Chapter 8

Reconstruction

The first two sections of this chapter are devoted to the description of the pattern recognition and vertex finding algorithms developed for the CMS Tracker. The Tracker alignment strategy problem is discussed in Section 8.3.

8.1 Pattern recognition and track reconstruction

The pattern recognition algorithms adopted for the CMS Tracker have to process a large number of hits per event: typically 5×10^3 hits at low luminosity and ten times more at high luminosity. In order to overcome the severe combinatorial problems the concept of *road* preselection is used in the first stage of the algorithms. In the second stage, the Kalman filter [8-1] is used to carry out final hit selection and track fitting. Three programs have been developed: the Global Track Finder (GTF), the Connection Machine (CM) and the Forward Kalman Filter (FKF). They all require a *learning* phase where the information of the detector geometry is processed once in order to create a database used in the pattern recognition phase. The algorithms work with a simplified geometry:

- detectors are 2D planes organised in layers;
- an average material distribution is used;
- the radial component of the magnetic field is assumed to be zero.

Tracks with $p_T > 0.8$ GeV/c[1] and $|\eta| < 2.4$ originating from the interaction point have ideally between 8 and 15 hits. Typically half of them are 3D points and the rest are 2D points. The hits in the Tracker fall into three categories. Firstly, there are pixel hits which have high accuracy in both local coordinates and they give precise 3D points. Secondly, hits in the individual silicon and gas detectors provide precise position measurements in the direction normal to the strips. Finally, measurements in twin detectors can be combined together to provide 3D points. This procedure, called stereo matching, can cause ambiguities, if the track density is too high. Some measurements may not be matched but can still be used as hits of the second type (2D). Since the two measurement planes are displaced by the detector thickness, the exact stereo coordinates of the matched hits depend on the track direction. At a preliminary stage of the pattern recognition the 3D points in the back-to-back detectors are constructed assuming infinite momentum tracks originating from the nominal centre of the Tracker. During the Kalman filtering the predicted track direction is used.

Of the thirteen or more nominal hits per track available in the detector, some may be missing due to inefficiency or badly measured because of overlapping tracks. The algorithms have to exclude these *bad hits* and skip missing layers during the pattern recognition phase. In the past,

[1] Tracks with $p_T < 0.8$ GeV/c in the 4 T magnetic field do not escape the Tracker volume in the radial direction and spiral towards one of the endcaps.

studies have also been done with the Local Track Finder (LTF) [8-2], a program that creates the roads for the track finder using the trajectories of Monte Carlo generated tracks and uses the detailed material description to account for multiple scattering effects.

8.1.1 Global Track Finder

In this approach, two sets of layers are defined during the learning phase: the Outer Layers (OL) and the Middle Layers (ML). Typically, barrel outer layers are the outmost MSGC layers and middle layers are the silicon layers providing 3D points. All pairs of detectors are considered with one detector belonging to OL and the other to ML. The pair is stored in the List of Starting Detectors (LSD) if the two detectors can be crossed by *interesting* tracks. Typical parameters of the *interesting* tracks are $p_T > 0.9$ GeV/c, $|\eta| < 2.4$ $|z| < 3\sigma_z$, where σ_z is the width in z of the luminous region. LSD consists of several hundred thousand pairs of detectors, ranging from 3×10^5 to 4×10^5 respectively for the low and the high luminosity versions of the Tracker. In addition, for each layer, a list of *nearby* layers is created. A *nearby* layer is defined as the previous or the succeeding layer hit by an *interesting* track when it crosses the layer under consideration.

During the pattern recognition phase the algorithm loops over LSD and searches for pairs where both detectors have hits. For any pair of hits a helix is constructed through the hits and the beam axis. Starting from the hit at OL, the track is then propagated backwards to the next internal layer computing the point of intersection of the trajectory with a specific detector. Hits are searched for in the vicinity of the intersection point. The track is then propagated to the next layer. The propagation ends successfully, when a minimum number of hits is found. If there are too few extra layers to add to the track or if the number of subsequent layers without hits is too large, the track candidate is abandoned.

Track candidates are eventually reprocessed with the Kalman filter using the parameters of the helix computed in the previous step as starting parameters. The track parameters and their covariance matrix are updated when a new hit is assigned to the track. The procedure is iterated with a variable road size taking into account the error introduced by the track extrapolation. When there are several hit candidates in a given detector, they are stored in a stack and tried in turn. Hits are not accepted if the χ^2 increment exceeds a given value. Tracks are eventually accepted if they have a minimum number of hits and an acceptable χ^2 per number of degrees of freedom. A track is rejected if the number of consecutive missing layers is too large.

Accepted tracks are stored in a list. Since the algorithm starts with many detector combinations, it often finds the same track or its segments several times. Duplicates are removed from the list of accepted tracks at the end of the pattern recognition. The longest track is kept, if its χ^2 is smaller than a given value. Otherwise, the track with smallest χ^2 is kept.

In the final step the algorithm performs a forward propagation – from inside to outside the Tracker volume – fitting the track parameters at the last measured point.

Since the algorithm starts from points in OL and ML, it cannot find tracks which do not have hits in those layers[2]. In addition it cannot find very low p_T tracks that do not reach the OL. The efficiency to find tracks with very large impact parameters (as K0 or Λ decay products) is small because the initial track direction is calculated using the beam constraint. On the other hand the algorithm performs well for tracks not too far from the primary vertex, e.g. tracks generated by heavy flavour vertices.

8.1.2 Connection Machine

This algorithm makes use of the concept of *cells* characterising the granularity of the initial track search. A cell may coincide with a detector or a portion of it but may also include several

[2]It still may find interacting tracks provided the produced tracks are collinear with the original track.

detectors placed on the same layer. For each cell a list of backward links with other cells in internal layers is created by propagating a large number of tracks ($\approx 10^9$) through the detector. Any hit cell is linked with the previous hit cell and also, for the sake of redundancy, with the cell hit before that, thus skipping just one layer.

A list of Starting Cells (SC) is also defined typically consisting of all cells in the three outer barrel layers and a large part of the forward layers. Cells that do not have links, the innermost cells, are called Ending Cells (EC).

During the pattern recognition phase the algorithm loops over the SC list searching for non-empty cells. Hits in these cells are connected backward to hits in the linked cells and the procedure is iterated. When the track candidate has at least three hits, the algorithm uses a helix constraint before adding an extra hit. The track parameters computed using the last three hits are compared with those computed using the new hit and the last two hits. The new hit is accepted if the distance between the two helices in the parameter space is not too large.

Once the list of track candidates related to a given SC cell is completed, the candidates above a given track length are reprocessed with the Kalman filter. The three most precise hits in the track are used to compute the starting parameters of the track. The procedure described above is iterated starting again from the SC hits. It uses all the features of the Kalman filter including the so-called *smoothing* or *forward re-propagation* that provides a better description of the extrapolation error.

CM is less sensitive than GTF to kinks, interactions and decays in flight since the track model used to generate the links is very general. However, a large memory is needed to store the links (≈ 30 MB) and the learning step takes several hundred CPU hours for any new geometry configuration.

The present CM version reconstructs tracks with p_T above 0.9 GeV/c. The algorithm can in principle be used for lower p_T tracks but the rapid increase of the number of links with decreasing p_T poses practical problems. Choosing a cell size of $\pi/100$, the typical number of links stored to identify tracks of p_T larger than 0.9 GeV/c is about 10 million and it is halved by increasing the p_T cut to 2 GeV/c.

8.1.3 Forward Kalman Filter

This algorithm starts the reconstruction from the precision pixel layers applying the same strategy as the CM algorithm.

During the learning phase the lists of backward links are created connecting only the detectors of the external pixel layers to those of the internal pixel layers. The SC list contains all detectors in the external pixel layers.

During the pattern recognition phase, the algorithm loops over the SC list searching for non-empty cells. Hits in the external pixel layer are then combined with hits in the linked detectors of the internal pixel layers. For any combination of hits, the helix parameters and their errors are estimated using the constraint that the track comes from the beam axis. If the helix parameters are within the acceptance cuts, the algorithm propagates the track to the outer detectors using the Kalman filter. Tracks are accepted if a minimum number of hits is found. Accepted tracks are eventually reprocessed again with the Kalman filter without the beam constraint in a way similar to the CM algorithm.

The FKF algorithm is particularly suited for b-tagging studies where only well-reconstructed tracks with enough hits in the innermost layers are selected for further analysis.

This method has a much smaller memory requirement than the CM algorithm since only a few tens of thousand links are accumulated in the learning stage. The efficiency is high for tracks that have two pixel hits and the performance is quite comparable to those of the two algorithms described in the previous sections. Since the number of combinations is smaller, it

runs about ten times faster. It reconstructs also low p_T tracks (the present threshold is set at 0.5 GeV/c) and short tracks that have an interaction in the external silicon layers.

A comparison of the execution time for the three algorithms is shown in Table 8.1 that reports the time needed to reconstruct tracks of p_T larger than several thresholds in a high luminosity di-jet event.

Table 8.1: Event Reconstruction Time (ERT) for GTF, CM and KFK; a high p_T di-jet event at high luminosity was reconstructed using three p_T thresholds. The machine used was a Sun Ultra workstation, corresponding to about 45 CERN units

p_T^{cut} (GeV/c)	$\mathrm{ERT_{GTF}}$ (s)	$\mathrm{ERT_{CM}}$ (s)	$\mathrm{ERT_{FKF}}$ (s)
2.0	146	80	17
0.9	670	390	37
0.5			55

8.2 Vertex reconstruction

Primary vertex finding in colliding beam experiments is essentially a one-dimensional problem, since the transverse coordinates, x and y, are determined by the beam line with a high precision. Therefore fairly simple algorithms can do the pattern recognition, and the primary vertex finding efficiency, as well as the two-vertex resolution, depends predominantly on Tracker performance rather than on the algorithm.

The principal task of primary vertex finding is to identify particles produced by the pp collision that caused the trigger and to separate them from tracks produced in superimposed minimum-bias events. Two algorithms to identify vertices are described in the following. The first is dedicated to Primary Vertex search (PVF), whereas the second performs Global Vertex searches (GVF). At the price of a complex formulation, GVF achieves good performance in the reconstruction of secondary vertices.

8.2.1 PVF algorithm

An important feature of this algorithm is to order the reconstructed tracks according to their importance in vertex finding/fitting, i.e. according to the accuracy of their impact parameter determination. High p_T tracks are likely to come from the trigger event and to have precise impact parameter measurements. For these reasons they have a high probability to enter the algorithm first and to create a seed for the primary vertex of the trigger event. The basic steps in the algorithm are qualitatively as follows:

- Track qualities are checked: tracks with too few hits, bad χ^2, very low p_T, large xy-impact parameter or too far from $z = 0$ are rejected.
- The remaining tracks are sorted with increasing z impact parameter error.
- The z-impact point of the first track in the list is taken as a seed of z-coordinate for the first vertex candidate.
- Subsequent tracks in the list are examined in that order. Each track is either associated to the closest vertex candidate, according to tight tolerances, or forms a seed for a new vertex candidate, if it is significantly displaced from all vertex candidates created so far.
- If the track is associated to a vertex candidate, the vertex position is recomputed including the new track.

- After all tracks have been examined, a smoothing procedure is performed: each vertex candidate is refitted in three dimensions with the beam constraint [8-3]; poorly associated tracks are removed from the list for that vertex and the vertex position is recomputed.
- After smoothing, the track-vertex association procedure is repeated until a stable configuration is achieved.

The vertex recognition efficiency and two-vertex resolution are being studied with the algorithm.

8.2.2 GVF algorithm

The global 3D vertex finding algorithm has been developed and used for performance studies [8-4]. The algorithm reconstructs vertices starting from the reconstructed tracks and establishes a unique matching between tracks and vertices. Tracks are eventually refitted using the vertex information.

Three types of vertices are reconstructed in a given event:

- primary vertex, with an arbitrary number of tracks and transverse position constrained to the beam axis;
- secondary vertices, with an arbitrary number of tracks and arbitrary position in the transverse plane;
- V^0 vertices with two opposite-sign tracks and effective mass compatible with a known particle (e.g. K0, Λ).

After preselecting the tracks, the algorithm proceeds in two major steps: the initialisation (seeding) phase and the vertex finding phase.

8.2.2.1 Initialisation

The initialisation step provides a rough estimate of the vertex positions and – for each vertex – the list of assigned tracks and their parameters at the vertex. At this stage a single reconstructed track can be assigned to several vertex seeds.

For 'seeding' the primary vertices an algorithm quite similar to the one described in the previous section is used, exploiting the localisation of the interaction point in the transverse plane. The primary vertex seed is formed using only tracks with small transverse impact parameters and then all tracks whose impact point is not too far away are assigned to the seed.

The initialisation procedure for secondary vertices uses a three-step procedure. In the first step all possible two-track vertices are formed by track pairs with minimal distance in space. In the second step nearby vertices are joined. In the last step other tracks are attached to these identified vertex seeds.

8.2.2.2 Deformable templates algorithm

The optimal match between tracks and vertices is searched for using the deformable templates (elastic arms) approach (DTA) [8-5]. The N vertex seeds found in the initialisation step are used to start the DTA.

Each vertex seed $n \in N$ is characterised by the vertex position $\overrightarrow{x_n}$ and the set M_n of tracks assigned to it (vertex tracks). Each track m belonging to the set M_n is fitted using the vertex constraint and the track parameters at the vertex position are computed ($\overrightarrow{P_n^m}$). Each vertex track m corresponds to a given track k of the set K of reconstructed and preselected tracks. For each track k a fit is performed without vertex constraint giving the parameters $\overrightarrow{\pi_k}$ at the first measured point.

The vertices are obtained by minimising a global fitness function defined as:

$$E[S_{n,k}, \overrightarrow{x_n}, \overrightarrow{P_n^m}, \overrightarrow{\pi_k}] = \sum_{n\in N, k\in K} D_{n,k} S_{n,k} + \lambda \sum_{k\in K} [1 - \sum_{n\in N} S_{n,k}]^2 \ , \tag{8.1}$$

where $S_{n,k}$ is a binary decision unit [8-5] and $D_{n,k}$ is a distance between the k-th reconstructed track and the n-th vertex. The weight $S_{n,k}$ is set to zero when the track k is not related to any vertex track of the set M_n; the 'noise' parameter λ characterises the weight of the second term of Eq. (8.1), which imposes a penalty when some track remains unmatched or is associated to more than one vertex. The term $D_{n,k}$ is defined only for tracks pointed by a vertex track of the set M_n as:

$$D_{n,k} = \sum_{i,j} (R^i_{n,m} - \pi^i_k) A^k_{ij} (R^j_{n,m} - \pi^j_k) \ , \tag{8.2}$$

where $\overrightarrow{R}(\overrightarrow{P_n^m}, \overrightarrow{x_n})$ are the parameters of the m-th vertex track propagated to the first measured point of the corresponding reconstructed track $k = k(n, m)$. Here A^k is the inverse covariance matrix of the reconstructed track parameters at this point.

To further constrain vertex identification, specific vertex properties can be used by modifying the definition of the distance parameter $D_{n,k}$ used in the fit:

$$D_{n,k} \to C_n (D_{n,k} + D_1) \ , \tag{8.3}$$

where C_n is a constant. The term D_1 brings in the additional constraint. For primary vertices D_1 takes the form:

$$D_1 = A_n (X_n^2 + Y_n^2) \ , \tag{8.4}$$

where A_n represents the inverse of the variance of the collision point in the transverse plane. For V^0's, D_1 is proportional to the squared difference between the reconstructed effective mass and the candidate particle mass.

The constant λ has to be large ($\sim$ 50–100) and it decreases slightly during the minimisation.

Two different approaches have been tested for the minimisation with similar performances: the empirical EM algorithm [8-6] and the simulated annealing algorithm [8-7]. The two approaches perform well in a simple environment, but they fail often when the track density is very high as in the core of a hard jet.

8.3 Calibration with reconstructed tracks

The ultimate position calibration (alignment) will be made using reconstructed tracks. The method uses the best in situ available measurement device, namely the detectors themselves. The smoothness of high p_T tracks introduces a strong position correlation between hit detector elements. It is this correlation which allows the detector elements to be aligned with respect to each other on a statistical basis.

The basic requirements to make track finding and initial reconstruction possible are discussed in Section 8.3.2 below. This simulation study shows [8-8] that a rather loose precision requirement, a few hundred microns, is sufficient for track finding and initial reconstruction of isolated tracks with a 90 % efficiency. Thanks to the precision assembly followed by optical survey of the fiducial marks, the detector positions will be determined to such precision thus providing the necessary starting point for the alignment by reconstructed tracks. In the next sections we describe the strategies and methods of these calibration procedures.

8.3.1 Calibration precision requirements

The precision requirements for the ultimate calibration of the detector positions must be of the order of the intrinsic detector resolution or better. To be more precise let us denote the detector resolution by σ_D and the r.m.s. calibration precision by σ_C and their ratio as $k = \sigma_C/\sigma_D$. Then the effective precision of the detectors is:

$$\sigma_{eff} = \sqrt{\sigma_D^2 + \sigma_C^2} = \sqrt{1+k^2}\ \sigma_D\ , \tag{8.5}$$

where $\sqrt{1+k^2} = \sigma_{eff}/\sigma_D$ is called the 'deterioration factor'. The dependence is shown graphically in Fig. 8.1. For example, allowing a maximum of 30 % deterioration of the effective detector precision, implies that the calibration precision should be better than about 80 % of the detector resolution:

$$\sigma_C \leq 0.8\ \sigma_D\ . \tag{8.6}$$

It can be shown [8-9] that the precision of the fitted track parameters scales with the effective detector resolution at large p_T (in the absence of multiple scattering): scaling of the effective detector resolution by a constant factor, scales the precision of the track parameters by the same factor. Hence in order to achieve less than 30 % (20 %) worsening of the p_T resolution from ideal, the detectors must be aligned to better than about 80 % (65 %) of their intrinsic resolution.

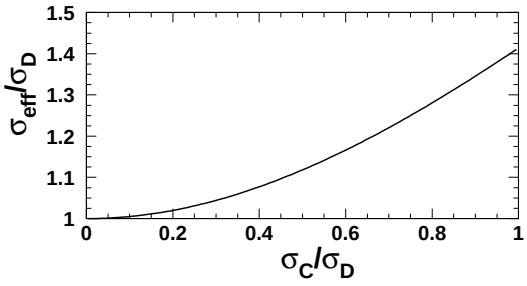

Fig. 8.1: Effective detector precision as a function of calibration precision.

This is a simplified picture of a complicated problem but gives a good idea of the required calibration precision. In reality there are 6 constants which define the position (3 coordinates) and orientation (3 angles) of a detector element. In principle all the 6 constants need calibration. Therefore of the order 10^5 constants are needed to calibrate the full detector which has about 20 thousand detector elements.

A detailed simulation analysis of the precision requirements has been performed [8-8]. The required precisions were deduced for the radial and longitudinal coordinates as well. These results are presented in the following.

8.3.2 Simulation study for requirements

Precision requirements for the Tracker alignment have been studied by a two-stage simulation procedure. In the first stage random offsets from ideal positions were applied to detector modules. Hits in the distorted detector were simulated and the results were stored in a separate file. In the second stage the offset hits were used for track reconstruction with a non-distorted detector description. The reconstruction was performed with the Global Track Finder.

The event samples used in the requirements study were single muons in the energy range 1–200 GeV/c^2 and Higgs events in the decay channel H$\rightarrow$ ZZ $\rightarrow 4\mu$ with 150 GeV/c^2 $< M_H <$ 400 GeV/c^2. Higgs events were used here for practical reasons only. The momentum cut for

muons from Z^0 decays was 5 GeV/c. The following quantities were extracted as a measure of the reconstruction quality: 1) reconstruction efficiency, the fraction of reconstructed tracks associated with initial muons within a corridor in ϕ, θ and momentum; 2) number of hits associated with reconstructed tracks; 3) χ^2/N_D averaged over all tracks; 4) r.m.s. of $(p_T - p_{Tr})/p_T$, where p_T is the true transverse momentum and p_{Tr} is the reconstructed one; 5) r.m.s. of the impact parameter distribution. Two levels of precision requirements for alignment have been considered:

1. precision required in installation and survey,
2. ultimate precision needed in reconstruction.

The first level is set to allow pattern recognition of isolated high p_T tracks with reasonable efficiency. The second level of precision allows the ultimate event reconstruction to be made with the required resolution (see below). At the first level of precision we impose two conditions on the $Z^0 \to \mu\mu$ events: i) the muon reconstruction efficiency has to be more than 0.9; ii) the number of associated hits in each subsystem has to be more than half of the detector layers in this subsystem. These requirements give a starting point for further alignment with reconstructed tracks to obtain precision which provides good performance of the Tracker.

At the second level the following conditions must be satisfied:

1. mean χ^2/N_D less than 1.2 of that with the non-distorted geometry,
2. less than 20 % deterioration of the impact parameter resolution,
3. less than 20 % deterioration of the r.m.s. of $(p_T - p_{Tr})/p_T$.

These requirements were applied to the analysis of Z^0 events and to single muons in the barrel region of the Tracker.

Random displacements of the detector positions were introduced separately for each subsystem in r, ϕ and z with a Gaussian distribution. As an example of the results, the reconstruction efficiency for MSGC *vs* distortion-r.m.s. for the Z^0 events is shown in Fig. 8.2.

Using these and similar results for Silicon and Pixel detectors, and following the criteria described above, the level-1 set of requirements is given in Table 8.2. For both MSGC and Silicon detectors the sensitivity to distortions in r and z appears to be quite low for isolated muon tracks with transverse momentum in the range between 5 GeV and 100 GeV.

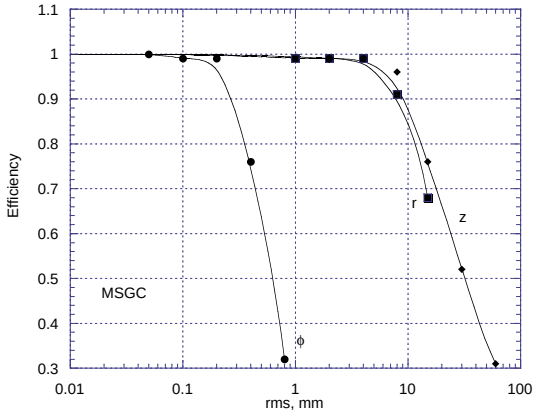

Fig. 8.2: Reconstruction efficiency *vs* distortion r.m.s. for the MSGC barrel.

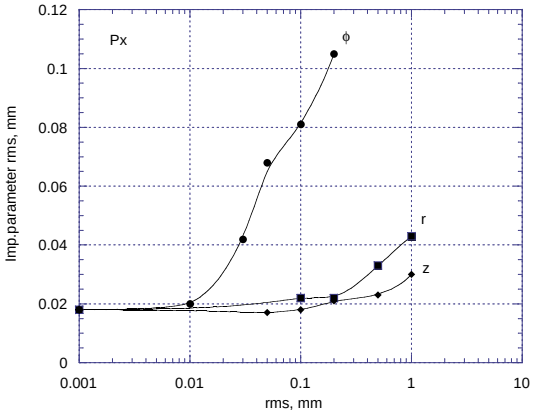

Fig. 8.3: The impact parameter resolution *vs* r.m.s. of the distortion.

Table 8.2: Precision requirements (r.m.s.) at level 1

System	r [mm]	$r\phi$ [mm]	z [mm]
MSGC	8	0.3	10
Silicon	4	0.2	10
Pixel	0.6	0.1	0.8

Table 8.3: Precision requirements (r.m.s.) at level 2

System	r [mm]	$r\phi$ [mm]	z [mm]
MSGC	0.6	0.05	2
Silicon	0.3	0.015	0.5
Pixel	0.1	0.015	0.5

Figure 8.3 shows an example of impact parameter deterioration due to distortion of the Pixel wafers. The full set of required precisions at level 2 is given in Table 8.3.

The sensitivity is again rather low for r and z suggesting that the necessary alignment precision in these coordinates can be obtained by precision installation and survey.

The results presented are considered preliminary particularly for the second-level requirements, because the relationship between general performance parameters, such as χ^2, momentum resolution and impact parameter, and physics performance parameters, such as B-tagging efficiency or reconstruction efficiency of muon in a jet, is not yet precisely established. The deterioration limit of 20 % which was used for all general performance parameters was chosen rather arbitrarily and can be changed when our knowledge about necessary physics performance parameters improves. For the moment we consider the second level of requirements as the minimum precision that should be achieved with the position monitoring system. These requirements are likely to change with the update of the results of the ongoing simulations.

8.3.3 Calibration strategies

Local and global calibration procedures are foreseen. The local calibration aims to align the detectors within a layer using the overlap feature. Tracks down to 1 GeV/c in p_T will be used for this purpose. The global calibration aims to perform longer range alignment of the detectors. High p_T tracks will be used and the most suitable for this purpose are muons from the Z^0 decay.

A number of constraints will be used in the calibration procedure. One of the most important constraints available at the LHC is the beam spot as a precise (about 10 μm in the transverse plane) origin of tracks. Another constraint arises from the precision-assembly: in detector units like rods the individual detectors are assembled very precisely with respect to each other on optical benches. The relative positions of the N detectors in such an assembly unit should remain constant apart from bending and twist effects when taken from the laboratory and assembled in the experimental area. In such a case the $6N$ calibration constants reduce to $6+3$ per unit, where the three special constants would account for possible bending and twist of the unit.

Local calibration

The detector elements in a given layer are arranged in such a way that a small part, typically a few per cent of the detector area, of the neighbouring detectors are overlapped. This means that occasionally a track traverses two nearby detectors in a layer which helps to align the detectors. Multiple scattering reduces the position correlation by introducing an error Δv according to the formula:

$$\Delta v = \frac{13.6\,\mathrm{MeV}}{p}\Delta r\sqrt{h_0}\,, \tag{8.7}$$

where p is the momentum, h_0 is the effective thickness of the detector in radiation lengths and Δr is the distance between the two detectors which are in overlap. For example for the

values $h_0 = 0.025$, $p = 1\,\mathrm{GeV}/c$ and $\Delta r = 3\,\mathrm{cm}$ we get for the multiple scattering fluctuation $\Delta v \approx 70\,\mu\mathrm{m}$. This means that in order to obtain $10\,\mu\mathrm{m}$ calibration precision we need at least $(70/10)^2 \approx 50$ overlapping tracks per detector for the alignment when using $\geq 1\,\mathrm{GeV}/c$ tracks.

The time needed to collect enough events for calibration of the whole Tracker is determined by those parts of the detector which get the smallest particle flux. These are the outermost MSGC detectors in the central region. Assuming $\mathrm{d}N/\mathrm{d}\eta = 8$ charged particles per rapidity unit and that a fraction Q of the particles is used for calibration, the flux of useful particles per one pp event at $\theta = \pi/2$ and $r = 113\,\mathrm{cm}$:

$$\frac{\mathrm{d}N}{\mathrm{d}A} = \frac{Q}{2\pi r^2}\frac{1}{\sin\theta}\frac{\mathrm{d}N}{\mathrm{d}\eta} \approx 1.0 \times 10^{-4} Q\,\mathrm{cm}^{-2}\,. \tag{8.8}$$

Then for the overlap area of $2\,\mathrm{cm}^2$ per detector pair, the flux of particles for calibration is $2 \times 10^{-4} Q$ per pp interaction. Some $Q = 10\%$ of the particles have $p_T \geq 1$ GeV/c. These numbers imply that, in order to obtain 100 particles per detector pair for calibration, the number of pp events needed is of the order 5×10^6. Events from any physics trigger can be used for local calibration, so that detector alignment does not disturb the physics program of the experiment.

The time needed to collect sufficient statistics does not scale inversely with the luminosity, but depends rather on the data acquisition capacity. For the DAQ rate of 5 Hz at high luminosity (≈ 20 pp events per trigger) sufficient statistics will be collected during a day of running. At low luminosity ($10^{33}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}$) the DAQ rate/trigger is higher due to smaller events and sufficient statistics may be obtained in a few days of running the experiment. A once-a-week frequency of the local calibration procedure is perfectly feasible, but only experience will tell how often it will be necessary.

Global calibration

Bending and twist effects will be parametrised and the values of the parameters will be obtained from measurements made in the laboratory (e.g. bending of the MSGC rods). Global sag of the detector will be obtained from the survey measurements and the monitoring system. These pre-determined parameters will be used either as such or as useful constraints in the calibration procedure by reconstructed tracks.

Muons from $\mathrm{Z}^0 \to \mu^+\mu^-$ decays above a p_T limit will be used for global alignment. The number of such events is estimated to be about 10^4, which is sufficient to obtain the related calibration constants with required precision. The cross section is estimated as:

$$\sigma_{\mathrm{Z}\to\mu\mu}(p_T(\mu) > 20\,\mathrm{GeV}) \approx 0.4\,\mathrm{nb}\,. \tag{8.9}$$

Hence the rate of such events is 4 Hz (0.4 Hz) and the necessary event collection time is about 10 min (2 hours) at the luminosity $\mathcal{L} = 10^{34}$ cm^{-2}s^{-1} ($\mathcal{L} = 10^{33}$ cm^{-2}s^{-1}). In view of these estimates the global calibration can be performed on a daily basis if necessary.