The Compact Muon Solenoid Experiment
Conference Report

Mailing address: CMS CERN, CH-1211 GENEVA 23, Switzerland

24 October 2011 (v3, 02 November 2011)

Software package for the characterization of Tracker layouts

S. Mersi, D. Abbaneo, N. De Maio, G. Hall

Abstract

The high luminosity operation of the LHC will require an upgrade of the CMS silicon tracker, possibly implementing trigger capabilities. In order to evaluate the possible options and geometries, a standalone software package has been developed (tkLayout) to generate detector layouts, evaluate the effect of inactive material and provide an a priori estimate of the tracking performance. The package can be used to compare the performance of different options, and then to optimize the chosen detector concept; tkLayout is not specific to CMS, thus it can be adapted to design studies for other tracking detectors. The technology of tkLayout is presented, along with some results obtained in the context of the CMS Tracker design studies.

Presented at *ICATPP 2011: 3th ICATPP Conference on Astroparticle, Particle, Space Physics and Detectors for Physics Applications*

Software package for the characterization of Tracker layouts

S. Mersi[†], D. Abbaneo[†], N. De Maio[†], G. Hall[‡]

On behalf of the CMS collaboration

[†] *European Organization for Nuclear Research*

CERN CH-1211

Genève 23, CH

[‡] *Imperial College London*

South Kensington Campus

London SW7 2AZ, UK

The high luminosity operation of the LHC will require an upgrade of the CMS Silicon Strip Tracker, possibly implementing trigger capabilities. In order to evaluate the possible options and geometries, a standalone software package has been developed (tkLayout) to generate detector layouts, evaluate the effect of inactive material and provide an a priori estimate of the tracking performance. The package can be used to compare the performance of different options, and then to optimise the chosen detector concept; tkLayout is not specific to CMS, thus it can be adapted to design studies for other tracking detectors. The technology of tkLayout is presented, along with some results obtained in the context of the CMS Tracker design studies.

Keywords: Tracking; Design; CMS; HL-LHC.

1. Overview

The Silicon Strip Tracker^{1,2} currently operating in CMS³ has just collected data for an integrated luminosity of about 5 fb^{-1} and the LHC performance is expected to grow in the next year. After delivering about 500 fb^{-1} the machine is expected to undergo a major upgrade in the early 2020s, after which its instantaneous luminosity should exceed the design goal, eventually reaching $5 \times 10^{34} \text{ cm}^{-2} \text{ s}^{-1}$. This scenario is known as High-Luminosity LHC (HL-LHC).

This scenario represents a challenge for the CMS detector due to the high radiation dose and number of pile-up events (up to 100 to 200), which will adversely affect the L1-Trigger.

Together with the HL-LHC upgrade the Silicon Strip Tracker of CMS should be replaced to cope with the charged particle density and the collaboration is therefore studying the option of instrumenting the upgraded tracker with detector modules capable of measuring a track transverse momentum^{5,6} (p_T) locally and sending high- p_T hits to a real-time tracking processor embedded in the Level-1 trigger.

With this design, the upgraded tracker should provide more functionality than the present one, and yet the tracking resolution of the current detector is limited by its amount of material. This poses a key question in the design of the tracker upgrade: what is the optimal trade-off between adding functionality (like high- p_T real-time tracking) and improving tracking performance? What is the impact of different design choices on the final detector performance?

2. A detector layout design software

When designing a new detector for high energy physics it is common practice to rely on detailed (and complex) Monte Carlo simulations. While this cannot be avoided for the qualification of a detector design, this approach needs a lot of effort to understand simulation details and to optimise event reconstruction algorithms.

It is therefore useful to evaluate the potential performance of a design by estimating the track parameter resolution from first principles.

For this reason a software tool was developed capable of creating the full three-dimensional description of a tracker starting from a small set of design parameters as described in Section 2.1. This was complemented with a customisable model of material, as shown in Section 2.2. Finally, the combined information on material amount and sensor properties is used to estimate the detector's potential performance in terms of tracking (Section 2.3).

2.1. *Layout creation*

The first functionality of this software is to place detector modules in three-dimensional space, so hermeticity is assured. A small number of plots are then produced for consistency checks.

The detector geometry is built starting from a few basic parameters. Modules can be arranged either in barrel layers or in end-cap disks. Here a right-handed system is used with the y axis vertical and the z axis aligned with the beam line.

Barrel modules are represented by rectangles of arbitrary size and are

Fig. 1. Placement of modules in the transverse (left) and longitudinal (right) planes. Figure is not to scale.

arranged in an arbitrary number of layers (optionally automatically adjusting the layer radii to make an optimal use of the detector surface). The length of the barrel is determined by the number of consecutive modules arranged in the z direction.

Detectors are staggered in the radial direction to avoid volume collisions, by requiring a small radial gap d_1 between modules on the same layer at the same φ and different z and a larger radial gap d_2 between modules on the same layer at the same z and different φ .

Hermetic coverage along z direction is guaranteed by placing adjacent modules so that an overlap of at least d_3 would be seen from the origin $(0, 0, 0)$ and no gap would be seen from $(0, 0, \pm\Delta z)$, where Δz is the expected variance of the z coordinate of the primary interaction points. Hermetic coverage in φ is guaranteed by placing adjacent modules with an overlap of at least d_3 along this direction (see Figure 1).

End-cap modules are represented by rectangles or wedge shapes and are placed in rings staggered in z by a small gap d_1 . Rings are staggered in z between each other by a larger gap d_2 . All the disks are built so that a small overlap d_3 is seen between all adjacent modules by straight tracks passing through the origin. All disks are built with the same algorithm (to simplify the construction) and they are designed providing hermetic coverage for tracks starting from the origin, regardless if the ring is placed near to or far from the interaction point (within the specified z range).

All modules are then assigned a set of parameters (number of strips, power consumption, etc.). and the expected detector occupancy is estimated with a simple extrapolation from the present CMS Silicon Strip Tracker. A *specific occupancy* parameter was defined ($\pi_{B,E}(r)$) for barrel and end-cap modules separately as the number of hits per unit of polar angle per unit of eta $\pi_{B,E}(r) = n_{B,E}(r)/(\Delta\varphi\Delta\eta)$. Using a standard simulation of the

current tracker with minimum bias events, a second order polynomial was used to fit $\pi_{B,E}$. This fit (scaled by the expected number of minimum bias events per collision) is then used to predict the expected occupancy in an arbitrary layout.

Hermetic coverage is confirmed by using straight lines originating from vertices placed on the beam line around the origin with a Gaussian distribution in z .

All summary information (number of modules, total power consumption, total number of channels, expected number of measurement points as a function of η , expected occupancy, ...) is stored in form of a mini-web site.

2.2. Model for the material budget

The material amount implied by detector modules is assigned to each module volume without any detail of the geometric distribution of material within the module itself. Material due to services running inside detector layers is also assigned to the module volumes, as these are hermetically covering the detection surfaces.

Several objects can be defined for each module type to represent the services reaching the module from the end-flange (A_i , like power cables and optical connection) and the components of the module itself (B_j , like sensors, supporting mechanics, cooling pipes). Each object is defined by its material composition (1 g Copper + 0.5 g PVC + ...).

The actual material assigned to a module depends on its position in the supporting structure: in a barrel layer the amount of material M is

$$M(n) = n \sum_i A_i + \sum_j B_j \quad (1)$$

where n is the module position on a rod ($n = 0$ for modules at $z \simeq 0$).

This parametrisation takes into account the accumulation of running services towards the end of a barrel layer: a module next to the end-flange will contain in its volume as many service lines as the number of modules it has in front. A similar computation is done for end-cap rings, with an additional scaling factor for the accumulation of services due to the fact that service density decreases when these spread outward from one module ring to the next.

Additionally each material A_i and B_j can be defined as “local” if it only adds to the material inside detecting layers (like support mechanics or silicon sensors) or “exiting” if it implies the presence of some other services

Fig. 2. Distribution of material (interaction length) in an (r, z) section of a tracker model. The accumulation of material along the barrel layers (a) is indicated by the arrows (not evident in this gray scale). The accumulation of routed services (b) is evident.

coming from outside (like cooling pipes or power lines).

Once the material of modules and services in the detecting layers is in place, additional volumes representing support structures and services running out of the detecting surfaces are created.

Materials in the service volumes are automatically created depending on the amount of material previously defined as “exiting” in the facing detecting layers with some configurable conversion rules (for example many small cooling pipes will join through a manifold into fewer larger pipes exiting). Part of these services are automatically routed up to the edge of the tracking volume with the same mechanism described above for services inside the detecting layers (for example the cooling pipes will be propagated, while the material of their manifolds will not). Additionally, some fixed amount of material can be added to the service volumes to represent specific objects. The routing procedure is sketched in Figure 2.

After the material assignment is performed, some summary plots are created: the η distribution of material, photon conversion and nuclear interaction probabilities, etc.

2.3. Performance estimation

The accuracy of the track parameters derived from a fitting procedure is described here, taking into account the precision of the measurement points and multiple scattering. This is done by considering two distinct fits: a circle in the (r, φ) plane and a straight line in the (r, z) plane. In reality these are not independent, but this approximation was proven to be valid, *a posteriori*, by a comparison of the results derived with a full simulation of the CMS Silicon Strip Tracker. These calculations closely followed those pub-

lished^{7,8} by Karimäki, with two main differences: first multiple scattering is taken into account here, while the author explicitly neglects it in the cited article; second we are only interested in the general solution of the problem, as tkLayout performs the computation for each particular case.

2.3.1. *Fit in the (r, φ) plane: a circle*

We consider a system of cylindrical coordinates (r, φ, z) , centred on the interaction point. In the hypothesis of uniform magnetic field and no multiple scattering, the projection of the track on the plane with $z = 0$ will be a circle. If the actual measured N points are $P_i = (r_i, \varphi_i)$ the error ε_i of the measurement point i will be given by:

$$\varepsilon_i = \frac{1}{2}\rho r_i^2 - (1 + \rho d)r_i \sin(\varphi_i - \varphi_0) + \frac{1}{2}\rho d^2 + d \quad (2)$$

where r_i is the sensor radial position, φ_0 is the initial track polar angle in the transverse plane, φ_i is the polar angle of each hit, ρ is the track curvature ($\rho = 1/R$, with R radius of curvature) and d is the distance of closest approach of the track to the z axis (transverse impact parameter).

The track fitting procedure is intended to obtain the estimate $\hat{\alpha}_i$ of the track parameters $\alpha_i = \{\rho, \varphi_0, d\}$ from the set of measured points P_i . Here we will derive the explicit formulation of the covariance matrix of the estimated parameters $U_{ij} = \text{cov}[\hat{\alpha}_i, \hat{\alpha}_j]$: we are interested in the error measurement $\sigma(\hat{\alpha}_i) = \sqrt{U_{ii}}$.

It is assumed here that the best fit of the trajectory to the measured points will be given by minimising $\chi^2 = \sum_{i,j} \varepsilon_i W_{ij} \varepsilon_j$ where $W = U^{-1}$ is the weight matrix. This is given by $W = D^T C^{-1} D$, where $D_{ij} = \partial \varepsilon_i / \partial \alpha_j$ and C is the covariance matrix of the measured points $C_{ij} = \text{cov}[\varepsilon_i, \varepsilon_j]$.

The matrix D can be derived from (2): $\partial \varepsilon_i / \partial \alpha_j \simeq \{r_i^2/2, -r_i, 1\}$. The approximation is not strictly necessary for the computation, but it is valid for tracks with bending radii much larger than the tracker radius ($p_T \gg 1 \text{ GeV/c}$ for CMS) and it makes the matrix D only dependent on the detector geometry as seen by the track.

If one rotates the reference frame by φ_0 , the track is directed along the new x axis at the origin and the coordinate measured by the sensors is y . If the sensor radial positions r_i are taken to be error-less and for high-momentum tracks ($\rho d \ll 1$) we have $d\varepsilon_i \simeq -dy_i$ and thus $C_{ij} = \text{cov}[y_i, y_j]$.

The particle multiple scattering can be considered a deviation from the ideal track to be measured, and thus it can be treated as a measurement error. Hence, the matrix C can be written as $C = C^M + C^R$, with C^M

the covariance matrix due to multiple scattering and $(C^R)_{ij} = \delta_{ij}\sigma(y_i)$ the covariance matrix due to the intrinsic resolution of the measurement point P_i , which is taken as an input by tkLayout.

The matrix elements $(C^M)_{ij}$ can be evaluated by first computing the larger matrix $(\tilde{C}^M)_{mn} = \text{cov}[\tilde{y}_m, \tilde{y}_n]$ representing the covariance matrix of the impact points of a straight track traversing M perpendicular planes, where it can interact through multiple scattering (not only the measurement planes). The matrix C^M can be obtained from $\tilde{C}^M$ by dropping the $M - N$ lines and columns corresponding to the interaction of the particle with a non-sensitive element.

Given the radial positions of interactions $r_n = r_1, r_2, \dots, r_M$ with scattering angles $\vartheta_n = \vartheta_1, \vartheta_2, \dots, \vartheta_M$, the deviation from the ideal path is $\tilde{y}_n \simeq \sum_{i=1}^n (r_n - r_i) \vartheta_i$ and, since the scattering angles ϑ_n are uncorrelated

$$\tilde{C}_{mn} = \langle \tilde{y}_m, \tilde{y}_n \rangle \simeq \sum_{i=1}^{\min(n,m)} (r_m - r_i)(r_n - r_i) \langle \vartheta_i^2 \rangle \quad (3)$$

With this method a number of sample tracks are generated for each layout and for each of them the list of crossed materials and modules is obtained, so that $C = C^M + C^R$ and D can be computed. Finally the expected resolution on the track parameters can be obtained from the covariance matrix $U = [D^T C^{-1} D]^{-1}$.

2.3.2. Fit in the (r, z) plane: a line

The fit in the (r, z) plane is actually the first to be evaluated for each selected track. In this plane the tracks are taken to be straight lines (again in the $\rho d \ll 1$ approximation), thus the track equation is

$$r_i = \frac{z_i - z_0}{\text{ctg}(\theta)} \quad (4)$$

The resolution on the last two track parameters (the longitudinal impact parameter z_0 and the polar angle θ) can be evaluated with the same method described above on the simpler linear fit. The error on z is taken from the longitudinal resolution Δz for the barrel modules. For the end-cap modules an effective $\Delta z = \Delta r \cdot \tan(\theta)$ is assigned, where Δz is the module's longitudinal resolution.

2.3.3. Notes on end-cap modules

In Section 2.3.1 it was assumed that r_i is known for all the hits and y_i is the only source of uncertainty. This is a good approximation only for

barrel modules, while for end-cap modules z_i is the known coordinate of the sensor. In the actual implementation we assumed that during the fit procedure the parameter $\text{ctg}(\theta)$ is obtained first (with its error) and then r_i is obtained through (4). To the first order approximation this can be translated into an effective $(\Delta y)_1$ error on y_i through:

$$(\Delta y)_1 = \rho r_i \left[r_i \frac{\Delta \text{ctg}(\theta)}{\text{ctg}(\theta)} \right] \quad (5)$$

Also in case of wedge-shaped end-cap modules, the actual measured coordinate is $y = r\varphi$, as for barrel modules, but in case of square-shaped modules an additional error is present (also depending on $\Delta \text{ctg}(\theta)$) and is taken into account in a similar way to $(\Delta y)_2$. For end-cap modules the effective resolution is taken to be $\sigma_{\text{eff}}^2(y_i) = \sigma^2(y_i) + (\Delta y)_1^2 + (\Delta y)_2^2$.

3. Model validation

The current CMS Silicon Strip Tracker was used as a bench-mark to test the tracking performance predicted by tkLayout. A layout was created with the same number of barrel layers and end-cap disks. The material model was tuned in order to reproduce the Outer Barrel material and it was then applied to the whole tracker layout. A smaller inner tracker was also generated to represent the pixel⁹ detector. The correct strip pitch p_i was assigned to all of the strip tracker sensors and the resolution $\sigma_i^2 = p_i^2/12$ was taken to be that of a binary readout system. The resolution of the pixel detector was instead assigned explicitly to match the actual detector. No further tuning was performed.

The modelling done by tkLayout produces a layout similar to that of the actual CMS Outer Barrel, so it fails to correctly model the Inner Barrel (these details are described in Ref. 2). Even more notably the peculiar design of the CMS Tracker, with a smaller end-cap inserted inside the Outer Barrel is not intended to be modelled in tkLayout, which thus fails to reproduce the actual service routing for this sub-detector.

The actual CMS tracker is accurately simulated in the official software of the Collaboration, based on GEANT4, which was validated against the collision data collected. The material amount (measured in radiation lengths) obtained from the full simulation is compared in Figure 3 with the same quantity estimated with tkLayout. The material amount is correctly reproduced at low η and at the material peak ($\eta = 1$ to 1.4). The accuracy in measuring p_T of a muon with $p_T = 10 \text{ GeV}/c$ is shown in the same figure. The tkLayout estimation matches closely the full simulation in the η range

Fig. 3. Material amount (left) and expected resolution on p_T for muons with $p_T = 10$ GeV/c (right) as a function of pseudorapidity η . Comparison between official CMS software full simulation (gray) and tkLayout estimate (black).

where the material amount is correctly reproduced.

A complete comparison between the resolution on track parameters estimated by tkLayout and by the full simulation is shown in Table 1. Two bench-mark cases are considered here: muons with $p_T = 10$ and 100 GeV/c. The estimated uncertainty of the parameter fit is averaged for tracks with η in the ranges shown below. The uncertainty is correctly reproduced by tkLayout with an error of around 10% to 20%.

Table 1. Residuals = (tkLayout / FullSim) - 1

η	$p_T = 10$ GeV			$p_T = 100$ GeV		
	0 to 0.8	0.8 to 1.6	1.6 to 2.4	0 to 0.8	0.8 to 1.6	1.6 to 2.4
p_T	-12 %	-19 %	-20 %	11 %	-0.5 %	-5 %
d_0	7 %	13 %	15 %	17 %	35 %	41 %
φ	12 %	12 %	10 %	13 %	16 %	7 %
$\text{ctg}(\theta)$	14 %	10 %	14 %	13 %	17 %	56 %
z_0	13 %	7 %	5 %	17 %	15 %	22 %

4. Conclusions

A generic analytic method to evaluate the accuracy of a tracking device was derived. This allows to compute the full covariance matrix of the track parameters for any configuration of the detector with respect to the track. A software tool (tkLayout) was developed, capable of describing a tracker detector geometry from few basic parameters. A simple model of the ma-

terial budget was also implemented, which allows a simple and coherent definition of detector materials and automatically takes into account the service routing in the tracking volume. TkGeometry also implements the mentioned estimation of tracking resolution and it was validated against a full simulation of the CMS tracker. The accuracy of tkLayout was proven to be around 10% to 20%.

This software is currently used within CMS to evaluate possible tracker layout concepts, and it proved to be specially useful in order to make a fair comparison between different design approaches.

References

1. *CMS: The Tracker Project Technical Design Report* 1998. CERN-LHCC-98-06.
2. *Addendum to the CMS Tracker TDR* 2000. CERN-LHCC-2000-016.
3. G. L. Bayatian *et al.* CERN-LHCC-2006-001.
4. F. Hartmann, *Evolution of Silicon Sensor Technology in Particle Physics* (Springer, 2009).
5. M. Pesaresi, *PoS VERTEX2010*, p. 047 (2010).
6. M. Pesaresi and G. Hall, *JINST* **5**, p. C08003 (2010).
7. V. Karimäki, *Nuclear Instruments and Methods in Physics Research Section A* **305**, 187 (1991).
8. V. Karimäki, *Nuclear Instruments and Methods in Physics Research Section A* **410**, 284 (1998).
9. C. Amsler, K. Bosiger, V. Chiochia, W. Erdmann, K. Gabathuler *et al.*, *JINST* **4**, p. P05003 (2009).