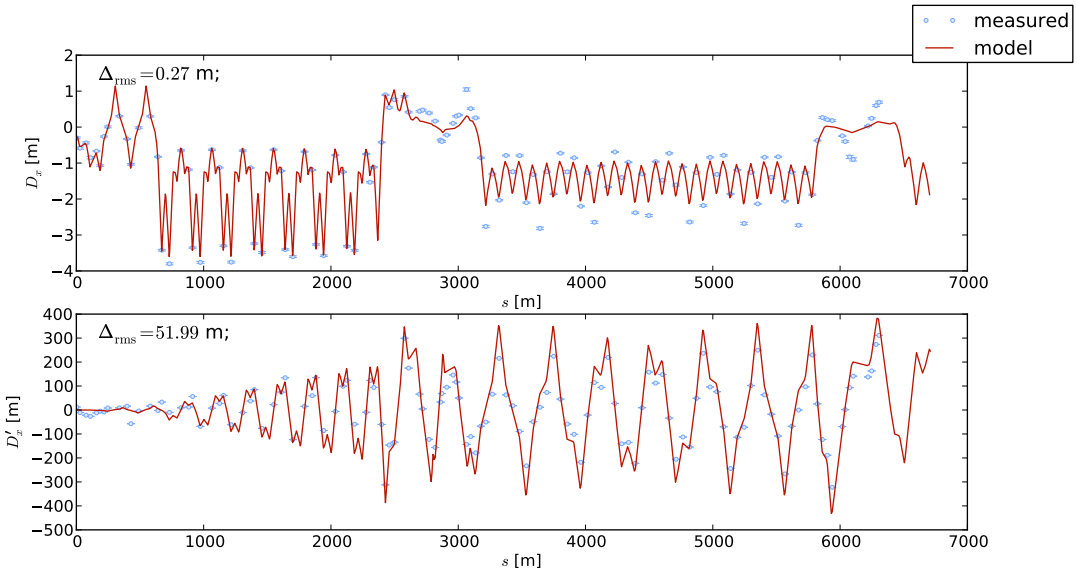


Figure 8.43.: Original horizontal dispersion in TI8 and LHC sector 78.

From these measurements we calculated a correction with the help of Aloha using two individually powered quadrupoles towards the end of the transfer line. The values which were trimmed into the machine are listed in Table 8.8.

Quadrupole name	initial strength [m^{-2}]	trim (delta) [m^{-2}]	relative trim
MQIF.87600	0.033581	-0.001038	3%
MQID.88000	0.024875	0.001998	8%

Table 8.8.: Trims to correct dispersion mismatch with two quadrupoles at the end of TI8.

After this correction we remeasured both kick response and dispersion to observe

[13]Data taken: 2011-03-01 23:30 to 2011-03-02 01:45

the effect of the trim[14]. The resulting horizontal dispersion is shown in Fig. 8.44. The residual for the dispersion is reduced from $\Delta^D_{\mathrm{rms}} = 0.27\,\mathrm{m}$ to $\Delta^D_{\mathrm{rms}} = 0.07\,\mathrm{m}$, without effecting the quality of kick response and second order dispersion.

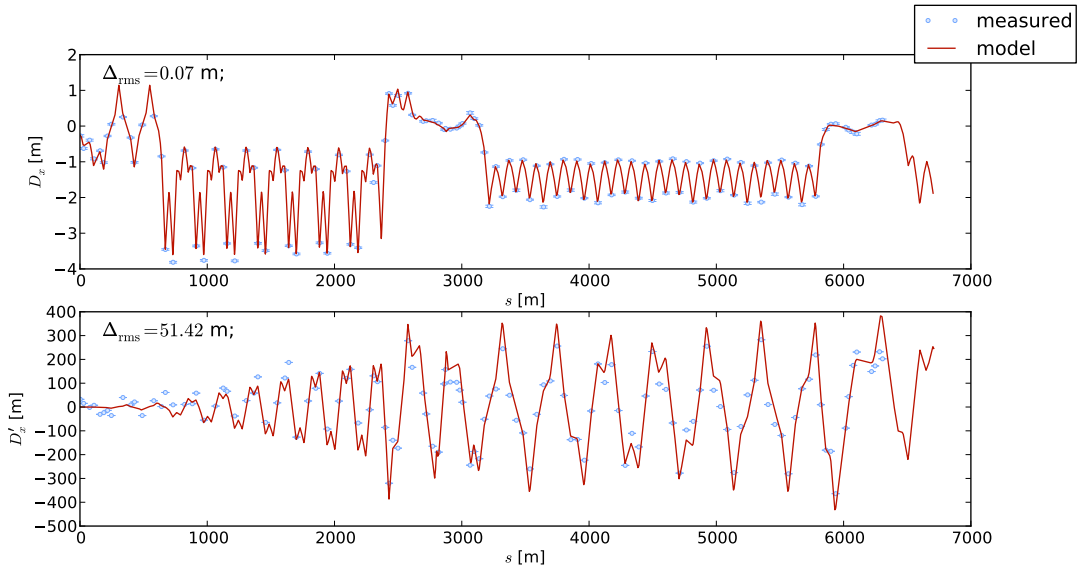


Figure 8.44.: Horizontal dispersion in TI8 and LHC sector 78 after applied correction as given in Table 8.8.

Although this shows that such corrections work in principle, one has to be a little bit more careful when applying them to the operational machine: This is, because the transfer line collimation systems add additional constraints to the transfer line optics. For example in TI8 there are three collimators per plane at the end of the line, which are located between the cells 874 an 881, just in the region, where also the matching quadrupoles are placed. To ensure the phase space coverage of the collimators, the betatron phase difference from one collimator to the next must be 60° within a tolerance of ±5°. Neither the correction calculated in Sec. 8.3.2, nor the correction calculated in this section respected these constraints.

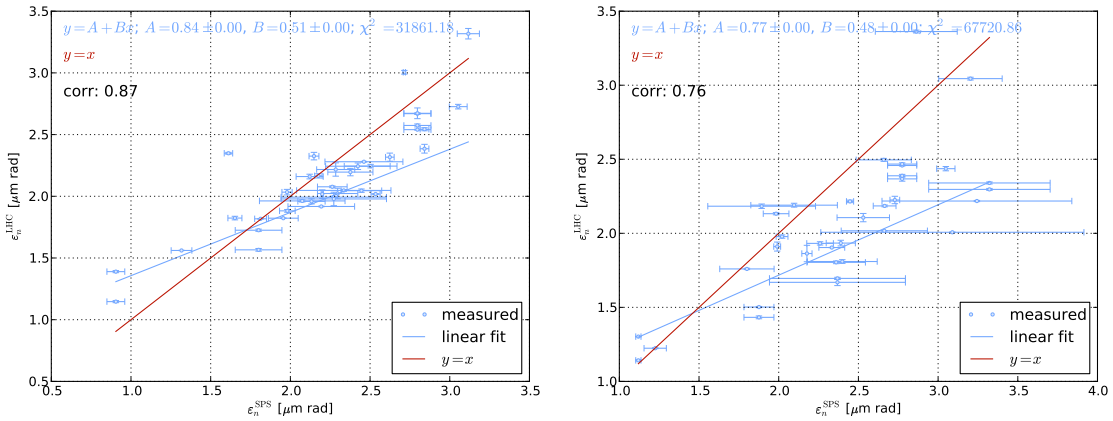
An additional `SensitivityMatrixContributor` (see Sec. 7.2) was added to Aloha, to cope with these additional constraints. It contributes to the sensitivity matrix in such a way that the errors on the constraints are minimized by the fits (in addition to minimizing dispersion and kick response errors). First tests of this method showed that it is possible to find solutions, which respect all the constraints, especially by fitting with SVD and using all the quadrupoles in the matching section at the end

[14]Data taken: 2011-03-02 02:15 to 03:00

of the line. The concrete solution, which could be put into the machine is still under discussion.

8.7. Emittance Preservation

We used wirescan measurements[15] from the logbook to get an impression of the ratio between the emittance in LHC and the one in SPS. Figures 8.45 show the results of this data collection for beam 1 and Figs. 8.46 for beam 2. The data points are the average of the in- and out- values from the wire scanners. Although the plots are well correlated (about 80%), the linear fits are not sginificant because of the large errorbars and the resulting high χ^2. To visually judge the trend, an additional line (red) is shown in the plots, which indicates $\varepsilon_n^{LHC} = \varepsilon_n^{SPS}$.



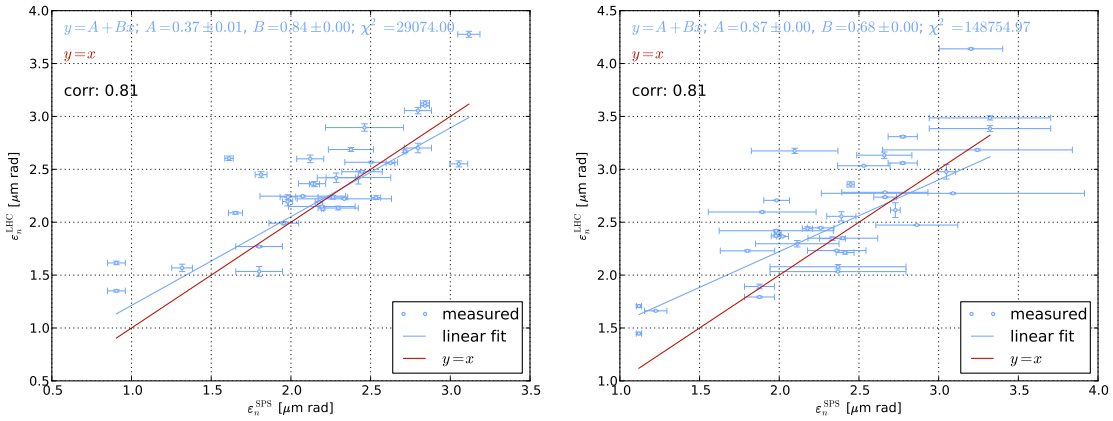
(a) Horizontal (b) Vertical

Figure 8.45.: Emittance transfer from SPS to LHC beam 1.

These plots show a clear difference between beam 1 and beam 2: While for beam 1 the measured emittance in the LHC tends to be smaller than in the SPS, for beam 2 it follows more the expected behavior i.e., that the emittance in LHC should be the same or even slightly larger than in SPS.

Later we performed dedicated measurements[16] to check the emittance preservation between SPS and LHC: We measured the emittance with wire scanners in the SPS at the flat top of the SPS cycle and in the LHC for the same shot. The wire scanners in the LHC were directly triggered after the injection event. The results of these measurements are plotted in Figs. 8.47 and Figs. 8.48. All these injections were done

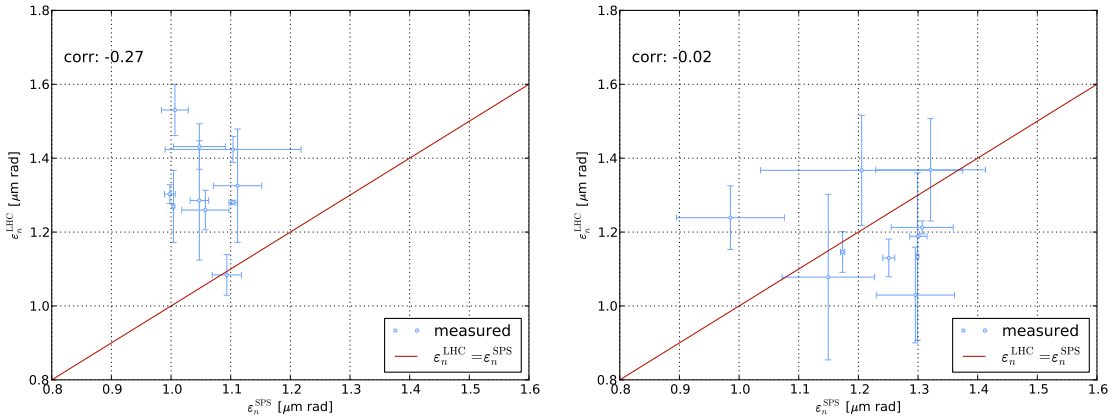
[15]Data taken: 2009-09-28 to 2009-11-04
[16]Data taken: 2011-02-27 10:00 to 12:00



(a) Horizontal (b) Vertical

Figure 8.46.: Emittance transfer from SPS to LHC beam 2.

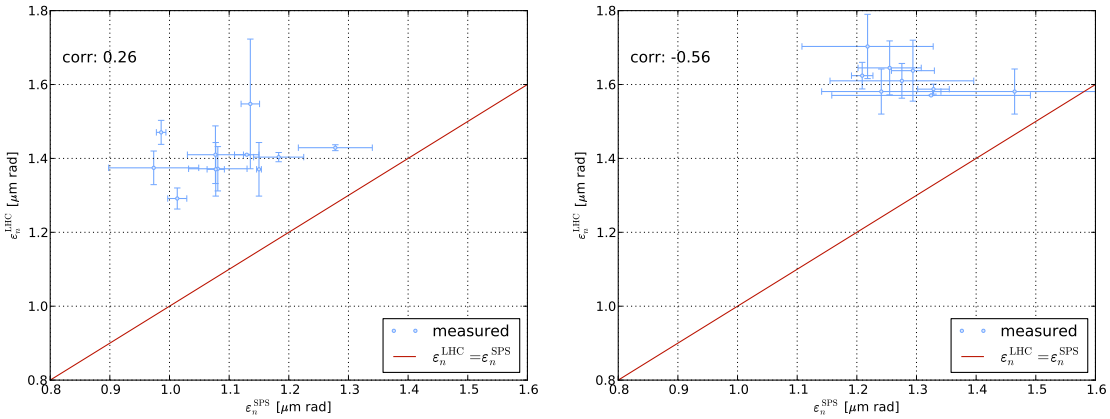
with the same setup in the preinjectors and the measured injected emittances were between 0.9 μm rad and 1.6 μm rad. It has to be noted, that the emittances in the SPS were larger in the vertical plane (1.26±0.02 μm rad) than in the horizontal plane (1.08±0.01 μm rad). For each beam and plane we got between 9 and 11 measurements. The averaged results are summarized in Table 8.9.



(a) Horizontal (b) Vertical

Figure 8.47.: Emittance transfer from SPS to LHC beam 1.

We consider the very small blowup for beam 1, vertical (0.97±0.04, which actually would mean beam 'shrinking') as systematic measurement error, which for the moment remains unexplained (Already the large errorbars in Fig. 8.47(b) indicate that



(a) Horizontal (b) Vertical

Figure 8.48.: Emittance transfer from SPS to LHC beam 2.

	ε_n^{SPS} [μm rad]	ε_n^{LHC} [μm rad]	$\frac{\varepsilon_n^{LHC}}{\varepsilon_n^{SPS}}$ [1]
beam 1, H	1.06 ± 0.01	1.32 ± 0.03	1.25 ± 0.04
beam 1, V	1.23 ± 0.03	1.19 ± 0.04	0.97 ± 0.04
beam 2, H	1.10 ± 0.02	1.40 ± 0.02	1.28 ± 0.03
beam 2, V	1.29 ± 0.03	1.62 ± 0.02	1.25 ± 0.03

Table 8.9.: Average measured emittances for SPS and LHC (data taken: 2011-02-27 10:00 to 12:00).

this dataset is not very useful in this context.). The rest of these dedicated measurements indicate an emittance growth between SPS and LHC of about 1.25. Although this is high compared to the maximum emittance growth budget as originally given in the LHC design report [BCM+04, p. 197], it causes no serious problems in the current situation, where the injected emittances are very small (in the range between 1 an 2 μm rad) compared to the design value of the emittance at injection into the LHC (3.5 μm rad).

Sources for this unexpected emittance growth can be for example the previously underestimated transverse coupling [KFG+09] and the dispersion mismatch at the injection point. Up to the time of writing, no dedicated time was available to repeat these emittance measurements with a corrected dispersion matching as described in the previous section. This would help to disentangle the various contributions to the emittance growth. The measurements should also be repeated with emittances closer to the nominal LHC emittance, especially, because the analysis of the data

from October 2010 points towards lower emittance growth for larger beams (at least for beam 1, Figs. 8.47).

9. Summary and Outlook

9.1. Summary

The initial situation and the motivation for this work was the observation of inconsistencies in the optics of the LHC injection transfer lines during transfer line tests in the years 2004 to 2008: The asymmetric phase advance error in the two tranverse planes, the dispersion mismatch at the junction between the transfer lines and the LHC and an unexpectedly high coupling at the same location. We started this document by outlining these issues in Chap. 5.

Since techniques which are commonly used to measure the optics in an accelerator ring (multiturn measurements) are not applicable to lines where the beam only passes once, we based our analysis on kick-response measurements. Optics parameters cannot be derived directly from these measurements, therefore the analysis had to be based on model fits to measurment data. Although this principle is well known, one of the key challanges is the vast amount of available parameters for these fits. To provide a framework which made it possible to explore this parameter space in an inuitive way, a new software had to be developed. Another key idea was to combine the kick-response analysis with dispersion measurments.

All this lead to the development of the Java software *Aloha*, which provides an user-friendly GUI to compare measurement- and model-data in many different ways. The goal was to develop an open analysis framework which can import various kind of measurement results and compare and fit them to the numerical models. This goal could be accomplished by a generalization of the fitting routines and the use of a simple plug-in system. The theoretical background of these concepts was outlined in Chap. 6 of this thesis and the design and implementation of the software tools were described in Chap. 7.

For the binding to the numerical model, we created a Java API for MadX (*JMad*), which is described in the same chapter. In its current version JMad offers the key features for using existing MadX models from JAVA, like changing model parameters and calculate optics values. Next to Aloha, JMad is currently also used by the LHC online model and various optics-analysis tools. It became the key component to link any JAVA application (like the LHC controls software) in a natural way to MadX accelerator models, which are available for almost all accelerators at CERN. Also a simple GUI is available which provides editing- and plotting capabilities and can

easily be integrated into other applications.

Being prepared in such a manner, we started studying the previously described issues systematically, after several empirical optics corrections in 2008 had not lead to satisfying results. We showed in Sec. 8.3 that the optics mismatch at the junction of the LHC was correctable with few quadrupoles at the end of the line.

Our studies pushed BPM accuracy and measurement methods to the limits. This revealed inconsistencies in the calibration of BPM data, which could afterwards be corrected. Furthermore, we improvemed the standard dispersion-measurement technique at CERN. This made it possible to investigate dispersion up to second order, which finally proved to be consistent with our new models (see Sec. 8.4).

Sec. 8.5 is dedicated to the part into which the most effort was put: To understand the physical reasons for the issues in order to attack the problems in an appropriate way. For that purpose we developed a dedicated method to determine the chromaticity of the transfer lines from off-momentum kick response measurements. These measurements demonstrated, that the observed effects could be explained by higher order field components in the injection main bends. The observed value for the sextupolar component was confirmed later by numerical magnet simulations and a quadrupolar component could be explained by the feed-down resulting from the sagitta within the magnets.

Finally, we summarized the measures taken to improve the models in Sec. 8.6 and demonstrated the very good measurement-model agreement by actual measurements, which guarantees an excellent beam quality in the LHC.

9.2. Outlook

Although we consider the original issues as well understood, there remain some open questions. The most puzzling one is the observed relatively high b_2, which is in contradiction with the expectation, because the magnets are shifted outwards by 3 mm, as sketched in Sec. 8.5.6. Magnet measurements are planned to verify the field distribution within the magnets. One could also imagine dedicated beam based measurements to check the centering of the beam within the lines. Another issue, where detailed analysis is still ongoing is the coupling at the injection points. Ideas to explain the observed coupling, which is still slightly higher than expected from the model, are e.g. wrong BPM calibrations at the time of measurement or even statistical errors [KFG+09].

The most exciting ideas for further improvements are related to the spin-offs of this thesis, the computational tools, Aloha and JMad. Since a lot of attention was given to a sound design of these tools, they have the potential to evolve to powerful standard tools:

The most obvious improvement for Aloha would be an optional online integration,

which would make it possible to calculate optics corrections online and directly send the calculated trims to the machine. This, especially in combination with the variety of possible measurement inputs (kick response, dispersion, beta beat), would result in a very powerful one-click solution for optics corrections, similar to the functionality that YASP provides for orbit corrections. For the moment, work is ongoing to improve the internal structure of Aloha and clean up the code.

The second software-spinoff, JMad, already made its way as a standerd library for Java programs to communicate with MadX. For that reason, the most urgent goals for JMad development are stabilizing the API and the model-definition format. To accomplish these goals, a code-reviewing process has already started. Discussion is also ongoing to realease JMad as open source and thus make it available for users outside of CERN.

Also for JMad there exist many ideas for new features: For example, the combination of different models (TI8+LHC in our case) proved to be a very powerful technique throughout this thesis. Pushing this idea further, one could e.g. imagine a unified model treatment, which would allow simulating arbitrary parts of CERNs accelerator complex. A promising feature would also be e.g. extending JMads functionality to tracking, which would allow simple ad-hoc analysis of accelerator parts.

Appendix

A. Schematics

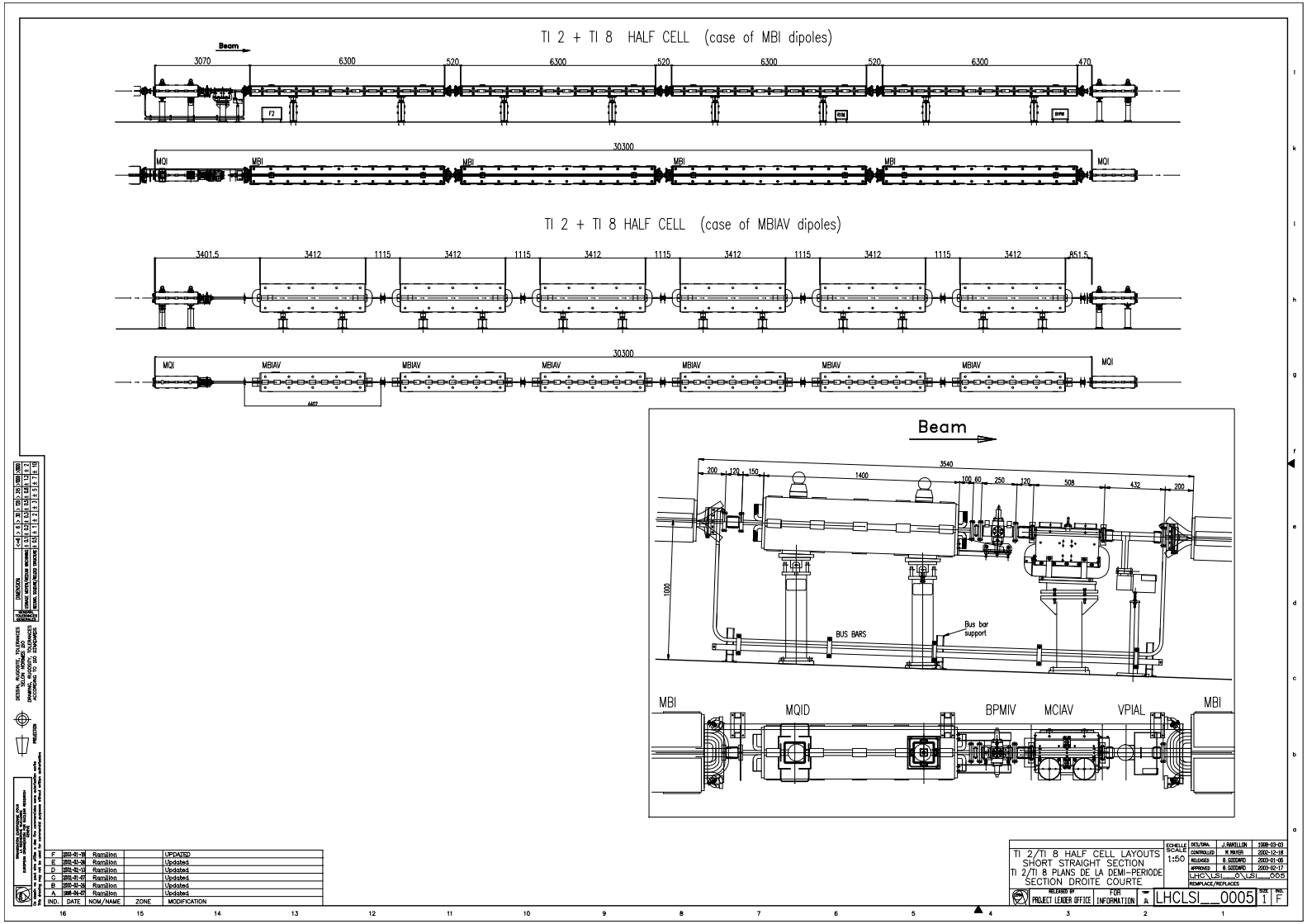


Figure A.1.: Half-cell layout of TI8 and TI2.

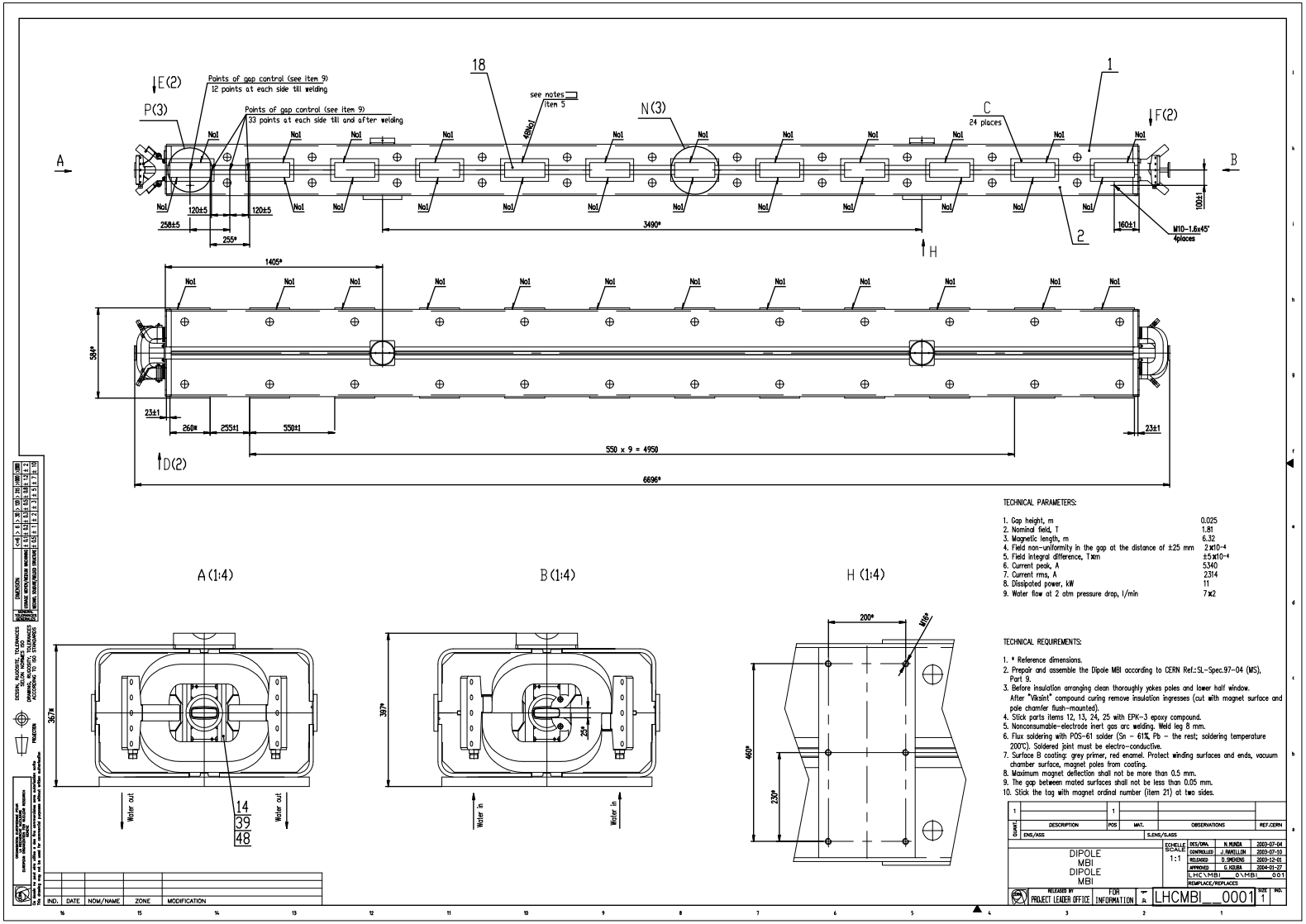


Figure A.2.: Schematics of the main bends in the LHC injection lines (MBI).

Bibliography

[AAA+08] O. Aberle, I. Agapov, M. Aiba, M. Albert, R. Alemany-Fernandez, Gianluigi Arduini, Ralph Wolfgang Assmann, R. Bailey, R. Billen, L. Bottura, Oliver Sim Brüning, A. Butterworth, R. Calaga, E. Carlier, P. Collier, B. Dehning, L. Deniau, S. Fartoukh, F. Follin, D. Forkel-Wirth, K. Fuchsberger, R. Giachino, M. Giovannozzi, B. Goddard, J J Gras, E. Hatziangeli, P. Hagen, D. Jacquet, L. Jensen, R. Jones, V. Kain, I. Kozsar, T. Kramer, G. Kruk, M. Lamont, J. Lewis, R. Losito, A. MacPherson, A. Masi, M. Meddahi, V. Mertens, M. Misiowiec, S. Page, L. Ponce, B. Puccio, S. Redaelli, C. Roderick, S. Roesler, F. Roncarolo, M. Sapinski, F. Schmidt, R. Schmidt, W. Sliwinski, R. Steinhagen, M. Strzelczyk, Y. Sun, B. Todd, E. Todesco, R. Tomas Garcia, J. Uythoven, W. Venturini-Delsolaro, H. Vincke, H. Vincke, E. Veyrunes, J. Wenninger, R. Wolf, C. Zamantzas, and F. Zimmermann, *The LHC injection tests*, Tech. Report LHC-Performance-Note-001, CERN, Geneva, 2008. 31

[ACC+05] M. Albert, F. Chevrier, G. Crockford, D. Jacquet, G. Kruk, M. Lamont, L. Mestre, and L. Normann, *LHC software architecture: TI8 commissioning*, 2005. 70

[AM73] Bruno Autin and Y. Marti, *Closed orbit correction of A.G.machines using a small number of magnets*, Tech. Report CERN-ISR-MA-73-17, CERN, Geneva, March 1973. 37

[Bau09] Jeremie Bauche, *Magnetic simulations of the MBI bendig magnets.*, July 2009, personal communication. 103

[BCL+04] Oliver Sim Brüning, Paul Collier, P. Lebrun, Stephen Myers, Ranko Ostojic, John Poole, and Paul Proudlock, *LHC design report volume 1, the LHC main ring*, CERN, Geneva, 2004. 7

[BCM+04] Michael Benedikt, Paul Collier, V. Mertens, John Poole, and Karlheinz Schindl, *LHC design report volume 3, the LHC injector chain*, CERN, Geneva, 2004. 25, 33, 115

[Bra09] Daniel Brandt, *CAS - CERN accelerator school: Course on beam diagnostics*, CERN yellow reports, no. CERN-2009-005, Geneva, 2009. 17

[CC96] P. Castro and P. Castro, *Luminosity and beta function measurement at the electron-positron collider ring LEP*, Ph.D. thesis, CERN, Geneva, 1996. 2, 17

[CER53] CERN, *Convention for the establishment of a european organization for nuclear research*, July 1953. 5

[CER08] ———, *CERN public website*, http://www.cern.ch, 2008. 5

[Cha99] Alex Chao, *Handbook of accelerator physics and engineering*, World Scientific Publishing Co Pte Ltd, March 1999. 9, 15

[Far08] S. Fartoukh, *Proposal for measuring chromaticity of the TI8 transfer line*, 2008, personal communication. 95

[FBG+10] K. Fuchsberger, V. Baggiolini, R. Gorbonosov, W. Herr, V. Kain, G. Müller, S. Redaelli, F. Schmidt, and J. Wenninger, *JMad-Integration of madx into the java world*, 2010. 61

[FFG+09] K. Fuchsberger, S. Fartoukh, B. Goddard, R. Jones, V. Kain, M. Meddahi, V. Mertens, and J. Wenninger, *Determination of the chromaticity of the TI 8 transfer line based on kick response measurements*, 2009. 99

[Fuc09] K. Fuchsberger, *Aloha - optics studies by combined kick-response and dispersion fits*, Tech. Report BE-Note-2009-020 OP, CERN, Geneva, May 2009. 48

[GFK+09] B. Goddard, K. Fuchsberger, V. Kain, M. Meddahi, V. Mertens, T. Risselada, and J. Wenninger, *Coupling at injection from tilt mismatch between the LHC and its transfer lines*, Tech. Report LHC-Performance-Note-003, CERN, Geneva, 2009. 33

[GGKR04] B. Goddard, M. Gyr, V. Kain, and Thys Risselada, *Geometrical alignment and associated beam optics issues of transfer lines with horizontal and vertical deflection*, 2004. 1

[HGK+04] P. Hagen, M. Giovannozzi, J. P. Koutchouk, T. Risselada, S. Sanfilippo, E. Todesco, and E. Wildner, *WISE: an adaptive simulation of the LHC optics*, EPAC06 proceedings, 2004, p. 2248. 102

[Hin08] Frank Hinterberger, *Physik der teilchenbeschleuniger und ionenoptik*, 2. aufl. ed., Springer, Berlin, April 2008. 9, 11, 15, 27, 95

[HS04] Werner Herr and Frank Schmidt, *A MAD-X primer*, Tech. Report CERN-AB-2004-027-ABP, CERN, 2004. 20, 47, 61

[Jen10] Lars Jensen, *Implementation of nonlinear correction coefficients for the LHC BPMs.*, November 2010, personal communication. 109

[Jon09] Rhodri Jones, *BPM scaling errors for different correction methods.*, July 2009, personal communication. 92

[Ker41] D. W. Kerst, *The acceleration of electrons by magnetic induction*, Physical Review **60** (1941), no. 1, 47. 12

[KFG+09] V. Kain, K. Fuchsberger, B. Goddard, D. Karadeniz, M. Meddahi, and J. Wenninger, *Phase-dependant coupling at injection from tilt mismatch between the LHC and its transfer lines*, 2009. 33, 115, 118

[KS05] Verena Kain and Rudiger Schmidt, *Machine protection and beam quality during the LHC injection process*, Ph.D. thesis, Wien TU, Wien, 2005. 2

[Lee04] S. Y. Lee, *Accelerator physics*, 0002 ed., World Scientific Pub Co, December 2004. 9, 13

[MAF+08] M. Meddahi, I. Agapov, K. Fuchsberger, Werner Herr, V. Kain, B. Goddard, E. Gianfelice, D. Missiaen, V. Mertens, Thys Risselada, J. Uythoven, and J. Wenninger, *TI 8 transfer line optics studies*, Tech. Report LHC-Performance-Note-004, CERN, Geneva, 2008. 1, 77, 79

[MAG+09] V. Mertens, I. Agapov, B. Goddard, M. Gyr, V. Kain, T. Kramer, M. Lamont, M. Meddahi, J. Uythoven, and J. Wenninger, *Beam commissioning of injection into the LHC*, 2009. 1

[MFF+10] M. Meddahi, S. Fartoukh, K. Fuchsberger, B. Goddard, W. Herr, V. Kain, V. Mertens, and J. Wenninger, *Results of 2009 optics studies of the SPS to LHC transfer lines*, 2010. 109

[MPB+10] G. Müller, C. A Pons, X. Buffat, M. Giovannozzi, S. Redaelli, K. Fuchsberger, M. Lamont, and F. Schmidt, *The online model for the large hadron collider*, 2010. 61, 70

[Mue] Gabriel Mueller, Ph.D. thesis, unpublished. 70

[PFTV07] William H Press, Brian P Flannery, Saul Arno Teukolsky, and William T Vetterling, *Numerical recipes: The art of scientific computing; 3rd ed.*, Cambridge University Press, 2007. 37

[Saf97] J Safranek, *Experimental determination of storage ring optics using orbit response measurements*, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **388** (1997), no. 1-2, 27–36. 2, 35, 47

[San69] Matthew Sands, *Head-Tail effect: An instability in storage rings*, Tech. Report SLAC-TN-69-8, March 1969. 13

[Sun10] Sun, *Java native access (JNA): pure java access to native libraries*, https://jna.dev.java.net/, 2010. 64

[Tur96] S. Turner, *CAS - CERN accelerator school : 5th advanced accelerator physics course*, CERN yellow reports, no. CERN 95-06, 1996. 13

[UAA+08] J. Uythoven, Gianluigi Arduini, Ralph Wolfgang Assmann, N. Gilbert, B. Goddard, V. Kain, A. Koschik, T. Kramer, M. Lamont, V. Mertens, R. Redaelli, and J. Wenninger, *Beam commissioning of the SPS-to-LHC transfer line TI 2*, 2008. 27, 29, 135

[UAC+05] Jan Uythoven, Gianluigi Arduini, Chao Yu Chiu, Brennan Goddard, Delphine Jacquet, Verena Kain, Mike Lamont, Volker Mertens, Alan Spinks, and Jörg Wenninger, *Commissioning of the LHC beam transfer line TI 8*, 2005. 26, 29, 135

[Wen05] J. Wenninger, *YASP steering program user guide*, Tech. report, CERN, 2005. 18

[Wen06] ———, *Study of the TI 8 optics and beam stability based on beam trajectories*, Tech. report, CERN, Geneva, 2006. 1, 30, 72, 85

[Wen08] ———, *Response matrix and dispersion study of the TI2 transfer line*, Tech. Report CERN-AB-Note-2008-008, CERN, Geneva, 2008. 1, 30

[Wet04] W Weterings, *Alignment compensation for bending radius in TI 2 transfer line magnets*, 2004. 108

[Wie07] Helmut Wiedemann, *Particle accelerator physics*, 3rd ed., Springer, June 2007. 10

[Wil00] Klaus Wille, *The physics of particle accelerators: An introduction*, Oxford University Press, March 2000. 9

[WWDB03] W. Weterings, W. Weterings, CERN. Geneva. AB Department, and BT, *Alignment compensation for bending radius in TT40 and TI 8 transfer line magnets*, 2003. 108, 140

[ZM03] Frank Zimmermann and Michiko G. Minty, *Measurement and control of charged particle beams*, 1 ed., Springer, Berlin, May 2003. 17

List of Figures

List of Tables