SeedTracker Update

- Material Modeling for Planar Geometry (Cosmin)
- Forward Tracking (Partridge)

Cosmin Deaconu and Richard Partridge

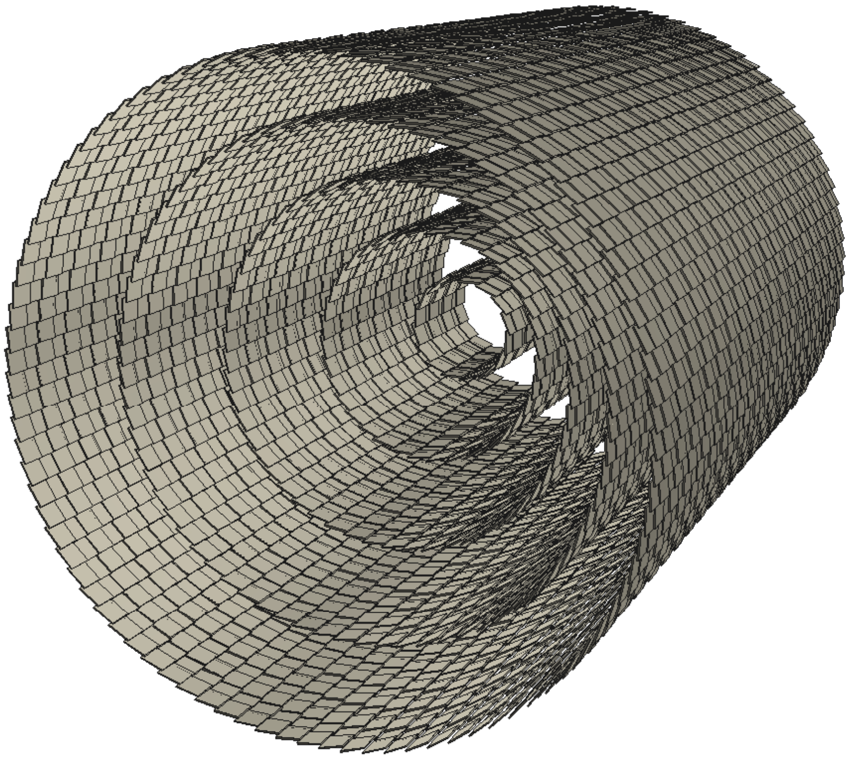
June 6, 2008

Material Modeling for Planar Geometry

- SeedTracker uses a χ^2 function for essentially all decisions
  - χ^2 takes into account residuals from the helix fit, any pulls required to meet kinematic constraints (eg p_T > xx), and in the future will include any pulls required to meet geometric constraints (eg track just missed end of strip)
- For most tracks, hit resolution is dominated by multiple scattering errors
  - Need to do a good job of modeling multiple scattering errors

Planar Geometry

- Detector Model has 1000s of individual geometry elements
- Inefficient to model material for each element individually
- Group together all elements that are daughters of a common compact.xml definition
- Model grouped elements as thin cylinders or disks



Geometry Trouble

- Setting up SeedTracker to use planar hits instead of virtual segmentation hits was up and running within ~1 hour of first attempt
  - SeedTracker was designed to be flexible on where hits came from
- Cosmin began looking at hit residuals, and saw very large tails
- Problem was traced to errors in material modeling
  - Coordinate transformations from local to global coordinate systems was not being done
  - Enormous amount of material lumped in small region
  - Tracks that traversed this region were given enormous multiple scattering errors, leading to large residuals since outer tracker hits were effectively ignored

Geometry Trouble - Solved

- New geometry system developed by Jeremy and Tim has hierarchical architecture
- Properties of each element are stored in bottom level of hierarchy (shape, material, local coordinates)
- Once at the bottom of the hierarchy, no methods exist to transform your local coordinates back to global coordinates
  - Not a problem for DetectorElements, which provide these methods
  - However, we want to pick up all dead material, not just the DetectorElements
- Need to construct a "path" as you work your way down from the top of the hierarchy to the bottom
  - Given such a path, you can get the local to global transform
  - Cosmin finally got this working yesterday after 2-3 weeks working with Tim and RP to understand geometry system
- This was much too hard to make work correctly!!

Forward Tracking

- Forward tracking differs from barrel tracking in two fundamental ways
  - Stereo sensor pairs in the forward region
  - Measurements coordinates are effectively $r^*\phi$ and r, not $r^*\phi$ and z
- Handling stereo hits ended up being fairly complicated
- Change in measurement coordinate was fairly straight-forward

Stereo Hits

- Tracker endcap stereo layers are separated by 4 mm in SiD01
  - This distance is nearly 3 orders of magnitude greater than the ~7 μm intrinsic resolution for a strip sensor
- Unless the track is traveling normal to the sensor planes, the x-y hit position will be different in the two stereo sensors
  - Hits formed from a stereo pair will in general end up in the wrong place unless you account for the direction of the track
  - ~1 mrad resolution on track direction needed before hit resolution is dominant
- Conundrum for tracking in the forward direction:
  - Can't fit the helix without knowing the hit positions
  - Can't determine the hit positions without knowing the helix

Iterative Solution

- A track seed's first fit is used to estimate helix so multiple scattering errors can be calculated
  - For this first fit, calculate stereo hit positions assuming the track is from the origin, with large hit position due to uncertainties in the track direction
- The track seed is immediately re-fit including multiple scattering errors
  - Use track direction and helix errors to generate a corrected position and covariance matrix for each stereo hit
- Additional fits are performed as hits are added to the track seed, further reducing uncertainties in the track direction

Stereo Hit Coordinates

- Let sensor coordinates be given by u, v, and w
  - u is the measurement coordinate
  - v is the coordinate along the strip direction
  - w is the coordinate normal to the sensor surface
- Take sensor origin to be the point in the sensor plane (w=0) where u=v=w=0
- Hit positions in the stereo layer pair are then given by

$$\vec{r}_1 = \vec{O}_1 + u_1 \hat{u}_1 + v_1 \hat{v}_1$$

$$\vec{r}_2 = \vec{O}_2 + u_2 \hat{u}_2 + v_2 \hat{v}_2$$

- The stereo hit coordinate is obtained by taking the midpoint between the hit positions in the two layers

$$\vec{r} = \frac{\vec{r}_1 + \vec{r}_2}{2}$$

Straight-Line Track from Origin

- For the first fit, we assume we have a straight-line track from the origin in calculating stereo hit posiitons

$$\vec{r}_2 = \gamma \vec{r}_1$$

$$\vec{O}_2 + u_2 \hat{u}_2 + v_2 \hat{v}_2 = \gamma \vec{O}_1 + \gamma u_1 \hat{u}_1 + \gamma v_1 \hat{v}_1$$

- 3 linear equations in 3 unknowns (γ, v_2, and γv_1)
  - Solvable, but algebraically messy
  - Don't even want to think about error matrix…

Straight-Line Track from Origin II

- Problem simplifies considerably if we assume sensors are parallel $\hat{u}_1 \times \hat{v}_1 = \hat{u}_2 \times \hat{v}_2 = \hat{w}$
- Can now solve for γ, v_1, v_2, and hit position

$$\gamma = \frac{\vec{O}_2 \bullet \hat{w}}{\vec{O}_1 \bullet \hat{w}}$$

$$v_1 = \frac{(\vec{O}_2 + u_2\hat{u}_2 - \gamma\vec{O}_1 - \gamma u_1\hat{u}_1) \bullet \hat{u}_2}{\gamma\hat{v}_1 \bullet \hat{u}_2}$$

$$v_2 = \frac{(\vec{O}_2 + u_2\hat{u}_2 - \gamma\vec{O}_1 - \gamma u_1\hat{u}_1) \bullet \hat{u}_1}{\hat{v}_1 \bullet \hat{u}_2} \qquad \hat{v}_2 \bullet \hat{u}_1 = -\hat{v}_1 \bullet \hat{u}_2$$

$$\vec{r} = \frac{(1+\gamma)}{2}\left(\vec{O}_1 + u_1\hat{u}_1 + \frac{(\vec{O}_2 + u_2\hat{u}_2 - \gamma\vec{O}_1 - \gamma u_1\hat{u}_1) \bullet \hat{u}_2}{\gamma\hat{v}_1 \bullet \hat{u}_2}\hat{v}_1\right)$$

Track with Known Direction

- Second fit uses track direction from first fit to estimate stereo hit positions
- Assume momentum vector is constant between sensor planes

$$\vec{r}_2 = \vec{r}_1 + \gamma\,\hat{p}$$

$$\vec{O}_2 + u_2\hat{u}_2 + v_2\hat{v}_2 = \vec{O}_1 + u_1\hat{u}_1 + v_1\hat{v}_1 + \gamma\,\hat{p}$$

- Can solve for unmeasured coordinates and hit position

$$\gamma = \frac{\left(\vec{O}_2 - \vec{O}_1\right)\bullet\hat{w}}{\hat{p}\bullet\hat{w}}$$

$$v_1 = \frac{(\vec{O}_2 + u_2\hat{u}_2 - \vec{O}_1 - u_1\hat{u}_1 - \gamma\,\hat{p})\bullet\hat{u}_2}{\hat{v}_1\bullet\hat{u}_2}$$

$$v_2 = \frac{(\vec{O}_2 + u_2\hat{u}_2 - \vec{O}_1 - u_1\hat{u}_1 - \gamma\,\hat{p})\bullet\hat{u}_1}{\hat{v}_1\bullet\hat{u}_2}$$

Hit Position Uncertainty

- For first fit, use previous equations with uncertainty:

$$\sigma(\hat{p} \bullet \hat{u}_i) = 2/\sqrt{12}$$

- For fits where track direction is known, there are two components to the hit uncertainty:
  - Uncertainties in measured coordinates u_1 and u_2
  - Uncertainty in track direction

$$\vec{r} = \frac{\vec{r}_1 + \vec{r}_2}{2}$$

$$\delta\vec{r} = \frac{\partial\vec{r}}{\partial u_1}\delta u_1 + \frac{\partial\vec{r}}{\partial u_2}\delta u_2 + \sum_i \frac{\partial\vec{r}}{\partial\hat{p}_i}\delta\hat{p}_i$$

$$\delta\hat{p}_i = \frac{\partial\hat{p}_i}{\partial\omega}\delta\omega + \frac{\partial\hat{p}_i}{\partial d_0}\delta d_0 + \frac{\partial\hat{p}_i}{\partial\phi_0}\delta\phi_0 + \frac{\partial\hat{p}_i}{\partial z_0}\delta z_0 + \frac{\partial\hat{p}_i}{\partial\tan\lambda}\delta\tan\lambda$$

Hit Position Uncertainty II

- Coordinate r given by

$$\vec{r} = \frac{\vec{O}_1 + u_1\hat{u}_1}{2} + \frac{(\vec{O}_2 + u_2\hat{u}_2 - \vec{O}_1 - u_1\hat{u}_1 - \gamma\hat{p}) \bullet \hat{u}_2}{2\hat{v}_1 \bullet \hat{u}_2}\hat{v}_1$$

$$+ \frac{\vec{O}_2 + u_2\hat{u}_2}{2} + \frac{(\vec{O}_2 + u_2\hat{u}_2 - \vec{O}_1 - u_1\hat{u}_1 - \gamma\hat{p}) \bullet \hat{u}_1}{2\hat{v}_1 \bullet \hat{u}_2}\hat{v}_2$$

- Measured coordinate contributions:

$$\frac{\partial\vec{r}}{\partial u_1} = \frac{\hat{u}_1}{2} - \frac{\hat{u}_1 \bullet \hat{u}_2\hat{v}_1}{2\hat{v}_1 \bullet \hat{u}_2} - \frac{\hat{v}_2}{2\hat{v}_1 \bullet \hat{u}_2} = -\frac{\hat{v}_2}{\hat{v}_1 \bullet \hat{u}_2}$$

$$\frac{\partial\vec{r}}{\partial u_2} = \frac{\hat{v}_1}{2\hat{v}_1 \bullet \hat{u}_2} + \frac{\hat{u}_2}{2\hat{v}_1 \bullet \hat{u}_2} + \frac{\hat{u}_1 \bullet \hat{u}_2\hat{v}_2}{2\hat{v}_1 \bullet \hat{u}_2} = \frac{\hat{v}_1}{\hat{v}_1 \bullet \hat{u}_2}$$

Hit Position Uncertainty III

- Direction Derivatives:

$$D_{i,j} \equiv \frac{\partial r_i}{\partial \hat{p}_j}$$

$$= -\frac{\left(\vec{O}_2 - \vec{O}_1\right) \bullet \hat{w}}{2\hat{v}_1 \bullet \hat{u}_2} \frac{\partial}{\partial \hat{p}_j} \frac{\hat{p} \bullet \hat{u}_2 \hat{v}_{1,i} + \hat{p} \bullet \hat{u}_1 \hat{v}_{2,i}}{\hat{p} \bullet \hat{w}}$$

$$= -\frac{\left(\vec{O}_2 - \vec{O}_1\right) \bullet \hat{w}}{2\hat{v}_1 \bullet \hat{u}_2} \left(\frac{\hat{u}_{2,j} \hat{v}_{1,i} + \hat{u}_{1,j} \hat{v}_{2,i}}{\hat{p} \bullet \hat{w}} - \frac{\hat{p} \bullet \hat{u}_2 \hat{v}_{1,i} + \hat{p} \bullet \hat{u}_1 \hat{v}_{2,i}}{\left(\hat{p} \bullet \hat{w}\right)^2} \hat{w}_j \right)$$

$$= \frac{\left(\vec{O}_2 - \vec{O}_1\right) \bullet \hat{w}}{2\hat{v}_1 \bullet \hat{u}_2 \left(\hat{p} \bullet \hat{w}\right)^2} \left(v_{1,i} \left(\hat{p} \times \hat{v}_2\right)_j + v_{2,i} \left(\hat{p} \times \hat{v}_1\right)_j \right)$$

Hit Position Uncertainty IV

- Helix derivatives:

$$H_{i,\alpha} = \frac{\partial \hat{p}}{\partial x_\alpha} \qquad x_\alpha = \{\omega, d_0, \phi_0, z_0, \tan\lambda\}$$

$$\hat{p}_x = \cos\phi \sin\theta = \omega(y - y_C)\sin\theta = (\omega y + (1 - \omega d_0)\cos\phi_0)\sin\theta$$
$$\hat{p}_y = \sin\phi \sin\theta = -\omega(x - x_C)\sin\theta = (-\omega x + (1 - \omega d_0)\sin\phi_0)\sin\theta$$
$$\hat{p}_z = \cos\theta$$

$$x_C = (\omega^{-1} - d_0)\sin\phi_0$$
$$y_C = -(\omega^{-1} - d_0)\cos\phi_0$$

Hit Position Uncertainty V

$$\frac{\partial \hat{p}_x}{\partial \omega} = \frac{\hat{p}_x}{\omega} - \frac{\cos\phi_0 \sin\theta}{\omega}$$

$$\frac{\partial \hat{p}_y}{\partial \omega} = \frac{\hat{p}_y}{\omega} - \frac{\sin\phi_0 \sin\theta}{\omega}$$

$$\frac{\partial \hat{p}_x}{\partial d_0} = -\omega \cos\phi_0 \sin\theta$$

$$\frac{\partial \hat{p}_y}{\partial d_0} = -\omega \sin\phi_0 \sin\theta$$

$$\frac{\partial \hat{p}_x}{\partial \phi_0} = -(1-\omega d_0)\sin\phi_0 \sin\theta$$

$$\frac{\partial \hat{p}_y}{\partial \phi_0} = (1-\omega d_0)\cos\phi_0 \sin\theta$$

$$\frac{\partial \hat{p}_z}{\partial \omega} = \frac{\partial \hat{p}_z}{\partial d_0} = \frac{\partial \hat{p}_z}{\partial \phi_0} = \frac{\partial \hat{p}}{\partial z_0} = 0$$

$$\frac{\partial \hat{p}_x}{\partial \tan\lambda} = -\hat{p}_x \sin\theta \cos\theta$$

$$\frac{\partial \hat{p}_y}{\partial \tan\lambda} = -\hat{p}_y \sin\theta \cos\theta$$

$$\frac{\partial \hat{p}_z}{\partial \tan\lambda} = \sin^3\theta$$

Hit Position Uncertainty VI

- Put it all together to form the covariance matrix for the hit
  - Let D be the 3x3 matrix containing the direction derivatives
  - Let H be the 3x5 matrix containing the helix derivatives
  - Let C_H be the covariance matrix for the helix parameters (ω, d_0, ϕ_0, z_0, $\tan\lambda$)
- The covariance matrix for the hit position is then given by:

$$\mathrm{cov} = \left\langle \delta\vec{r}\,\delta\vec{r}^{\,T} \right\rangle$$

$$= \frac{1}{(\hat{v}_1 \bullet \hat{u}_2)^2}\left(\sigma_{u_1}^2 \hat{v}_2 \hat{v}_2^T + \sigma_{u_2}^2 \hat{v}_1 \hat{v}_1^T\right) + DHC_H H^T D^T$$

Impact of Stereo Hits on Hit Infrastructure

- This required some replumbing of the hit infrastructure used by SeedTracker
  - Added strip class to encapsulate strip information for stereo hits
  - Added public methods to retrieve corrected hit position and covariance matrix
  - Incorporated methods in stereo hit class to set helix direction and uncertainties used in calculating the corrected hit position and covariance matrix
- TrackerHit class is now rather over-loaded with functionality…
- Persistance of this information may be an issue in the future…

Helix Fit Issues for Forward Tracking

- Only major issues is handling measurement coordinates
- For forward disks, we measure r*ϕ and r, not r*ϕ and z
- Helix fitter assumes we are fitting z vs s (s is the x-y path length of the track

$$z = z_0 + s \tan \lambda$$

- To the extent that r*ϕ measures the bend coordinate, the orthogonal coordinate r measures the non-bend coordinate
  - In this approximation, take $\delta r = \delta s$
  - This isn't exactly true because a track that has curved is not perfectly radial
  - Not clear if this is important – ignored for barrel as well
- Use an effective error in z for the s-z fit:

$$\delta z = \delta r \tan \lambda$$

First Results