

Track Reconstruction: the trf & ftf toolkits

Norman Graf (SLAC)

ILD Software Meeting, DESY
July 6, 2010

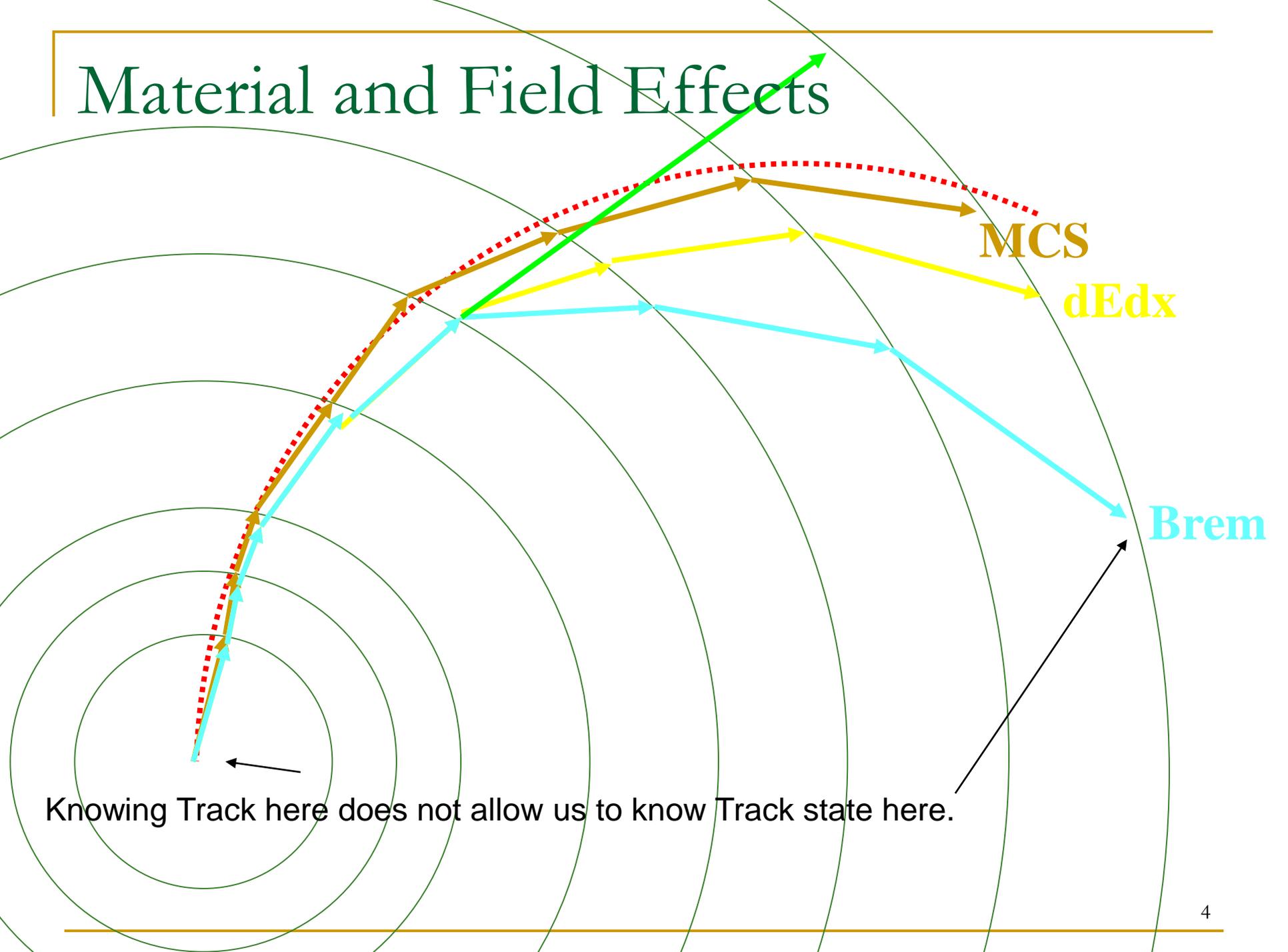
What is a track?

- Ordered association of digits, clusters or hits (finder)
 - Digit = data read from a detector channel
 - Cluster = collection of digits
 - Hit = Cluster (or digit) + calibration + geometry (+track candidate)
 - Provides a measurement suitable to fit a track
 - E.g. a 1D or 2D spatial measurement on a plane
- Trajectory through space (fitter)
 - Space = 6D track parameter space
 - 3 position + 2 direction + 1 curvature
 - 5 parameters and error matrix at any surface
- Track is therefore only piecewise helical.
 - default is to break track down by measurement layers.
 - could increase granularity for inhomogeneous fields

Track Definition

- Six parameters are required to determine a charged particle's ideal path in a magnetic field.
- However, knowing these parameters at a single point (e.g. the distance of closest approach to the beam, **dca**) is insufficient for precision fits due to material effects (dE/dx, MCS, bremsstrahlung) and field inhomogeneities.
 - No global functional form for the fit.
- Current LCIO Track interface definition is too simplistic by not allowing for these effects.

Material and Field Effects



Knowing Track here does not allow us to know Track state here.

Infrastructure components

■ Hit

- ❑ Defined at a surface.
- ❑ Provides a measurement and associated error
- ❑ Provides a mechanism to predict the measurement from a track fit
- ❑ Provides access to underlying cluster and/or digits

TrackerHit

- Current LCIO TrackerHit interface only accommodates three dimensional hits.
- Many tracking subdetectors only provide one dimensional measurements (silicon microstrips) or two dimensional hits (such as silicon pixels).
- Furthermore, using Cartesian coordinates is not always the most natural for individual subdetectors.
- Cylinder:
 - 1D Axial: ϕ
 - 1D Stereo: $\phi + \kappa z$
 - 2D Combined: (ϕ, z)
- XYPlane:
 - 1D Stereo: $w_v * v + w_z * z$
 - 2D Combined: (v, z)
- ZPlane:
 - 1D Stereo: $w_x * x + w_y * y$
 - 2D Combined: (x, y)

Hits

■ trfcyl:

- HitCylPhi : a phi measurement on a cylinder.
- HitCylPhiZ : stereo measurement on a cylinder.
 - $\text{phiz} = \text{phi} + \text{stereo} * z$.
- HitCylPhiZ2D : measurement of both phi and z on a cylinder.

■ trfxyp:

- HitXYPlane1 : one dimensional v-z measurement on a XYPlane.
 - $\text{avz} = \text{wv} * v + \text{wz} * z$
- HitXYPlane2 : two dimensional (v,z) measurement on an XYPlane

■ trfzp:

- HitZPlane1 : one dimensional xy measurement on a ZPlane.
 - $\text{axy} = \text{wx} * x + \text{wy} * y$
- HitZPlane2 : two dimensional (x,y) measurement on a ZPlane

Surfaces

- Surfaces generally correspond to geometric shapes representing detector devices.
- They provide a basis for tracks, and constrain one of the track parameters.
- The track vector at a surface is expressed in parameters which are “natural” for that surface.
- Abstract interface defined, most common surface implementation provided.

Cylinder

- Surface defined coaxial with z , therefore specified by a single parameter r .
- Track Parameters: $(\phi, z, \alpha, \tan\lambda, q/p_T)$
- Bounded surface adds z_{\min} and z_{\max} .
- Supports 1D and 2D hits:
 - 1D Axial: ϕ
 - 1D Stereo: $\phi + \kappa z$
 - 2D Combined: (ϕ, z)

XY Plane

- Surface defined parallel with z , therefore specified by distance u from the z axis and an angle ϕ of the normal with respect to x axis.
- Track Parameters: (**v , z , dv/du , dz/du , q/p**)
- Bounded surface adds polygonal boundaries.
- Supports 1D and 2D hits:
 - 1D Stereo: $w_v * v + w_z * z$
 - 2D Combined: (v, z)

Z Plane

- Surface defined perpendicular to z, therefore specified by single parameter z.
- Track Parameters: (**x, y, dx/dz, dy/dz, q/p**)
- Bounded surface adds polygonal boundaries.
- Supports 1D and 2D hits:
 - 1D Stereo: $w_x * x + w_y * y$
 - 2D Combined: (x,y)

Distance of Closest Approach

- DCA is also a 5D **Surface** in the 6 parameter space of points along a track.
- It is **not** a 2D surface in 3D space.
- Characterized by the track direction and position in the (x,y) plane being normal; $\alpha=\pi/2$.
- Track Parameters: (**r**, **z**, ϕ_{dir} , **tan** λ , **q/p_T**)

Detector

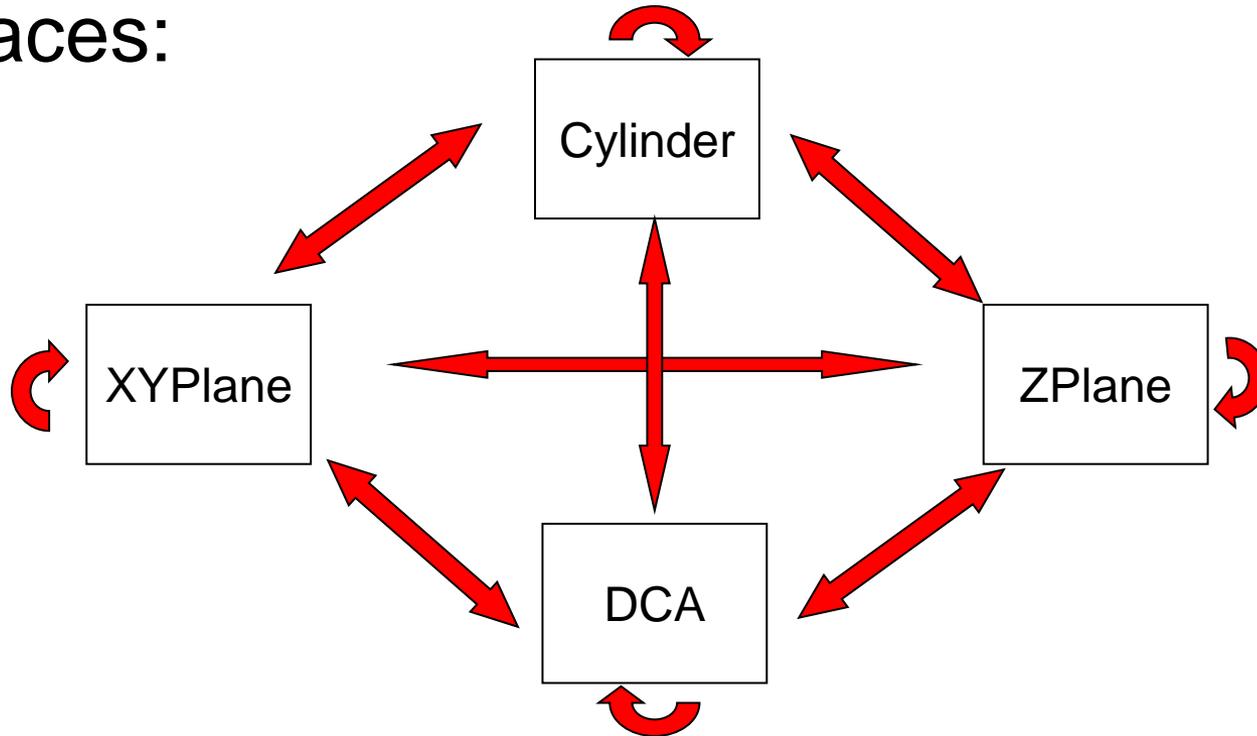
- Use compact.xml to create a tracking Detector composed of surfaces, along with interacting propagators to handle track vector and covariance matrix propagation, as well as energy loss and multiple scattering.
- Convert SimTrackerHits in event into:
 - 1-D phi measurements in Central Tracker Barrel
 - 2-D phi-z measurements in Vertex Barrel (pixel)
 - 2-D x-y measurements in forward disks (assume stereo strips)
 - 2-D phi-z measurements in TPC (place hits on cylinders in middle of readout pads)

Propagator

- Propagators propagate a track (and optionally its covariance matrix) to a new surface.
- A propagator returns an object of type PropStat which describes the status of the attempted propagation:
 - *i.e.* whether it was successful and, if so, in which direction the track was propagated (forward or backward).
- Interacting Propagators modify the track and its covariance matrix (in case of energy loss), or just the covariance matrix (thin multiple scattering.)

Propagators

- Propagators are defined for all combinations of surfaces:



- Provide both simple, but fast, constant-field and full, Runge-Kutta propagators.

Interactors

- Describes the interface for a class which modifies an ETrack. Examples are:
- Multiple Scattering
 - ThinCyIMS
 - ThickCyIMS
 - ThinXYPlaneMS
 - ThinZPlaneMS
- Energy Loss
 - CylELoss

Track Finding: ftf

- Using a conformal mapping technique
 - Maps curved trajectories onto straight lines
 - Simple link-and-tree type of following approach associates hits.
 - Once enough hits are linked, do a simple helix fit
 - circle in r-phi
 - straight line in s-z
 - simple iteration to make commensurate
 - Use these track parameters to predict track into regions with only 1-D measurements & pick up hits.
 - Outside-in, inside-out, cross-detector: completely flexible as long as concept of *layer* exists.
 - Simple fit serves as input to final Kalman fitter.

Summary

- Improvements are being considered for the LCIO tracker hit and track infrastructure and the ILD track finding and fitting packages.
- trf toolkit contains a well-tested detector model, track & hit classes and Kalman filter fitting code which accounts for energy loss and MCS.
- ftf toolkit provides a fast, efficient, pattern recognition package based on a conformal mapping of hits on topological layers.
- Implementing ftf & trf into the ILD software would require some work.