Workshop on Tracking In high Multiplicity Environments

Zürich 3rd – 7th October 2005

Software Alignment for Tracking Detectors

Volker Blobel – Universität Hamburg

Abstract

Tracking detectors in high energy physics experiments require an accurate determination of a large number of alignment parameters in order to allow a precise reconstruction of tracks and vertices. In addition to the initial optical survey and corrections for electronics and mechanical effects the use of tracks in a special software alignment is essential. Several different methods are in use, ranging from simple residual-based procedures to complex fitting systems with many thousands of parameters. The methods are reviewed with respect to their mathematical basis and accuracy, and to aspects of the practical realization.

Keys during display: enter = next page; → = next page; ← = previous page; home = first page; end = last page (index, clickable); C-← = back; C-N = goto page; C-L = full screen (or back); C-+ = zoom in; C-− = zoom out; C-0 = fit in window; C-M = zoom to; C-F = find; C-P = print; C-Q = exit.

1. Introduction

What is alignment and calibration? . . . a web search

Alignment:
1. an adjustment to a line; arrangement in a straight line.
2. the line or lines so formed.
3. the proper adjustment of the components of an electronic circuit, machine, etc., for coordinated functioning: The front wheels of the car are out of alignment.
4. a state of agreement or cooperation among persons, groups, nations, etc., with a common cause or viewpoint.
5. a ground plan of a railroad or highway.
6. (Archaeol.) a line or an arrangement of parallel or converging lines of upright stones or menhirs.

Alignement at Kermario (Bretagne)

Purpose of instrument calibration: Instrument calibration is intended to **eliminate or reduce bias** in an instrument's readings over a range for all continuous values. For this purpose, **reference standards** with known values for selected points covering the range of interest are measured with the instrument in question. Then a **functional relationship** is established between the values of the standards and the corresponding measurements [. . . from NIST].

Alignment/calibration

Alignment/calibration of HEP track detectors:

- based **only** on track residual minimization (*incomplete* data, several degrees of freedom undefined) and on survey data;
- no "reference standards with known values" exist ($\approx$ exceptions are data from $e^+e^- \to \mu^+\mu^-$ and cosmics without field).

Methods in HEP: many papers and collaboration reports on the Web, but sometimes it is difficult to understand in detail what is really done: e.g. "The detector parameters are found via a χ^2 minimization of the residuals." " . . . constraint, pull . . . "

Alignment/calibration requires to *understand* the detector (functional relationship) and to optimize thousands or ten thousands of parameters. Aim is – after an initial optical survey and corrections for electronics and mechanical effects:

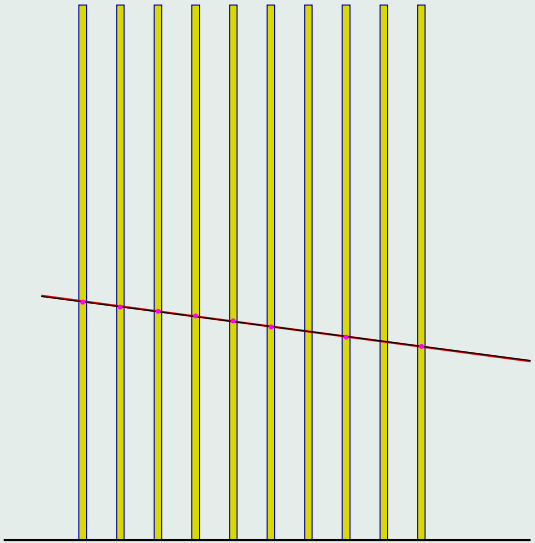
- reduce χ^2 of the track fits, in order to improve track and vertex recognition, and
- increase precision of reconstructed tracks and vertices, eliminating or reducing bias in detector data.

. . . essential for important aspects of physics analysis with large accurate vertex detectors with potential precision of a few μm.

2. Alignment of a toy detector

Test of alignment method with a MC toy track detector model:

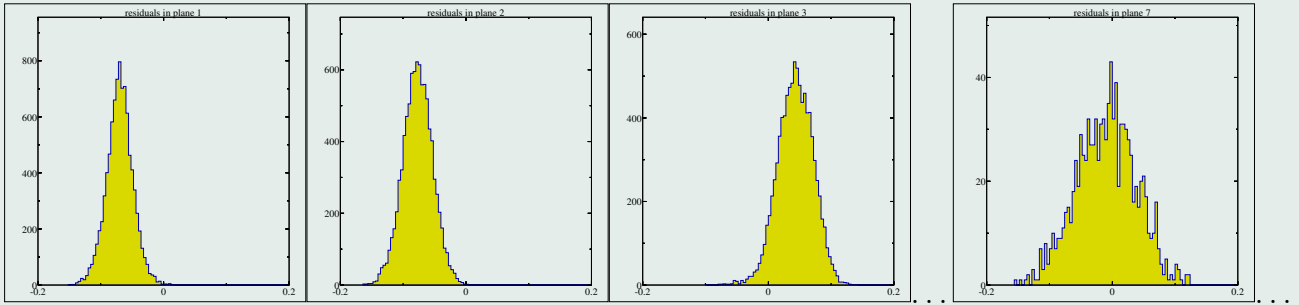
- 10 planes of tracking chambers, 1 m high, 10 cm distance, no magnetic field;
- accuracy $\sigma \approx 200\mu$m, with efficiency $\epsilon = 90\%$;
- plane 7 sick: accuracy $\sigma \approx 400\mu$m, with efficiency $\epsilon = 10\%$;
- 10 000 tracks with 82 000 hits available for alignment;
- Misalignment: the vertical position of the chambers are displaced by ≈ 0.1cm (normal distributed).



First attempt based on residuals

The first alignment attempt is based on the distribution of hit residuals:

- A straight line is fitted to the track data.
- The residuals (= measured vertical coordinate − fitted coordinate) are histogrammed, separately for each plane.



- The mean value of the residuals is taken as correction to the vertical plane position.

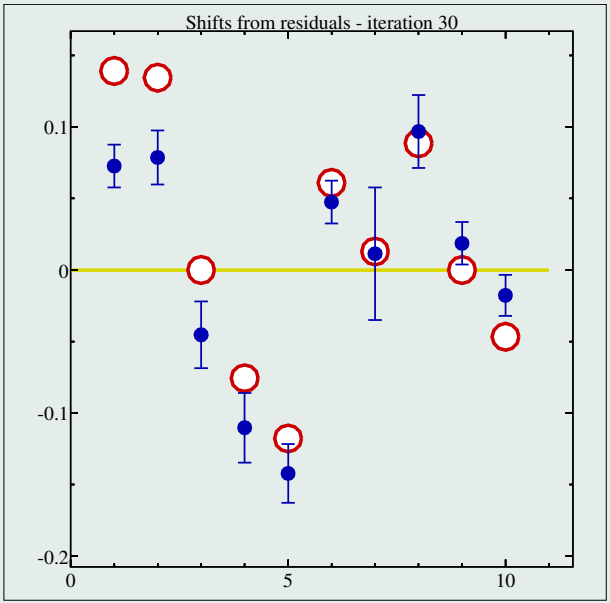
This is the standard method used in many experiments.

Result from the first attempt

Large changes in first iteration, small changes in second iteration, almost no progress afterwards.

After 30 iterations ...

ID	true shift	determined	mean residual
1	0.1391	0.0727	0 ± 150
2	0.1345	0.0786	0 ± 189
3	0.0000	−0.0453	0 ± 234
4	−0.0756	−0.1102	0 ± 244
5	−0.1177	−0.1422	0 ± 205
6	0.0610	0.0475	0 ± 150
7	0.0130	0.0114	0 ± 464
8	0.0886	0.0968	0 ± 255
9	0.0000	0.0186	0 ± 149
10	−0.0467	−0.0176	0 ± 143



red circle = true shift (displacement)

blue disc = displacement, determined from residuum

First attempt – Discussion

The result is not (yet) encouraging!

The reason for non-convergence is simple:

> Two degrees of freedom are undefined: a simultaneous shift and a overall shearing of the planes!

(. . . this simple fact is not always mentioned in reports on the method!)

Improvement for second residual attempt:

> Fix the displacement (i.e. displacement = 0) of two planes, which are assumed to be carefully aligned externally (e.g. planes 3 and 9).

Other possibilities are:

- Use only fixed planes (planes 3 and 9) in the fit, and determine the residuals of other planes;
- for the determination of the displacement of a certain plane use all other planes in the fit.

These possibilities are in fact used by several collaborations!

Results from the second attempt

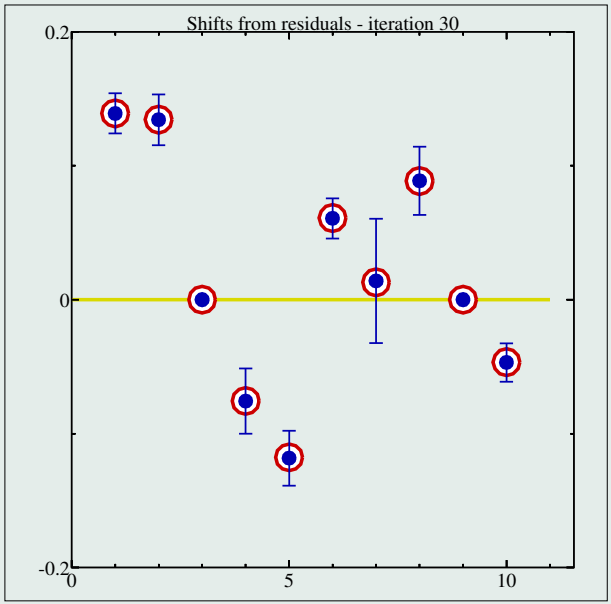
Large changes in first iteration, then many smaller and smaller changes: convergence is **linear** and slow, because the determination of displacements is based on biased fits.

After 30 iterations with planes 3 and 9 fixed (displacement = 0) ...

ID	true shift	determined	mean residual
1	0.1391	0.1391	-1 ± 150
2	0.1345	0.1344	0 ± 189
3	0.0000	0	2 ± 234
4	−0.0756	−0.0758	0 ± 244
5	−0.1177	−0.1183	0 ± 205
6	0.0610	0.0607	0 ± 150
7	0.0130	0.0140	0 ± 464
8	0.0886	0.0888	0 ± 255
9	0.0000	0	0 ± 149
10	−0.0467	−0.0469	0 ± 143

red circle = true shift (displacement)

blue disc = displacement, determined from residuum



Use of higher mathematics?

Residual-based methods work with biased results. Can the bias be avoided by an improved fit?

Yes: include the alignment parameters in the parameters fitted in track fits – requires a simultaneous fits of many tracks, with determination of (global) alignment parameters and (local) track parameters.

model: $y_i \cong a_1^{\text{local}} + a_2^{\text{local}} \cdot x_i \quad + a_j^{\text{global}}$ $\qquad a_j^{\text{global}} =$ shift for plane j, where y_i is measured

1 tracks	$2 + 10 = 12$ parameters	9 equations
2 tracks	$4 + 10 = 14$ parameters	18 equations
...	...	...
10 000 tracks	20 010 parameters	82 000 equations

... a linear least squares problem of $m = 82\,000$ equations (measurements) and $n = 20\,010$ parameters with $n \ll m$, which requires the solution of a matrix equation with 20010-by-20010 matrix.

... a nice problem, which has a fast solution.

Results from a simultaneous fit

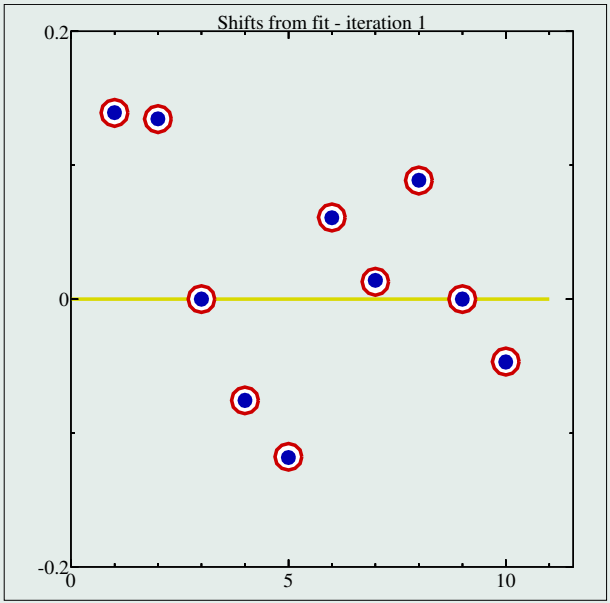
After one step (with planes 3 and 9 fixed at displacement = 0) ...

ID	true shift	determined	ρ	mean residual
1	0.1391	0.1393 ± 0.0004	0.68	0 ± 150
2	0.1345	0.1346 ± 0.0003	0.66	0 ± 189
3	0.0000			0 ± 234
4	−0.0756	-0.0756 ± 0.0003	0.58	0 ± 244
5	−0.1177	-0.1182 ± 0.0003	0.53	0 ± 205
6	0.0610	0.0608 ± 0.0003	0.50	0 ± 150
7	0.0130	0.0141 ± 0.0007	0.20	0 ± 464
8	0.0886	0.0888 ± 0.0003	0.53	0 ± 255
9	0.0000			0 ± 149
10	−0.0467	-0.0469 ± 0.0003	0.57	0 ± 143

(ρ = global correlation coefficient)

red circle = true shift (displacement)

blue disc = displacement, determined in fit



One step is sufficient: 1. step $\Delta\chi^2 = 1.277 \times 10^6$ 2. step $\Delta\chi^2 = 1.159 \times 10^{-5}$

But how can this problem be solved in a small fraction of a second?

Determination of drift velocities ... 10 additional parameters

Improvement: include, in addition, corrections to the drift velocities for each plane: $\Delta v_{\text{drift}}/v_{\text{drift}}$

$$y_i \cong a_1^{\text{local}} + a_2^{\text{local}} \cdot x_i \quad + a_j^{\text{global}} + \ell_{\text{drift},i} \cdot \left(\frac{\Delta v_{\text{drift}}}{v_{\text{drift}}}\right)_j$$

a_j^{global} = shift for plane j

$\left(\frac{\Delta v_{\text{drift}}}{v_{\text{drift}}}\right)_j$ = relative v_{drift} difference

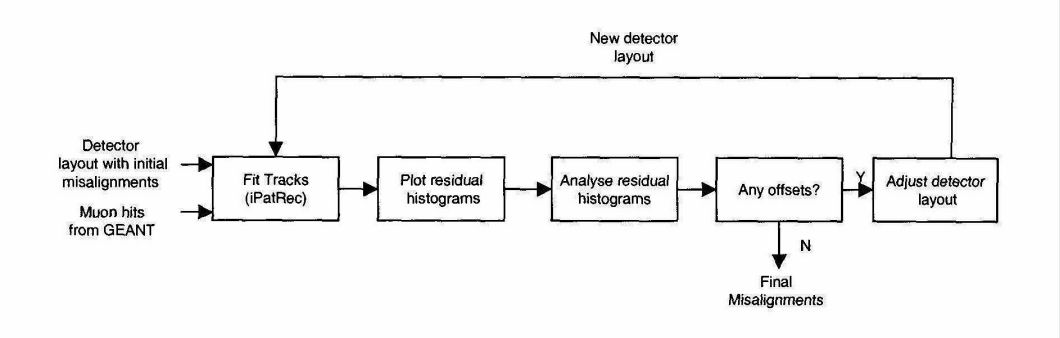
reduction of residual σ by 30 - 40 %

ID	true shift	determined	ρ	$\Delta v_{\text{drift}}/v_{\text{drift}}$	determined	ρ	mean residual
1	0.1391	0.1393 ± 0.0004	0.68	0.0020	0.0019 ± 0.0002	0.016	0 ± 119
2	0.1345	0.1346 ± 0.0003	0.66	-0.0153	-0.0150 ± 0.0002	0.020	0 ± 128
3	0.0000			0.0193	0.0194 ± 0.0002	0.017	0 ± 137
4	-0.0756	-0.0756 ± 0.0003	0.58	0.0200	0.0197 ± 0.0002	0.013	0 ± 139
5	-0.1177	-0.1182 ± 0.0003	0.53	-0.0138	-0.0136 ± 0.0002	0.013	0 ± 141
6	0.0610	0.0608 ± 0.0003	0.50	0.0003	0.0004 ± 0.0002	0.019	0 ± 139
7	0.0130	0.0141 ± 0.0007	0.20	-0.0306	-0.0303 ± 0.0006	0.038	0 ± 348
8	0.0886	0.0888 ± 0.0003	0.53	0.0237	0.0238 ± 0.0002	0.018	0 ± 134
9	0.0000			-0.0044	-0.0044 ± 0.0002	0.008	0 ± 127
10	-0.0467	-0.0469 ± 0.0003	0.57	0.0021	0.0019 ± 0.0002	0.013	0 ± 117

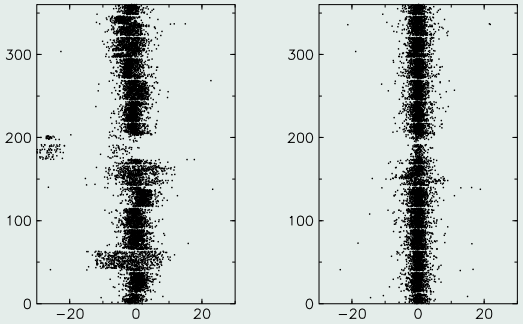
... this would be rather difficult with a pure residual-based method.

The next improvement would be the introduction of wire T_0's – additional $10 \times 25 = 250$ parameters.

Basic idea: make histograms of track residuals – parameter correction from peak (or mean or median) of histogram.



- Simple, possible from n-tuples;
- No precision alignment, because based on biased fits; constraints difficult; limited to parameter shifts;
- **But** useful to find large misalignments of certain detectors, if *unused hits* are included. Detection of large misalignments may be impossible otherwise.



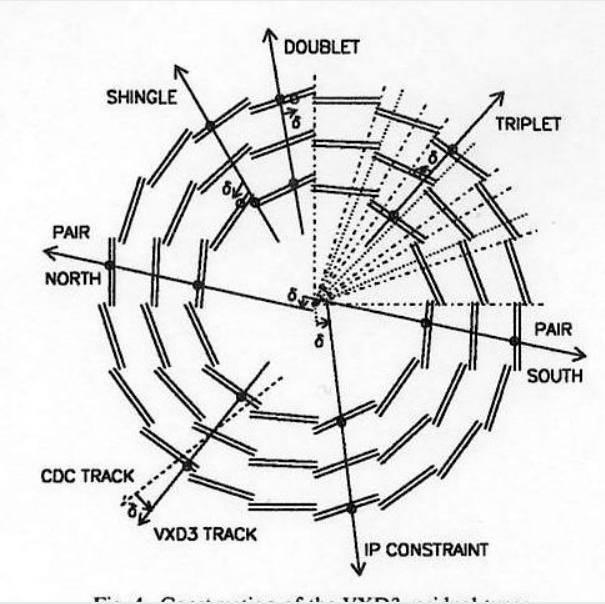
Avoid iterations . . . (there are undefined degrees of freedom).

3.2 Parametrization of residual dependence

Example: Internal alignment of the SLD vertex detector
D. J. Jackson, D. Su and F. J. Wickens, Internal alignment of the SLD vertex detector using a matrix singular value decomposition technique. Nuclear Instr. Methods **A 491** *(2002) 351-365*

Classification of types of tracking "constraints" →

Good quality tracks were constrained to pass through two of the CCD hits with the corresponding residual measued to the third, reference, CCD.



Residual types of the SLD pixel vertex detector

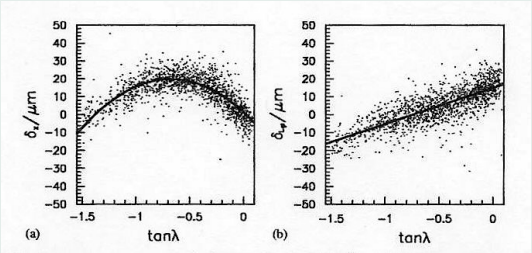
For each type of residuals, n-tuples were accumulated containing the deviation. Data in the n-tuples were fitted to the functional form given below to determine the coefficients and their covariance matrix (using MINUIT).

CCD shape corrections were taken into account from the optical survey data of the CCD surfaces (2108 → 5026 coefficients).

Table 1
Functional forms for fitting the various residual types. The superscripts ∥ and ⊥ indicate coefficients of fits to residuals measured parallel and perpendicular to the z-axis, respectively. N_I lists the total number of independent residual fits involved while N_C is the number of coefficients determined.

Type	Functional form	N_I	N_C
Shingles	$\delta_z = s_1^{\parallel} + s_2^{\parallel} \tan\lambda + s_3^{\parallel} \tan^2\lambda$	96	288
	$\delta_{L\phi} = s_1^{\perp} + s_2^{\perp} \tan\lambda$	96	192
Doublets	$\delta_z = d_1^{\parallel} + d_2^{\parallel} L_\phi$	48	96
	$\delta_{L\phi} = d_1^{\perp} + d_2^{\perp} L_\phi + d_3^{\perp} L_\phi^2$	48	144
Triplets	$\delta_z = t_1^{\parallel} + t_2^{\parallel} \tan\lambda + t_3^{\parallel} \tan^2\lambda + t_4^{\parallel} L_\phi \tan\lambda + t_5^{\parallel} L_\phi$	80	400
	$\delta_{L\phi} = t_1^{\perp} + t_2^{\perp} L_\phi + t_3^{\perp} L_\phi^2 + t_4^{\perp} L_\phi \tan\lambda + t_5^{\perp} \tan\lambda$	80	400
Pairs	$\delta_{rz} = p_1^{\parallel} + p_2^{\parallel} \tan\lambda + p_3^{\parallel} \tan^2\lambda$	28	84
	$\delta_{r\phi} = p_1^{\perp} + p_2^{\perp} \tan\lambda$	28	56
	$\delta_\phi = p_1^{\phi} + p_2^{\phi} \tan\lambda$	28	56
CDC angle	$\delta_\lambda = c_1^{\lambda} + c_2^{\lambda} \tan\lambda + c_3^{\lambda} \tan^2\lambda$	56	168
	$\delta_\phi = c_1^{\phi} + c_2^{\phi} \tan\lambda$	56	112
IP constraint	$\delta_{r\phi} = i_1^{\perp} + i_2^{\perp} \tan\lambda$	56	112
	Total	700	2108

Functional forms for fitting the various residual types.



Examples for residual fits, here as a function of $\tan\lambda$ in a layer before the alignment.

In total there are 2108 coefficients from 700 residual fits (exercise in book-keeping).

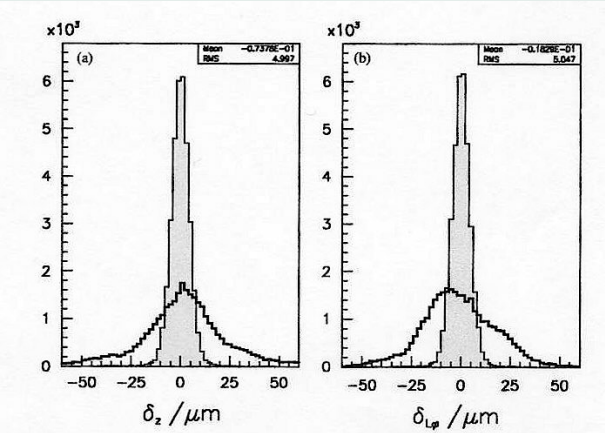
Taking into account the covariance matrices of the residual fits, 866 alignment corrections were determined from 5026 coefficients from the residual fits, using SVD techniques for the least-squares fit minimization.

> " . . . Although the size of matrices involved appears daunting only $\approx 1\%$ or $\approx 35,000$ elements of the final 5026×866 design matrix $\boldsymbol{A}$ were given non-zero values . . . "

Note: nr of coefficients (5026) is determined by parametrization types, not by nr of events.

Only a single iteration is required.

With the aligned geometry the design performance is achieved →



Triplet residuals obtained with the original survey geometry and after alignmnent.

Similar method: DELPHI Barrel Silicon Tracker alignment, P. Brückman de Renstrom: The Final Alignment of the Barrel Silicon Tracker at LEP2, DELPHI 2004-047 TRACK 098.

3.3 Alignment using Kalman filter

CMS: About 20 000 silicon sensors, with resolution 10 μm to 40 μm. Expected precision from mechanical mounting and LASER beam alignment worse than intrincic resolution – alignment using tracks is necessary.

Extension of the Kalman filter method of track fitting, with updating the current alignment parameters after each track.

R. Frühwirth, T. Todorov and M. Winkler, Estimation of Alignment Parameters, using the Kalman Filter with annealing, CMS Note 2002-008

R. Frühwirth, T. Todorov and M. Winkler, Estimation of Alignment Parameters using the Kalman Filter with annealing, Journal of Physics G: Nuclear and Particle Physics **29** *(2003) 561–574*

Update formulae for global parameters $\boldsymbol{a}$ and their covariance matrix $\boldsymbol{E}$:

$$\boldsymbol{a}_1 := \boldsymbol{a}_0 + \boldsymbol{E}_0\boldsymbol{D}^{\mathrm{T}}\boldsymbol{W}\left[\boldsymbol{m} - \boldsymbol{f}(\boldsymbol{p}_0, \boldsymbol{a}_0\right] \qquad \text{with} \quad \boldsymbol{W} = \left[\boldsymbol{V} + \boldsymbol{H}\boldsymbol{C}_0\boldsymbol{H}^{\mathrm{T}} + \boldsymbol{D}\boldsymbol{E}_0\boldsymbol{D}^{\mathrm{T}}\right]^{-1}$$
$$\boldsymbol{E}_1 := \boldsymbol{E}_0 - \boldsymbol{E}_0\boldsymbol{D}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{D}\boldsymbol{E}_0$$

. . . requires some matrix operations.

Convergence to local minima not excluded: introduction of "annealing" = gradually stepping up the weights of the observations in the course of the estimation process.

4. Matrix methods for alignment

Alignment of large detectors requires determination of large number n of corrections $\boldsymbol{\Delta p}$ to the alignment parameters $\boldsymbol{p}$

100 (Pluto) ... 1 000 (H1) ... 5 000 (Hera B) ... 35 000 (Atlas) ... 100 000 (CMS)

Optimization: The standard method for the determination of a large number of *correlated* parameters results is the solution of a large system of linear equations ($\boldsymbol{\Delta p}$ = corrections to alignment parameters)

$$\boxed{\boldsymbol{C}\ \boldsymbol{\Delta p} = \boldsymbol{b}}$$

which perhaps has to be solved iteratively because of nonlinearities. In the least squares method the matrix $\boldsymbol{C}$ is a symmetric $n \times n$ matrix, with

$$n = 100 \quad \ldots \quad 1\,000 \quad \ldots \quad 5\,000 \quad \ldots \quad 35\,000 \quad \ldots \quad 100\,000$$

Two problems to be solved:

- construction of correct matrix $\boldsymbol{C}$ and vector $\boldsymbol{b}$, or at least of a good approximation to the matrix $\boldsymbol{C}$ and vector $\boldsymbol{b}$, and ...
- correct solution of the large matrix equation, or at least a good approximation to the correct solution.

Global alignment fit

Determination of corrections $\boldsymbol{\Delta p}$ for alignment parameters $\boldsymbol{p}$ is based on minimization of residuals – the difference between fitted and measured track position:

$$\Delta_i = \text{ fitted value } - \text{ measured value}$$

A "global" χ^2-function is constructed, which either depends only on the alignment corrections $\boldsymbol{\Delta p}$

Method 1:
$$\chi^2\left(\boldsymbol{\Delta p}\right) = \sum_{\text{data sets}} \left(\sum_{\text{events}} \left(\sum_{\text{tracks}} \left(\sum_{\text{hits}} \Delta_i^2/\sigma_i^2 \right) \right) \right)$$

or which depends on the corrections $\boldsymbol{\Delta p}$ and **all** track parameters $\boldsymbol{q}_j$

Method 2:
$$\chi^2\left(\boldsymbol{\Delta p}, \boldsymbol{q}_j\right) = \sum_{\text{data sets}} \left(\sum_{\text{events}} \left(\sum_{\text{tracks}} \left(\sum_{\text{hits}} \Delta_i^2/\sigma_i^2 \right) \right) \right)$$

Data sets are

- Physics data: e^+e^--, $e^-\,p$-, $p\,p$-reactions,
- Cosmics with magnetic field (large distance to IP) and without magnetic field (straight tracks, curvature zero),
- exclude low momentum tracks (multiple scattering).

A mixture of different data is recommended, in order to introduce different correlations between the alignment parameters.

Normal equations of least squares

Minimization requires first derivatives of hit residuals w.r.t. the parameters – *linearization*. Residuals and derivatives are used to determine $n \times n$-matrix $\boldsymbol{C}$ and n-vector of the so-called normal equations of least squares.

Method 1: Fitted parameters are only the alignment corrections $\boldsymbol{\Delta p}$, while the fits of tracks are made before - *ignoring the alignment parameters*. But the fitted track parameters depend on the alignment and the method ignores (within one alignment fit) the correlation between track and alignment parameters.

Good solution requires *iterations*, with alternating track fits and alignment fits. Rate of convergence is unknown (linear and slow?) and *many iterations* may be neccessary – hopefully with convergence.

Method 1: $\boldsymbol{C}\ \boldsymbol{\Delta p} = \boldsymbol{b}$ $\qquad \boldsymbol{C} =$ symmetric $n \times n$ matrix

or

Method 2:

$$\left(\begin{array}{c|ccc} \boldsymbol{C} & \dots & \boldsymbol{H}_k & \dots \\ \hline \dots & \dots & \dots & \dots \\ \boldsymbol{H}_k^{\mathrm{T}} & \mathbf{0} & \boldsymbol{\Gamma}_k^{\mathrm{track}} & \mathbf{0} \\ \dots & \dots & \dots & \dots \end{array}\right) \times \left(\begin{array}{c} \boldsymbol{\Delta p} \\ \dots \\ \boldsymbol{q}_j \\ \dots \end{array}\right) = \left(\begin{array}{c} \boldsymbol{b} \\ \dots \\ \boldsymbol{b}_k^{\mathrm{track}} \\ \dots \end{array}\right) \qquad k = \text{track index}$$

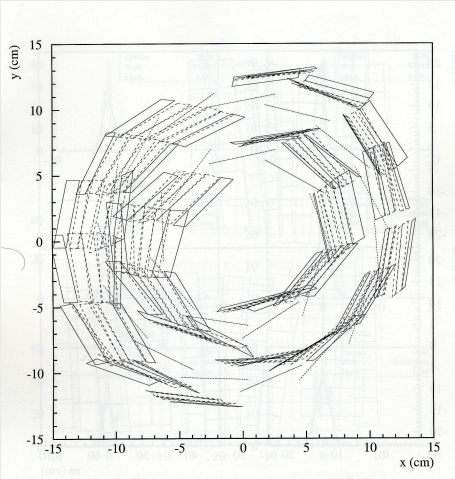
Alignment of the upgraded VDET at LEP2 (ALEPH) Method 1 example

A. Bonissent et al., Alignment of the upgraded VDET at LEP2, ALEPH 97-116

A "global" χ^2 involving all $864 = 144 \times 6$ degrees of freedom is built, using single tracks and vertex constraints. Selected information is used from the outer tracking. 16 000 hadronic Z events used.

Precise faces measurement are used to reduce the degrees of freedom ($\rightarrow$ 378 "directions"), while allowing for parametrized distortions.

"The method is shown to provide accurate results, even with a limited number of events."



Local alignment of the VDET in the $r\phi$ view. The dashed lines show the nominal position. The end wafers are drawn with solid lines, the other with dashed lines. Displacements are amplified by a factor of 100.

Parameters of a Vertex detector

Planar sensors (silicon pixel or strip detectors): local (sensor) coordinates $\boldsymbol{q} = (u, v, w)$ and global detector coordinates $\boldsymbol{r} = (x, y, z)$ are transformed by

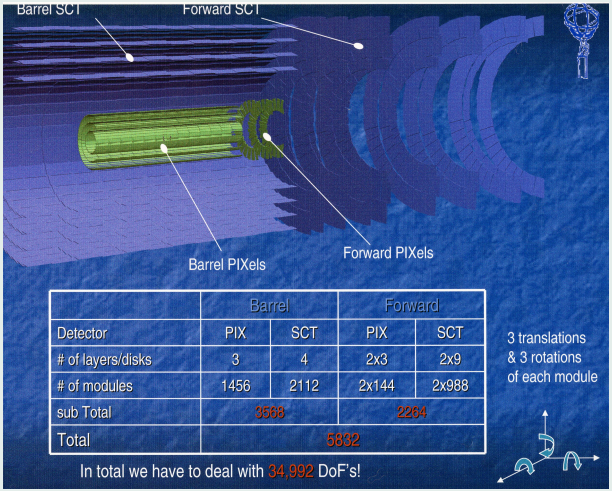
$$\boldsymbol{q} = \boldsymbol{R}\,(\boldsymbol{r} - \boldsymbol{r}_0) \qquad \boldsymbol{R} = \text{ nominal rotation}, \quad \boldsymbol{r}_0 = \text{ nominal position}$$

After alignment the transformation becomes (with $\Delta\boldsymbol{q} = (\Delta u, \Delta v, \Delta w)$)

$$\boldsymbol{q}^c = (\boldsymbol{R}_\alpha \boldsymbol{R}_\beta \boldsymbol{R}_\gamma)\ \boldsymbol{R}\,(\boldsymbol{r} - \boldsymbol{r}_0) - \Delta\boldsymbol{q}$$

Six parameters are required for each individual detector element, out of which **three** parameters (two translations, one rotation) are very sensitive.

Detector	nr of elements	nr of parameters
SLD	96	$96 \times 6 = 576$
Aleph	144	$144 \times 6 = 864$
Delphi	24	72
Zeus	30	$30 \times 6 = 180$
CDF	352	$352 \times 3 = 1056$
Atlas	5832	$5832 \times 6 = 34992$
CMS	≈ 20000	≈ 100000



Reduction of large matrix C ... to many 6×6 matrices (Method 1)

V. Karimäki, A. Heikkinen, T. Lampén and T. Lindén, Sensor alignment by tracks, CMS Conference Report, CHEP03, La Jolla, California, March 24-28, 2003, and talk by T. Lampén.

Repeated track fitting, ignoring dependence on alignment parameters, with χ^2-minimization of many "independent" sets of 6 parameters (three location and 3 orientation) and inversion of many 6×6 matrices (no correlation between 6-parameter sets).

Test beam setup:	layer 1	144 sensors at 4 cm
	layer 2	240 sensors at 8 cm

Monte Carlo simulation:
Fit I: all sensors in layer 2 fixed, all sensors in layer 1 misaligned, requires 8 iterations for 864 parameters.
Fit II: only one sensor in layer 2 fixed, all remaining 383 sensors misaligned, requires ≈ 100 iterations for 2 298 parameters.

Similar method developed in Atlas.

Similar method: T. Kohno, ZEUS (ACAT05) for $30 \times 6 = 180$ parameters (with refit of each track with all hits except one); requires 15 iterations.

Method 2: Fitted parameters are the alignment corrections $\boldsymbol{\Delta p}$ and all track parameters $\boldsymbol{q}_k$ from the data used in the fit.

Huge matrix (size depends on number of tracks), but

- linear algebra allows reduction to $n \times n$ matrix $\boldsymbol{C}$,
- solved in a single step (no iterations),
- taking all correlations into account.
- and allowing to introduce equality constraints.

No iteration or small number of iterations:

- check of code (derivatives),
- linearization or linear approximation,
- accuracy of solution of large system of linear equations,
- cuts to remove outliers, reduced from iteration to iteration.

Simultaneous least squares fit of all global and all local parameters (i.e. all tracks).

$$k'\text{th track:} \qquad y_i \cong f(x_i; \boldsymbol{p}^{\text{global}}, \boldsymbol{q}^{\text{local}}) + \left(\boldsymbol{d}_i^{\text{global}}\right)^{\text{T}} \Delta\boldsymbol{p}^{\text{global}} + \left(\boldsymbol{\delta}_i^{\text{local}}\right)^{\text{T}} \Delta\boldsymbol{q}_k^{\text{local}}$$

The complete matrix equation for global and local parameters includes sums over track index k and contains many matrices: n-by-n matrices $\boldsymbol{C}$ for n global parameters and m-by-m matrices $\boldsymbol{C}_k^{\text{local}}$ and n-by-m matrices $\boldsymbol{H}_k^{\text{global-local}}$

$$\begin{pmatrix} \sum_k \boldsymbol{C}_k^{\text{global}} & \cdots & \boldsymbol{H}_k^{\text{global-local}} & \cdots \\ \hline \vdots & \ddots & 0 & 0 \\ \hline \left(\boldsymbol{H}_k^{\text{global-local}}\right)^{\text{T}} & 0 & \boldsymbol{C}_k^{\text{local}} & 0 \\ \hline \vdots & 0 & 0 & \ddots \end{pmatrix} \times \begin{pmatrix} \Delta\boldsymbol{p}^{\text{global}} \\ \hline \vdots \\ \hline \Delta\boldsymbol{q}_k^{\text{local}} \\ \hline \vdots \end{pmatrix} = \begin{pmatrix} \sum_k \boldsymbol{b}_k^{\text{global}} \\ \hline \vdots \\ \hline \boldsymbol{b}_k^{\text{local}} \\ \hline \vdots \end{pmatrix}$$

If the $\boldsymbol{H}_k^{\text{global-local}}$ are neglected, the complete equation decays into $1 + K$ independent matrix equations.

But the solution $\Delta\boldsymbol{p}^{\text{global}}$ can be calculated without approximation with a great simplification: $\Rightarrow$

For each track in a loop, on all tracks:

1. **Track- or other fit:** perform fit by finding the best local parameter values for the actual track until convergence with determination of the covariance matrix $\boldsymbol{V}_k$ of the local parameters

2. **Derivatives:** calculate for all hits (index i) the vectors of derivatives $\boldsymbol{\delta}_i^{\text{local}}$ and $\boldsymbol{d}_i^{\text{global}}$ for all local and the relevant global parameters, and update matrices:

$$\boldsymbol{C} := \boldsymbol{C} + \sum_i w_i \boldsymbol{d}_i^{\text{global}} \left(\boldsymbol{d}_i^{\text{global}}\right)^{\text{T}} \qquad \boldsymbol{b} := \boldsymbol{b} + \sum_i w_i r_i \boldsymbol{d}_i^{\text{global}} \qquad \boldsymbol{H}_k = \sum_i w_i \boldsymbol{d}_i^{\text{global}} \left(\boldsymbol{\delta}_i^{\text{local}}\right)^{\text{T}}$$

and finally for the track $\boldsymbol{C} := \boldsymbol{C} - \boldsymbol{H}_k \boldsymbol{V}_k \boldsymbol{H}_k^T$

The two 'blue' equations transfer the 'local' information to the global parameters.

After the loop on all tracks the complete information is collected; now the matrix equation for the global parameters has to be solved:

$$\Delta \boldsymbol{p}^{\text{global}} = \boldsymbol{C}^{-1}\, \boldsymbol{b}$$

Note: matrices $\boldsymbol{C}$ and vectors $\boldsymbol{b}$ from several data sets can be simply added to get combined result.

Used (or under test) by H1(1997), CDF(2001), Hera-B, ZEUS, CMS, Atlas . . .

V. Blobel: Experience with Online Calibration Methods, Contribution to CHEP'97, Berlin 1997 (including the **Millepede** method), not accepted.

V. Blobel, Linear Least Squares Fits with a Large Number of Parameters, (2000),
`http://www.desy.de/~blobel` including Fortran code.
V. Blobel and C. Kleinwort: A New Method for the High-Precision Alignment of Track Detectors, PHYSTAT2002, Durham, `arXiv-hep-ex/0208021`

Atlas ID Alignment (group convenor A. Hicheur) [see also: P. Brückman de Renstrom, S. Haywood: Least squares approach to the alignment of the generic tracking system, PHYSTAT 2005, Oxford.] Reduction of matrix size with different derivation; contains several formulas for vertex constraints etc.

The algorithm is general and can be applied to other problems with large number of (global) parameters and a huge number of measurements with local parameters.

Principle of reducing matrix size (perhaps) used earlier:
Schreiber, O. (1877): Rechnungsvorschriften für die trigonometrische Abteilung der Landesaufnahme, Ausgleichung und Berechnung der Triangulation zweiter Ordnung. Handwritten notes. Mentioned in W. Jordan (1910): Handbuch der Vermessungskunde, Sechste erw. Auflage, Band I, Paragraph III: 429-433. J.B.Metzler, Stuttgart.

Undefined degrees of freedom . . . or weakly defined degrees of freedom

Alignment of HEP track detectors . . . based **only** on track residual minimization: *incomplete* data, with several degrees of freedom undefined! Certain parameters are **undefined** or only weakly defined and could distort the detector.

General **linear** transformation with translation and 3×3 matrix $\boldsymbol{R}$

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} d_x \\ d_y \\ d_z \end{pmatrix} + \boldsymbol{R} \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

defined by $3 + 9$ parameters, will not affect the χ^2 of the fits, The matrix (9 parameters) can be decomposed into

- three rescaling factors of coordinate axes: f_x, f_y, f_y,
- three rotations: $\mathcal{D}_x$, $\mathcal{D}_y$, $\mathcal{D}_z$ and three shearings: $\mathcal{T}_{xz}$, $\mathcal{T}_{yz}$, $\mathcal{T}_{xy}$.

In addition there may be weakly defined **nonlinear** transformations.

These parameters can be fixed by external measurements → hardware alignment devices, i.e. alignment by tracks has to be supplemented by external information, or fixed by equality constraints or by orthogonalization methods.

Equality constraints

Undefined degrees of freedom can be avoided by adding equality constraint equations of the type

$$g(\boldsymbol{p}) = 0 \qquad \text{e.g.} \quad d_x = \sum_i \Delta x_i = 0$$

e.g. "zero average displacement", or "zero rotation of the whole detector".

There are two possibilities:

- fix certain parameters, or fix linear combinations of parameters (by SVD)
- add equality constraint equation, i.e.

append linearized Lagrange multiplier equation $\lambda \left(g(\boldsymbol{p}) + \boldsymbol{g}^{\mathrm{T}} \cdot \Delta\boldsymbol{p} = 0 \right)$ with $\boldsymbol{g} = \partial g(\boldsymbol{p}) / \partial \boldsymbol{p}$:

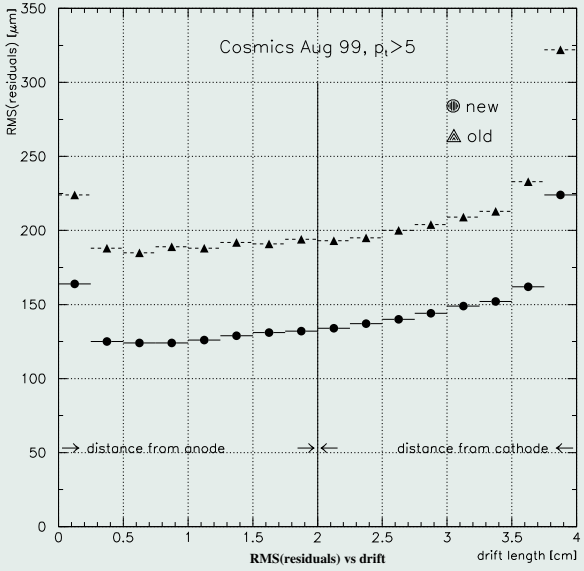
$$\left(\begin{array}{c|c} \boldsymbol{C}^{\text{global}} & \boldsymbol{g} \\ \hline \boldsymbol{g}^{\mathrm{T}} & \boldsymbol{0} \end{array}\right) \left(\begin{array}{c} \Delta\boldsymbol{p}^{\text{global}} \\ \hline \lambda \end{array}\right) = \left(\begin{array}{c} \boldsymbol{b}^{\text{global}} \\ \hline -g(\boldsymbol{p}) \end{array}\right)$$

Driftchamber alignment

Claus Kleinwort et al., Detailed calibration of H1 drift chamber using MILLEPEDE with about 1400 parameters.

1400 parameters using 50 000 tracks:

- *common* alignment of the drift chamber and the silicon detector;
- for both CJC1 and CJC2 14 global parameters representing an overall shift or tilt are introduced;
- local variations of the drift velocity v_{drift} for cells halfs and layers halfs are observed, which are parametrized by 180 + 112 corrections, which change with the HV configuration;
- for each wire group (8 wires) corrections to T_0 are introduced (330 corrections).



Reduction of residual RMS by MILLEPEDE.

5. Numerical linear algebra

Determination of *good* solution for $\boldsymbol{\Delta p}$ in

$$\boxed{\boldsymbol{C\,\Delta p = b}} \qquad \boldsymbol{C^{-1} = V} = \text{ covariance matrix for parameters}$$

with large or huge $n \times n$ matrix depends on **algorithm** and **data**, and requires

0. Scaling: Use consistent units in data and variables – matrix and vector should be well scaled, i.e. elements should have similar precision.

1. Algorithm – Stability: With a stable algorithm the computed solution is the exact solution of a nearby problem. Gaussian Elimination with diagonal pivoting (restricted scan for largest next element) is considered to be a stable algorithm for positive definite matrices.

2. Data – Conditioning: Data are called ill-conditioned, if small changes in the data can cause large changes in the solution ($\rightarrow$ small eigenvalue(s)). Matrix is ill-conditioned if variables are undefined or poorly defined or strongly correlated.

Whole memory filled by symmetrix matrix (if huge); compact storage for *sparse* matrix. Most algorithms could work *in-space*.

Note: printing the elements of a n-by-n matrix for 100 000 requires ≈ 10 m^3 paper (double-sided printing).

Standard is stable Gauss algorithm with pivot selection of diagonal, with

$$\text{Computing time} \;=\; \text{constant} \times n^3 \qquad < 1 \text{ hour for } n = \text{ several thousand}$$

(in practice "constant" increases for large n).

A standard matrix routine will fail – at least a few parameters out of many thousands will be badly defined.

Subroutine SPMINV (in Millepede) for symmetric matrices in $(n^2+n)/2$ words: choose largest pivot, but stop inversion if no acceptable pivot found, i.e. invert largest possible submatrix; return zero corrections for remaining parameters.

All variances and covariances available in inverse matrix.

The **global correlation coefficient**, ρ_j is a measure of the total amount of correlation between the j-th parameter and *all* the other variables. It is the largest correlation between the j-th parameter and every possible linear combination of all the other variables.

$$\rho_j = \sqrt{1 - \frac{1}{(\boldsymbol{V})_{jj} \cdot (\boldsymbol{C})_{jj}}} \qquad \text{and} \qquad (\boldsymbol{V})_{jj} \cdot (\boldsymbol{C})_{jj} = \frac{1}{1-\rho_j^2}$$

Algorithms allow to recognize singularity or near singularity by the determination of singular or eigenvalues, and to ignore corresponding linear combinations of parameters.
Algorithms are iterative, computing time ≈ 10 times larger, and solution less precise.

$$\begin{aligned} (\ \boldsymbol{C} &= \boldsymbol{V}\ \boldsymbol{D}\ \boldsymbol{U}^{\mathrm{T}} && \text{Singular value decomposition (SVD) for } m \times n \text{ matrix } \ ) \\ \boldsymbol{C} &= \boldsymbol{U}\ \boldsymbol{D}\ \boldsymbol{U}^{\mathrm{T}} && \text{Diagonalization of symmetric matrix} \end{aligned}$$

with $\boldsymbol{D}$ diagonal, $\boldsymbol{U}$ square and orthogonal with $\boldsymbol{U}\,\boldsymbol{U}^{\mathrm{T}} = \boldsymbol{U}^{\mathrm{T}}\,\boldsymbol{U} = \mathbf{1}$. Note: $\boldsymbol{C}^{-1} = \boldsymbol{U}\ \boldsymbol{D}^{-1}\ \boldsymbol{U}^{\mathrm{T}}$

ordering in $\boldsymbol{D} = [\mathrm{diag}\,(\lambda_i)]$: $\quad \lambda_1 \ \geq\ \lambda_2 \ \geq\ \ldots \geq \lambda_k \geq \lambda_{k+1} = \ldots \lambda_n = 0$ (or very small)

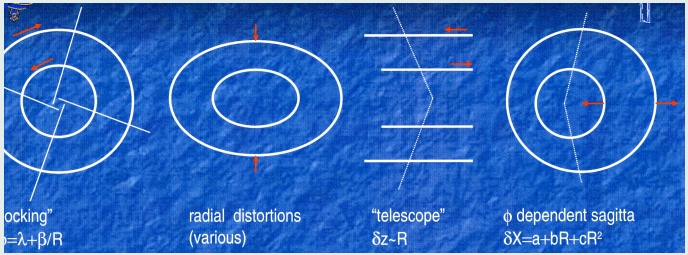
Solution of $\boxed{\boldsymbol{C}\,\boldsymbol{\Delta p} = \boldsymbol{b}}$ by $\quad \boldsymbol{\Delta p} = \boldsymbol{U}\ \left[\mathrm{diag}\left(\frac{1}{\sqrt{\lambda_i}}\right)\right] \underbrace{\left[\mathrm{diag}\left(\frac{1}{\sqrt{\lambda_i}}\right)\right]\ (\boldsymbol{U}^{\mathrm{T}}\boldsymbol{b})}_{= \boldsymbol{q} \quad \text{with} \quad \boldsymbol{V}[\boldsymbol{q}] = \mathbf{1}}$

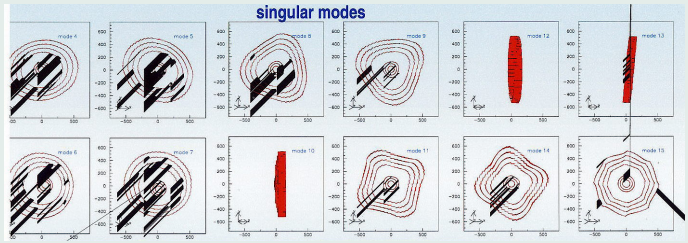
with $1/\lambda_i = 0$ for $\lambda_i = 0$ or small q_i with $|q_i| \lesssim 1$

$\Rightarrow$ Suppression of insignificant linear combinations, which could produce distortions of the detector.

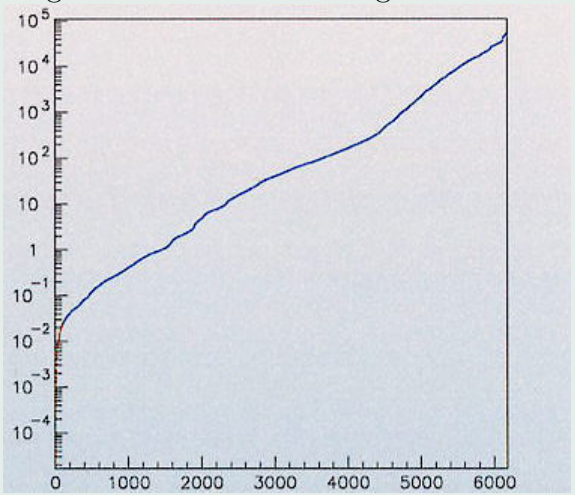
From PHYSTAT05 contribution by P. Brückman de Renstrom:
Singular values determined by SVD (no equality constraints applied to fix trivial degrees of freedom), and used to suppress potential distortions and singular modes.

(Distortions could be suppressed by the use of cosmic data with and without magnetic field.)





Singular values in increasing order



(several tiny singular values)

Singular modes removed by suppression of small singular values.

Algorithm III — Generalized minimal residual method (GMRES)

Solution of a very large system of linear equations with sparse matrix, by an clever solution of a quadratic minimization problem, in analogy to the method of conjugate gradients (Hestenes, Stiefel, Zürich 1952; E. Stiefel founded the Institute for Applied Mathematics at ETH).

Example: MINRES (M. A. Saunders), designed to solve

system of linear equations $\boxed{\boldsymbol{C}\,\boldsymbol{\Delta p} = \boldsymbol{b} \qquad \text{or} \qquad \min ||\boldsymbol{C}\,\boldsymbol{\Delta p} - \boldsymbol{b}||_2}$

where $\boldsymbol{C}$ is a symmetric matrix of logical size $n \times n$, which may be indefinite and/or singular, very large and sparse. It is accessed *only* by means of a subroutine call

```
call Aprod ( n, x, y )
```

which must return the product $\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x}$ for any given vector $\boldsymbol{x}$.

Example of **compact storage**, optimized for product (6 lines of code): row-index sparse storage *) with two arrays of $(n + qn(n-1)/2 + 2)$ words (q = fraction of non-zero off-diagonal elements) with double precision array for matrix elements and integer array for indices.

For $n =$ 100 000 and 512 Mbytes memory: $q = 1\%$
4 Gbytes memory: $q = 7\%$

C. C. Paige and M. A. Saunders (1975), Solution of sparse indefinite systems of linear equations, SIAM J. Numer. Anal. 12(4), pp. 617-629.

`www.stanford.edu/group/SOL/software/minres.html`

*) W. H. Press, S. A. Teukolsky, W. T. Vetterling, B.P. Flannery, Numerical Recipes – The Art of Scientific Computing, Cambridge Univ.

Decomposition of the symmetric matrix $\boldsymbol{C}$

$$\boldsymbol{C} = \boldsymbol{L}\boldsymbol{D}\boldsymbol{L}^{\mathrm{T}} \qquad \text{decomposition}$$

is "numerically extremely stable", and can be made *in-space*. Matrix $\boldsymbol{L}$ is a left unit triangular matrix (diagonal elements =1) and $\boldsymbol{D}$ is a diagonal matrix (less stable for semi-definite matrix).

Solution of $\boldsymbol{C}\boldsymbol{\Delta p} = \boldsymbol{L}\left(\boldsymbol{D}\boldsymbol{L}^{\mathrm{T}}\boldsymbol{\Delta p}\right) = \boldsymbol{b}$ by

$$\boldsymbol{L}\boldsymbol{v} = \boldsymbol{b} \quad \text{forward substitution} \qquad \boldsymbol{L}^{\mathrm{T}}\boldsymbol{\Delta p} = \boldsymbol{D}^{-1}\boldsymbol{v} \quad \text{forward substitution}$$

With clever ordering of parameters the matrix C can be approximated by a band matrix.

Band matrices with band-width m: the band structure is kept in this decomposition and computing time for solution is only $\propto n \times m^2$. Selected elements of the inverse matrix (corresponding to the band of the original matrix $\boldsymbol{C}$) can be calculated quickly.

Fast methods exist also for
Variable-bandwidth matrices ("sky-line" matrix), and for
Bordered band matrices ("arrow" matrix), where e.g. border (additional full rows and columns) is due to Lagrange multiplier constraints.

Decomposition is similar to the original Cholesky factorization; the introduction of the diagonal matrix $\boldsymbol{D}$ avoids the square-root operations.

Matrix algebra libraries

Numerical Algorithms Group (NAG): now includes 1,450+ routines

BLAS (Basic Linear Algebra Subprograms) from US National Science Foundation: standard building blocks for performing basic vector and matrix operations. Freely-available software package .... Commonly used in the development of high quality linear algebra software, LAPACK for example. Highly efficient machine-specific implementations.

LAPACK Linear Algebra PACKage from US National Science Foundation: freely available, written in Fortran77. Provides routines for solving systems of simultaneous linear equations, least-squares solutions of linear systems of equations, eigenvalue problems, and singular value problems. Dense and banded matrices are handled, but not general sparse matrices.

LINPACK, EISPACK: largely superceded by LAPACK

others: e.g. Intel Math Kernel Library (MKL), includes LAPACK. Highly optimized, thread-safe math routines for High-Performance Computing (HPC) science.

others: e.g. MINRES (mentioned before), SYMMLQ.

... mostly Fortran77

Atlas plans/ideas: system of linear equations with 30000×30000 matrix; with hardware of 16 processors (64 bits achitecture) with 900 Mb memory for each processor (ScaLAPACK).

6. Alignment strategies and summary Atlas and CMS

LHC detectors have huge amount of independent sensors with excellent resolution, from about 10 μm to about 50 μm, with n = several ten thousand parameters!

Atlas Silicon inner detector: Atlas ID Alignment (group convenor A. Hicheur)

Local χ^2 alignment: modules aligned on an individual basis with N 6×6-matrices iteratively - align $\rightarrow$ refit tracks $\rightarrow$ align $\rightarrow$ ...

Global χ^2 alignment: simultaneous alignment and track fits; full Pixel + SCT Barrel and Endcaps: $6N = 35\,000$. PHYSTAT05 contribution by P. Brückman de Renstrom and S. Haywood.

Alignment with overlaps: relative module to module misalignment determined from overlap residuals - robust, straightforward, but iterations and degrees of freedom

CMS: Aligment Coordinator O. Buchmüller

Sensor alignment by tracks: iterative procedure that considers individual measurement devices with 6×6-matrices (not taking into account correlations between measurement devices); CHEP03 contribution CMS-2003/022 and at TIME05.

Millepede II: upgrade of Millepede for $n \gg 10\,000$, with dynamic definition of sparse matrix storage, and three (four) different solution algorithms (V.B.)

Kalman filter alignment: developed by Rudolf Frühwirth and Vienna group.

Summary continued

Alignment problems with n = several thousand parameters have been successfully solved, either

- using special methods, or
- by algorithms with many iterations, or
- by single-step "global χ^2" algorithm.

Some recommendations concerning alignment:

- study the impact of the mis-alignment;
- check time stability;
- integration of alignment into reconstruction code;
- simultaneous use of all relevant detectors, not too many parameters ($6 \rightarrow 3$ per sensor?);
- simultaneous use of several (all) types of events and data – physics data: single tracks, vertices, invariant-mass constraints, overlaps and cosmics with and without field;
- find general strategy to suppress unwanted distortions, even for huge value of n.

Best of all strategies is unknown at present – develop in parallel several strategies and decide later ...

Contents