A KALMAN FILTER FOR TRACK-BASED ALIGNMENT

E. Widl*, R. Frühwirth†, W. Adam‡
Intitute for High Energy Physics, Austrian Academy of Sciences, Vienna, Austria

Abstract

An iterative method for track-based global alignment is proposed. It is derived from the Kalman filter and is designed to avoid the inversion of large matrices. The update formulas for the alignment parameters and for the associated covariance matrix are described. The implementation and the computational complexity is discussed, and it is shown how to limit the latter to an acceptable level by restricting the update to detectors that are close in the sense of a certain metrics. The performance of the Kalman filter in terms of precision and speed of convergence is studied in a simplified setup. First results from an implementation in the CMS reconstruction program ORCA are presented, using a subsection of the barrel part of the CMS Tracker.

INTRODUCTION

We describe here an iterative method, derived from the Kalman filter [1, 2, 3], for the global alignment of large tracking systems using charged tracks. The method is iterative, because the alignment parameters are updated after each track. The method is global, because the update is not restricted to the detector units that are crossed by the track. In case the number of detector units to be aligned is very large, it is proposed to limit the update to those detector units that have significant correlations with the ones in the current track trajectory. In this case it turns out that no large matrices have to be inverted. However, a certain amount of bookkeeping is required in order to restrict the computational load to an acceptable level.

Alignment with tracks requires an already aligned and fixed reference system. All updates are relative to this reference system. The problem of obtaining such a reference system is not discussed here.

In the formalism that is proposed here it is possible to use prior information about the alignment obtained from mechanical and/or laser measurements. It is also possible to fix the position of certain detector units by giving them a large prior weight (small prior uncertainty). A requirement that several detectors move along with each other can be enforced by large prior correlations.

A more detailed discussion of the algorithm and its properties as well as results from various simulation studies can be found in [4].

* Edmund.Widl@oeaw.ac.at
† Rudolf.Fruehwirth@oeaw.ac.at
‡ Wolfgang.Adam@cern.ch

SEQUENTIAL UPDATING OF ALIGNMENT PARAMETERS

Assume that there exist alignment parameters $\boldsymbol{d}$ with a covariance matrix $\boldsymbol{D}$. They are to be updated by using information from a track with track parameters $\boldsymbol{x}_\mathrm{t}$ and observations $\boldsymbol{m}$. The observations $\boldsymbol{m}$ of a track depend on the track parameters $\boldsymbol{x}_\mathrm{t}$ via the track model $\boldsymbol{f}$:

$$\boldsymbol{m} = \boldsymbol{f}(\boldsymbol{x}_\mathrm{t}) + \varepsilon, \quad \mathrm{cov}(\varepsilon) = \boldsymbol{V}.$$

The stochastic vector ε contains the effects of the observation error and of multiple scattering. Its variance-covariance matrix $\boldsymbol{V}$ can be assumed to be known. It is essential that a preliminary track fit is performed in order to get a provisional estimate $\boldsymbol{x}$ of the track parameters and of the momentum in particular. As the tracks used for alignment are mainly high-energy minimum-ionizing particles, energy loss can be considered as deterministic and is dealt with in the track model.

If the detector units are not perfectly aligned, the observations also depend on the alignment parameters $\boldsymbol{d}_\mathrm{t}$, and the track model has to be modified accordingly:

$$\boldsymbol{m} = \boldsymbol{f}(\boldsymbol{x}_\mathrm{t}, \boldsymbol{d}_\mathrm{t}) + \varepsilon, \quad \mathrm{cov}(\varepsilon) = \boldsymbol{V}.$$

As is usual, the track model is linearized by a first-order Taylor expansion at expansion points $\boldsymbol{d}_0$ and $\boldsymbol{x}_0$:

$$\boldsymbol{m} = \boldsymbol{c} + \boldsymbol{A}\boldsymbol{d}_\mathrm{t} + \boldsymbol{B}\boldsymbol{x}_\mathrm{t} + \varepsilon = \boldsymbol{c} + \begin{pmatrix}\boldsymbol{A} & \boldsymbol{B}\end{pmatrix} \begin{pmatrix}\boldsymbol{d}_\mathrm{t} \\ \boldsymbol{x}_\mathrm{t}\end{pmatrix} + \varepsilon.$$

The matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ are the derivatives w.r.t. the alignment parameters and the track parameters respectively, taken at the expansion points. The constant term $\boldsymbol{c}$ depends on $\boldsymbol{A}$, $\boldsymbol{B}$ and the expansion points. The expansion point $\boldsymbol{d}_0$ is either the nominal or the currently estimated sensor position, and the expansion point $\boldsymbol{x}_0$ is the result of a preliminary track fit.

In principle, the Kalman filter requires a prediction $\boldsymbol{x}$ of the track parameters, along with its variance-covariance matrix $\boldsymbol{C}$. This prediction has to be independent of the observations $\boldsymbol{m}$. It is conceivable that such a prediction exists, for instance as the result of a vertex and/or kinematic fit constraining some or all of the track parameters. In this case the update equation of the Kalman filter reads:

$$\begin{pmatrix}\widehat{\boldsymbol{d}} \\ \widehat{\boldsymbol{x}}\end{pmatrix} = \begin{pmatrix}\boldsymbol{d} \\ \boldsymbol{x}\end{pmatrix} + \boldsymbol{K}\left(\boldsymbol{m} - \boldsymbol{c} - \boldsymbol{A}\boldsymbol{d} - \boldsymbol{B}\boldsymbol{x}\right)$$

with the gain matrix of the filter:

$$\boldsymbol{K} = \begin{pmatrix} \boldsymbol{D}\boldsymbol{A}^T \\ \boldsymbol{C}\boldsymbol{B}^T \end{pmatrix} \underbrace{\left(\boldsymbol{V} + \boldsymbol{A}\boldsymbol{D}\boldsymbol{A}^T + \boldsymbol{B}\boldsymbol{C}\boldsymbol{B}^T \right)^{-1}}_{\boldsymbol{G}}.$$

In general, however, there is no independent prediction of the track parameters. In this case, the preliminary track parameters $\boldsymbol{x}_0$ are used as the "prediction", but with zero weight in order not to bias the estimation. This is accomplished by multiplying $\boldsymbol{C}$ with a scale factor α and letting α tend to infinity:

$$\begin{aligned} \boldsymbol{G} &= \lim_{\alpha \longrightarrow \infty} \left(\boldsymbol{V} + \boldsymbol{A}\boldsymbol{D}\boldsymbol{A}^T + \alpha \boldsymbol{B}\boldsymbol{C}\boldsymbol{B}^T \right)^{-1} \\ &= \boldsymbol{V}_D^{-1} - \boldsymbol{V}_D^{-1}\boldsymbol{B}(\boldsymbol{B}^T\boldsymbol{V}_D^{-1}\boldsymbol{B})^{-1}\boldsymbol{B}^T\boldsymbol{V}_D^{-1} \end{aligned}$$

with $\boldsymbol{V}_D = \boldsymbol{V} + \boldsymbol{A}\boldsymbol{D}\boldsymbol{A}^T$. Here the Sherman-Morrison inversion formula has been used (see e.g. [5]).

The update equation of the alignment parameters can therefore be simplified to:

$$\widehat{\boldsymbol{d}} = \boldsymbol{d} + \boldsymbol{D}\boldsymbol{A}^T\boldsymbol{G}\,(\boldsymbol{m} - \boldsymbol{c} - \boldsymbol{A}\boldsymbol{d})\,.$$

Finally, the update of the covariance matrix can be calculated by linear error propagation:

$$\widehat{\boldsymbol{D}} = \left(\boldsymbol{I} - \boldsymbol{D}\boldsymbol{A}^T\boldsymbol{G}\boldsymbol{A} \right) \boldsymbol{D} \left(\boldsymbol{I} - \boldsymbol{A}^T\boldsymbol{G}\boldsymbol{A}\boldsymbol{D} \right) + \\ + \boldsymbol{D}\boldsymbol{A}^T\boldsymbol{G}\boldsymbol{V}\boldsymbol{G}\boldsymbol{A}\boldsymbol{D}.$$

As both terms on the right hand side are positive definite the left hand side is guaranteed to be positive definite as well.

IMPLEMENTATION AND COMPUTATIONAL COMPLEXITY

The total number of detector units is denoted by N. The current track crosses a certain number of detector units, denoted by k. If each of them gives a two-dimensional measurement, the dimension $n = 2\,k$ of the observation vector $\boldsymbol{m}$ is small for high-energy tracks, usually not larger than 30. The matrix $\boldsymbol{B}$ is of size $n \times 5$ and is therefore small. The matrix $\boldsymbol{A}$ is a row of N blocks $\boldsymbol{A}_i$ of size $n \times m$, where m is the number of alignment parameters per detector unit (usually 6). For each track, only k out of these N blocks are different from zero. The set of detector units crossed by the current track is denoted by $I = \{i_1, \ldots, i_k\}$. Then the matrix $\boldsymbol{A}$ has the following form:

$$\boldsymbol{A} = \left(\boldsymbol{0} \ldots \boldsymbol{0}\ \boldsymbol{A}_{i_1}\ \boldsymbol{0} \ldots \boldsymbol{0}\ \boldsymbol{A}_{i_2}\ \boldsymbol{0} \ldots\ldots \boldsymbol{0}\ \boldsymbol{A}_{i_k}\ \boldsymbol{0} \ldots \boldsymbol{0} \right).$$

Update of the Alignment Parameters

The only large matrix in the parameter update is the product $\boldsymbol{D}\boldsymbol{A}^T$. It is a column of N blocks each of which has size $m \times n$. However, only those blocks need to be computed that correspond to the detector units that have significant correlation with the ones in the current track. In order to keep track of the necessary updates, a list L_i is attached to each detector unit i, containing the detector units that have significant correlations with i. This list may contain only i itself in the beginning and grows as more tracks are processed. The length of the list can hopefully be restricted to a fairly small number, as the correlations between detector units that are far from each other tend to be small. For the update the individual lists L_i of the detector units crossed by the current track are merged to the list $L = \bigcup_{i \in I} L_i$. The computational complexity of the parameter update is of the order $|L| \cdot |I|$.

Update of the Covariance Matrix

In the beginning the covariance matrix $\boldsymbol{D}$ is block-diagonal and contains the prior uncertainty of the alignment parameters, derived from laser alignment and mechanical measurements. If required, it may also contain prior correlations between different detector units. After each track, only the blocks in the list L need to be updated. The computational complexity of the update of the covariance matrix is of the order of $|L|^2$. Restricting the size of the lists L_i is therefore of crucial importance. An algorithm is proposed in the following subsection.

Update of the Lists L_i

First, a relation "$\sim$" between two different detector units i and j is defined:

> $i \sim j \Longleftrightarrow i$ and j have been crossed by the same track.

The relation is symmetric, but not transitive. On the basis of this relation a distance between different detector units i and j can be defined:

> If $i \sim i_1 \sim i_2 \sim \cdots \sim i_n \sim j$ is the shortest chain connecting i to j, the distance is $d(i,j) = n + 1$. In particular, if $i \sim j$, then $d(i,j) = 1$.

Using this distance, the following algorithm for updating the lists L_i, $i \in I$, is proposed:

> For all $i \in I$ do:
>
> 1. For all $j \in I \setminus \{i\}$ do:
>
> For all $k \in L_j$ with $d(k,j) < d_{\max}$, add k to L_i and store $d(k,i) = d(k,j) + 1$.
>
> 2. If a detector k occurs several times in L_i, keep only the occurence with the smallest distance $d(k,i)$.

The computational complexity of the list update is of the order $|L| \cdot |I|$. It is assumed that the distance $d(k,i)$ is stored along with k in the list L_i. $d_{\max}$ is the largest distance for which correlations are deemed to be significant.

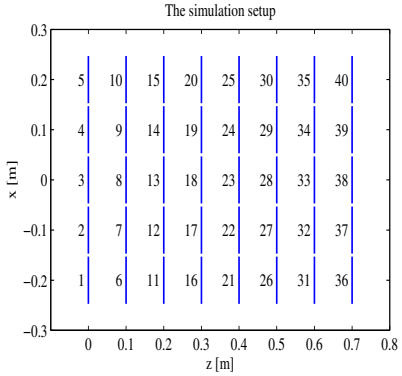

Figure 1: A schematic view of the simulation setup. The y-coordinate is chosen such that a right-handed coordinate system is formed with the shown x- and the z-axes.

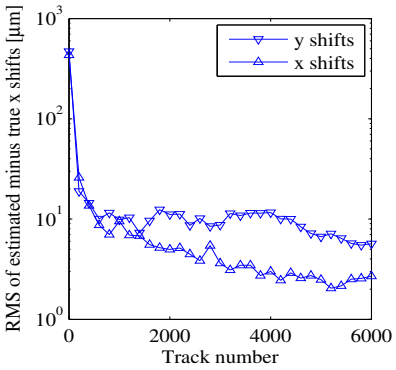

Figure 2: Evolution of the RMS of the differences between true and estimated shifts.

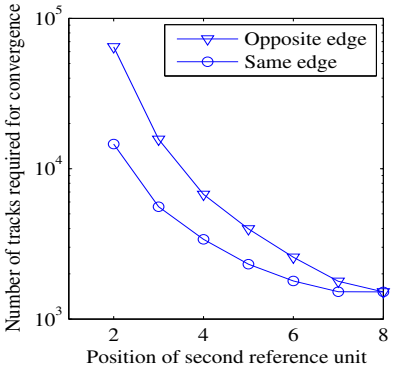

Figure 3: Number of tracks required until convergence. The reference units are at the same and opposite edge of the respective layers.

SIMULATION EXPERIMENTS IN A SIMPLIFIED SETUP

Description of the Setup

The basic properties of the method have been studied in a simple, small setup. It consists of eight detector layers along the z-axis, with a spacing of 10 cm. In each layer, there is a row of five detector units, each $10 \times 10\,\mathrm{cm}^2$ (see Fig. 1).

Straight tracks are simulated in this setup such that each track crosses all detector layers. Neither multiple scattering nor energy loss are simulated. The intersection points of the simulated tracks were smeared by adding a gaussian observation error with a standard deviation of 50 μm both in x and in y.

At least two detector units in different layers have to be fixed to define the reference frame. All detector units apart from the two reference units were misaligned by shifts in x and y. The shifts were generated randomly by drawing from a Gaussian distribution ten times as wide as the observation error. The positions of the reference units were fixed by giving them a very small prior uncertainty of the order of 0.1 μm. The prior uncertainty of the other units was set to 1 mm. All alignment parameters and the full covariance matrix were updated after each track.

Precision and Speed of Convergence

A quantitative assessment about the algorithm's precision can be made by computing the RMS of the difference δ between true and estimated shifts. The speed of convergence is measured by the number of tracks required to bring the standard errors of all estimated shifts, computed from their variance-covariance matrix, below a certain bound. In the following, we have used a bound of 10 μm.

An example of the evolution of the RMS of δ is shown in Fig. 2. In this particular run of 6000 tracks the two reference units were located in the first and last layer. The number and relative position of the reference units has a large influence on the speed of convergence. This influence has been systematically investigated by putting one reference unit in the first layer and a second reference unit in any of the other layers. Figure 3 shows the number of tracks required for convergence as a function of the position of the second reference unit.

As expected, convergence is slowest when the second reference unit is in the layer closest to the first one, and gets faster when the second reference unit is moved to more distant layers. Also, convergence is slower when the reference units are at opposite edges of the respective layres, as in this case they are rarely hit by the same track.

RESULTS FROM THE BARREL PART OF THE CMS TRACKER

Description of the Setup

A wheel-like setup (see Fig. 4) containing 156 modules from the Tracker Inner Barrel was used to investigate the convergence and stability in the environment of the CMS Tracker [6]. A prototype algorithm was implemented in the CMS reconstruction framework ORCA [7]. The concept of update lists was applied, such that at a given event the update was performed only for the modules with a hit and for the modules that were assiociated with them via the list. The update of the alignment parameters is restricted to modules with a distance of at most six from the modules in the current track, according to above definition.

The setup used the Pixel detector as the reference system, which therefore was not misaligned. The remaining modules were misaligned by shifting them in the local x- and y-directions of the detector plane, where local x is the direction perpendicular to the strips. The shifts were generated randomly by drawing from a Gaussian distribution with $\sigma = 100\,\mu$m. The prior uncertainties of the misaligned modules, i.e. the starting values of the diagonal

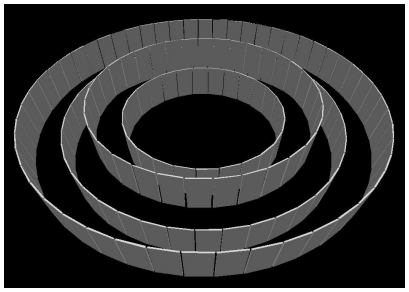

Figure 4: A schematic view of the (sub-)detector geometry.

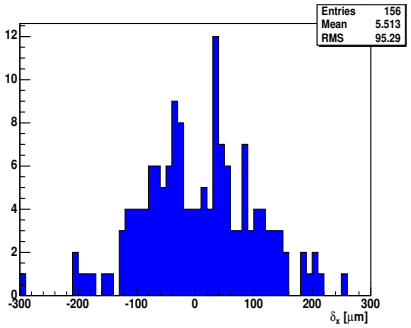

Figure 5: Residuals of the local x-shift estimates before alignment.

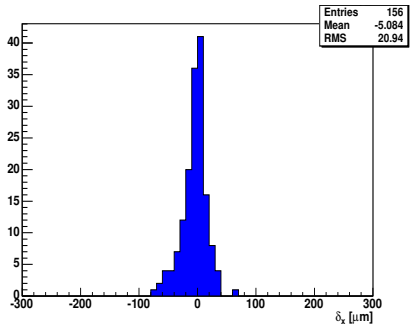

Figure 6: Residuals of the local x-shift estimates after 100,000 processed tracks.

elements of the variance-covariance matrix $\boldsymbol{D}$, have been set to 0.5 mm in x-direction and 0.5 cm in y-direction, respectively. The positions of the reference units are fixed by giving them a very small prior uncertainty of the order of 0.01 μm.

The tracks come from simulated muon and antimuon events (both with $p_T = 100$ GeV), produced with a simplified fast simulation. All simulations are done within a homogeneous 4 Tesla magnetic field and include multiple scattering effects simulated under the same hypotheses as used in the reconstruction. The intersection points are smeared by a Gaussian function using the nominal resolutions for each module type. The tracks are then reconstructed using standard algorithms [8] without any additional information about the misalignment.

Precision and Speed of Convergence

Figures 5 and 6 show the distribution of the residuals for the estimates of the x-shifts before alignment and after 100,000 processed tracks, respectively. More detailed studies show that the convergence is slower in the layers that are farther away from the reference system (the Pixel detectors). There are two reasons for this. Firstly, the bigger distance to the reference modules results in bigger uncertainties when propagating information to these layers; secondly, the modules in the outer layers are less often hit by tracks than modules in the inner layers because of the detector geometry, although they are updated with about the same frequency.

The choice of the upper distance bound $d_{\max}$ in the update list is a compromise between computational performance and the attempt to include as many correlations in the update as possible. It turns out that the choice of $d_{\max} = 6$ does not exclude dector units with significant correlations from the update. A systematic investigation reveals that a choice of $d_{\max} = 3$ gives in fact the same precision as $d_{\max} = 6$. Table 1 lists the RMS of the final alignment parameters and the time spent in the alignment algorithm for 100,000 tracks, excluding simulation and track reconstruction, as a function of $d_{\max}$. The times have been measured on a 2.2 GHz CPU, an AMD Athlon 64 Processor 3500+.

Table 1: Precision and computing time as function of $d_{\max}$.

$d_{\max}$	1	2	3	4	5	6
σ [μm]	24.75	21.38	20.97	20.95	20.94	20.94
T [s]	472	604	723	936	1152	1319

CONCLUSIONS AND OUTLOOK

Although the method has been shown to work in principle, clearly more development, testing and tuning is required to meet the challenge of a full alignment of the CMS Tracker.

The distance cut $d_{\max}$ has to be optimized; this is particularly important in view of the influence of the maximal distance on the computation time. Alternative ways of defining the range of the update will be explored. The scaleability of the algorithm has to be studied on a larger number of modules. The simplified fast track simulation has to be replaced by a full simulation. In view of the slower convergence for modules in the outer layers, alternatives to using single tracks are desirable. Using constrained muon pairs from Z- or J/ψ-decays is an auspicious possibility.

REFERENCES

[1] D.E. Catlin, *Estimation, control, and the discrete Kalman filter*. Springer, New York, 1989.

[2] R. Frühwirth, *Nucl. Instrum. and Meth.* **A 262** (1984) 444.

[3] R. Frühwirth, T. Todorov and M. Winkler, *J. Phys. G: Nucl. Part. Phys.* **29** (2003) 561.

[4] E. Widl, R. Frühwirth, W. Adam, *A Kalman Filter for Track-based Alignment*, CMS Note 2006/022.

[5] R. Zurmühl and S. Falk, *Matrizen und ihre Anwendungen*, 5th edition. Springer, Berlin–Heidelberg, 1984.

[6] The CMS Collaboration, The Tracker Project Technical Design Report, CERN/LHCC 98-6. The CMS Collaboration, Addendum to the Tracker TDR, CERN/LHCC 2000-016.

[7] The reference manual and the user guide can be found on [http://cmsdoc.cern.ch/orca].

[8] W. Adam et al., *Track reconstruction in the CMS tracker*, CMS Note in preparation.