

# Track fitting in non-uniform magnetic field with segment-wise helical track model

Bo Li<sup>1</sup>   Keisuke Fujii<sup>2</sup>   Yuanning Gao<sup>1</sup>

<sup>1</sup>Center for High Energy Physics  
Tsinghua Univ.

<sup>2</sup>Institute of Particle and Nuclear Studies  
High Energy Accelerator Research Organization (KEK)

International Workshop on Future Linear Colliders 2013

*This talk is based on our paper at [arXiv:1305.7300 \[physics.ins-det\]](https://arxiv.org/abs/1305.7300) and *Computer Physics Communications* (2013), <http://dx.doi.org/10.1016/j.cpc.2013.11.003>.*

# Outline

## 1 Motivation

## 2 Track fitting algorithm

- Helical track model
- The updated algorithm
- Implementation
- Test of the algorithm

## 3 Summary

# Motivation

- High performance of tracking is essential to the physics program on the future linear collider experiment:
  - ▶ Tracking detector: has to have excellent spatial resolution;
  - ▶ **Tracking algorithm**: has to have the ability to get momentum with high accuracy in non-uniform magnetic field.
- The Kalman filter tracking software package, Kaltest:
  - ▶ It works successfully for tracking in uniform magnetic field;
  - ▶ Code is well designed;
  - ▶ Kalman filter itself allows the non-uniformity of magnetic field can be taken into account site by site.

## Equation of motion

- The equation of motion of a charged particle in a magnetic field is

$$m \frac{d^2 \mathbf{x}}{dt^2} = Q \frac{d\mathbf{x}}{dt} \times \mathbf{B}(\mathbf{x}), \quad (1)$$

where  $m$  is the relativistic mass, and  $Q$  is the charge of particle.

- If the magnetic is uniform, and we assume its direction is parallel with  $z$  axis of coordinate system, then the trajectory of charge particle can be solved analytically,

$$\begin{cases} x = x_0 + d_\rho \cos \phi_0 + \frac{\alpha}{\kappa} [\cos \phi_0 - \cos(\phi_0 + \phi)] \\ y = y_0 + d_\rho \sin \phi_0 + \frac{\alpha}{\kappa} [\sin \phi_0 - \sin(\phi_0 + \phi)] \\ z = z_0 + d_z - \frac{\alpha}{\kappa} \tan \lambda \cdot \phi \end{cases}, \quad (2)$$

which is a helix equation.

# Helical track model

- According to the parametrized track equation, helix in  $xy$  plane is plotted as:

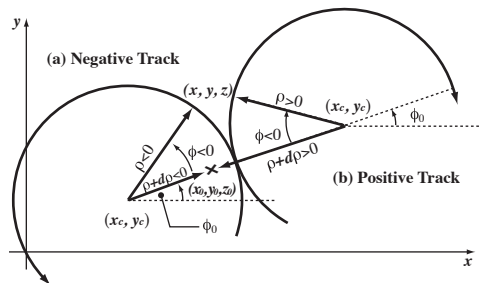


Figure 1 : Helical track model.

- The **state vector** of a track is defined as

$$\mathbf{a}_k = \left( d_\rho, \phi_0, \kappa, d_z, \tan \lambda \right)^T. \quad (3)$$

# Kalman Filter

- For each site, Kalman filter algorithm has two steps:

- ▶ Prediction:

$$\mathbf{a}_k^{k-1} = \mathbf{f}_{k-1}(\mathbf{a}_{k-1}), \quad (4)$$

in which,  $\mathbf{f}_k$  is propagation function. And the corresponding **propagation matrix** is defined by

$$\mathbf{F}_{k-1} = \frac{\partial \mathbf{f}_{k-1}}{\partial \mathbf{a}_{k-1}}. \quad (5)$$

- ▶ Filtering:

$$\mathbf{a}_k = \mathbf{a}_k^{k-1} + \mathbf{K}_k \left( \mathbf{m}_k - \mathbf{h}_k(\mathbf{a}_k^{k-1}) \right), \quad (6)$$

where  $\mathbf{K}_k$  is the gain matrix which is related to the propagation matrix,  $\mathbf{h}_k$  is the measurement function.

- Kalman filter is implemented in KalTest<sup>1</sup>, together with track models and basic detector geometries.

---

<sup>1</sup>KalTest manual is at <http://www-jlc.kek.jp/jlc/en/subg/soft/tracking/kaltest>.

## Basic idea to update the algorithm

To use the helical track model of KalTest in the non-uniform magnetic field, we have to:

- assume the magnetic field between two nearby layers is uniform;
- transform the frame to make the  $z$  axis point to the direction of magnetic field.

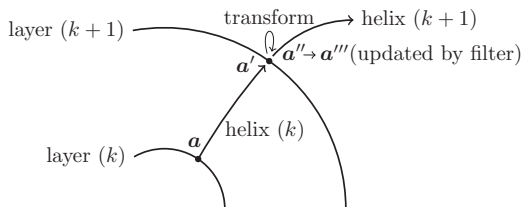


Figure 2 : The updated track propagation procedure.

Therefore we now have a **segment-wise helical track model**.

## How to transforming the frame

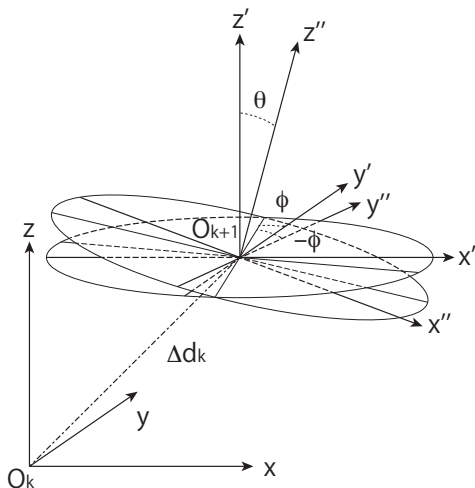


Figure 3 : Transformation: translation and rotation



## Rotation matrix

- Define the rotation to make the  $z'$  axis rotate by angle  $\theta$  in the  $z'Oz''$  plane, so that the  $z''$  axis pointing to the B-field direction at the new site. Then the rotation matrix is equivalent to the product of three successive rotations:

$$\Delta \mathbf{R} = \Delta \mathbf{R}_{z''}(-\phi) \Delta \mathbf{R}_{y''}(\theta) \Delta \mathbf{R}_{z'}(\phi). \quad (7)$$

- We use passive rotation, then the matrices are

$$\Delta \mathbf{R}_{z'}(\phi) = \begin{pmatrix} \cos \phi & \sin \phi & 0 \\ -\sin \phi & \cos \phi & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (8)$$

and

$$\Delta \mathbf{R}_{y''}(\theta) = \begin{pmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{pmatrix}. \quad (9)$$

## Modified propagator

- The propagation procedure can be represented by four equations:

$$\begin{cases} \mathbf{a}' &= \mathbf{f}_k(\mathbf{a}_k) \\ \mathbf{p} &= \mathbf{c}(\mathbf{a}') \\ \mathbf{p}' &= \mathbf{t}(\mathbf{p}) \\ \mathbf{a}'' &= \mathbf{c}^{-1}(\mathbf{p}') \end{cases} . \quad (10)$$

In Eq.(10),  $\mathbf{f}_k$  is known in the original KalTest;  $\mathbf{c}$  is the function to convert state vector to momentum, and  $\mathbf{c}^{-1}$  is its inverse function; The transformation function  $\mathbf{t}$  is actually the **rotation matrix**.

- Therefore, the propagation matrix should be modified accordingly:

$$\mathbf{F}_{k-1}^m = \frac{\partial \mathbf{a}''}{\partial \mathbf{p}'} \frac{\partial \mathbf{p}'}{\partial \mathbf{p}} \frac{\partial \mathbf{p}}{\partial \mathbf{a}'} \frac{\partial \mathbf{a}'}{\partial \mathbf{a}} = \mathbf{F}_{k-1}^r \mathbf{F}_{k-1}. \quad (11)$$

# Implementation

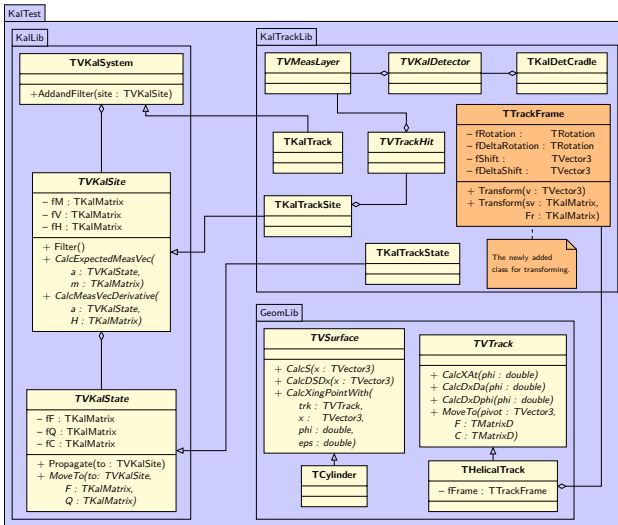


Figure 4 : Class diagram of KalTest

## Simulation conditions

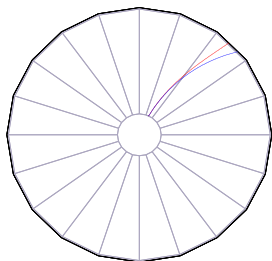
- Suppose the non-uniform magnetic field has a form of

$$\begin{cases} B_x &= B_0 kxz \\ B_y &= B_0 kyz \\ B_z &= B_0(1 - kz^2) \end{cases},$$

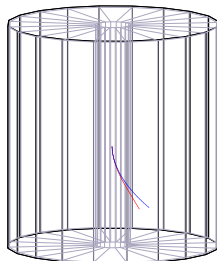
in which,  $k = \frac{k_0}{z_m r_m}$ ,  $B_0 = 3 \text{ T}$ ,  $z_m = r_m = 3000 \text{ mm}$ ;

- Runge-Kutta track generator: TEveTrackPropagator in ROOT and bisection method are used;
- Track parameters: dip angle  $\lambda \in [0, 0.5]$ , azimuth angle  $\phi \in [0, 2\pi]$ ;
- Detector:
  - ▶ 251 layers;
  - ▶ distance between two nearby layers is 6 mm;
  - ▶  $R_{\text{in}} = 300 \text{ mm}$ ;
  - ▶ Point resolution  $\sigma_{r\phi} = 100 \text{ }\mu\text{m}$ .
- To evaluate the non-uniform magnetic field, the tracks with the same initial parameters are also simulated in uniform magnetic field.

## Event display



(a)  $xy$  view



(b) 3D view

**Figure 5 :** Event display. 2 GeV tracks generated in uniform magnetic field (blue curve), and non-uniform magnetic field (red curve,  $k_0 = 5$ ).

# Momentum resolution

- $k_0 = 1$ ,  $p = 10$  GeV;
- Tracks are reconstructed in **uniform** magnetic field.

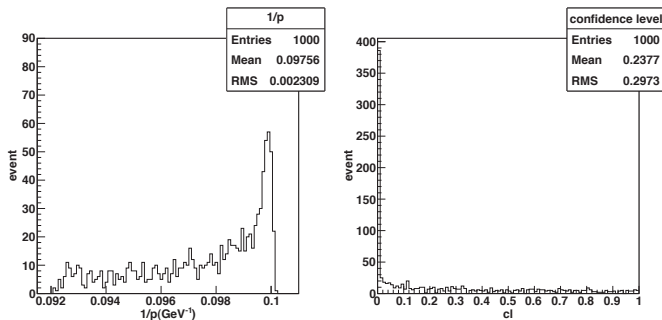


Figure 6 : Momentum and confidence level by the original algorithm.

# Momentum resolution

- $k_0 = 1$ ,  $p = 10$  GeV;
- Tracks are reconstructed in **non-uniform** magnetic field.

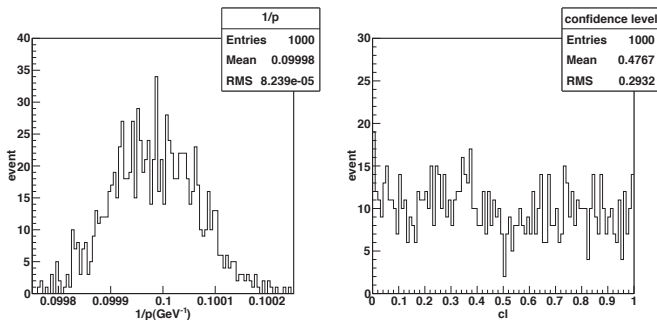


Figure 7 : Momentum and confidence level by the updated algorithm.

# Results with different non-uniformity and tracking step size

Table 1 : Mean and RMS of  $\frac{1}{p}$  (in units of  $10^{-1} \cdot (\text{GeV}/c)^{-1}$  and  $10^{-5} \cdot (\text{GeV}/c)^{-1}$  respectively) at  $10 \text{ GeV}/c$ .

(a) Step size 6 mm

$k_0$	$\lambda = 0.1$	$\lambda = 0.3$	$\lambda = 0.5$
1	1.0000/8.03	0.9998/7.89	0.9995/7.65
2	1.0000/8.05	0.9997/8.09	0.9990/8.36
3	0.9999/8.07	0.9995/8.31	0.9984/9.20

(b) Step size 1 mm

$k_0$	$\lambda = 0.1$	$\lambda = 0.3$	$\lambda = 0.5$
1	1.0000/8.03	1.0000/7.89	0.9999/7.65
2	1.0000/8.05	0.9999/8.10	0.9998/8.36
3	1.0000/8.07	0.9999/8.32	0.9997/9.21



## CPU expense

- MacBook Pro, OS X 10.6; 2.4 GHz Intel Core 2 Duo; 4 G Memory.
- 1,000 tracks

Table 2 : Time consumption of functions (sec.)

Function	Time expense
Total	18.82
TVKalState::Propagate	11.53
TVKalSite::Filter	7.27
TTrackFrame::TTrackFrame	0.87
TTrackFrame::TransformVector	6.59
TTrackFrame::TransformSv	2.58
TVSurface::CalcXingPointWith	5.90

- The total CPU time is only about two times of the original code.

# Summary

- Algorithm:
  - The algorithm for non-uniform magnetic field is based on Kalman filter;
  - By transforming the track frame, helical track model in KalTest can be used and the non-uniformity of magnetic field is taken into account.
  - A frame transformation class is made and the modifications are well localized to the track model class, so that the algorithm should also be easily ported to track fitting in detectors other than ILD.
- Test results:
  - The updated algorithm can get reasonable momentum results;
  - Good performance on CPU expense, and it can be optimized further.
  - In the high non-uniformity situation, using smaller tracking step size can improve the fitting results.
- Further plans:
  - Simulation with the magnetic field map of LPTPC/LCTPC.