Implementation of Covariance Matrix on ReconstructedParticle

C. Calancha
ILD Analysis & Software Meeting

April 16, 2014

Motivation

- ReconstructedParticle.getCovMatrix is not implemented in current `ILCSOFT` release (return 0, $\forall\, p \in$ PandoraPFOs) .
- This method provide covariance matrix of the reco. particle 4 vector {px,py,pz,E}.
- I was suggested to apply this cov. matrix to obtain dimuon mass error event-by-event basis.

- I have written Marlin processor adding new LCCollection to the event.
- This collection is a copy of PandoraPFOs but with filled cov. matrix.
- Code is available here: http://www-jlc.kek.jp/jlc/en/node/209

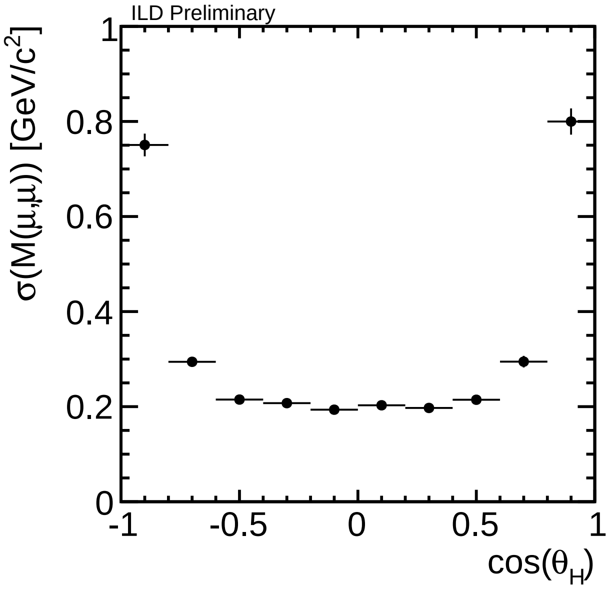
Calculation

$\Sigma_i = \Sigma_i(tan\lambda, \phi, \Omega)$, $\Sigma'_i = \Sigma'_i(px, py, pz, E)$

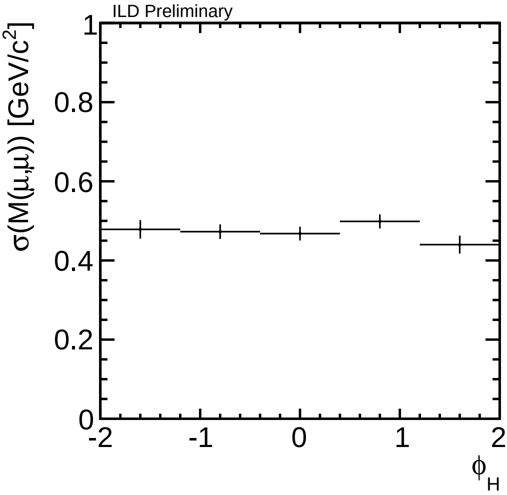
- Covariance matrix on helix parameters, Σ_i, from associated track.
- Obtain jacobian ($\mathfrak{J}$) and perform: $\Sigma'_i = \mathfrak{J}^T \, \Sigma_i \, \mathfrak{J}$

Dimuon invariant mass error

- Checked calculation with muons from $\mathrm{H} \rightarrow \mu\mu$.
- The covariance matrix is used to obtain the dimuon invariant mass event-by-event.

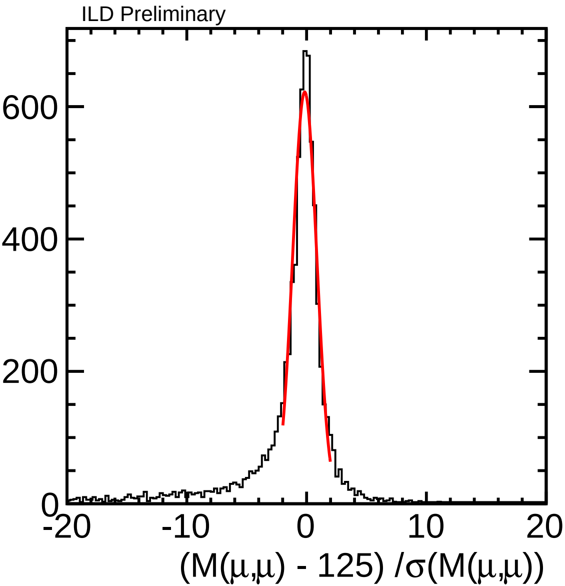


- Better precision at central region (tracks have more hits).



- No dependence on azimutal angle.

Gaussian Fit in [-2,2]



Gaussian fit [-2,2]:

- mean: -0.157
- sigma: 1.0001

Summary / Plan

Summary

- Current ILCSOFT does not provide covariance matrix on {px,py,pz,E} for the reco. particles.
- Developed Marlin processor calculating this matrix for charged particles.

Plan

- Use cov. matrix in update of $\mathrm{H} \rightarrow \mu\mu$ analysis.

BACKUP

BACK UP

Relation between variables

- Original base: $\mathfrak{A} = \{\tan\lambda\,, \Omega\,, \phi\,, d_0\,, z_0\,\}$
- New base: $\mathfrak{B} = \{p_x\,, p_y\,, p_z\,, E\}$

$$
\begin{aligned}
p_x &= p_T\, cos\phi \\
p_y &= p_T\, sin\phi \\
p_z &= p_T \tan\lambda \\
E^2 &= (a \frac{B_z}{\Omega\, cos\,\lambda})^2 + m^2 \\
&= (\frac{p_T}{cos\,\lambda})^2 + m^2
\end{aligned}
$$

$$
\begin{aligned}
p_T &= |\frac{\kappa}{\Omega}| \\
\kappa &= |a\, B_z|\ (constant)
\end{aligned}
$$

Momenta does not depend on $d_0\,, z_0$

- $p_x = p_x\,(\tan\lambda\,, \Omega, \phi)$
- $p_y = p_y\,(\tan\lambda\,, \Omega, \phi)$
- $p_z = p_z\,(\tan\lambda\,, \Omega)$

Change of cov. matrix

1. $\Sigma'_i = \mathfrak{J}^T\,\Sigma_i\,\mathfrak{J}$
2. Σ_i cov. matrix in $\mathfrak{A}$.
3. Σ'_i cov. matrix in $\mathfrak{B}$.

Jacobian helix parameters to momenta space

- After some derivative exercises ...

$$\mathfrak{J} = \begin{bmatrix} \frac{\partial P_x}{\partial tan\lambda} & \frac{\partial P_y}{\partial tan\lambda} & \frac{\partial P_z}{\partial tan\lambda} & \frac{\partial E}{\partial tan\lambda} \\ \frac{\partial P_x}{\partial \Omega} & \frac{\partial P_y}{\partial \Omega} & \frac{\partial P_z}{\partial \Omega} & \frac{\partial E}{\partial \Omega} \\ \frac{\partial P_x}{\partial d_0} & \frac{\partial P_y}{\partial d_0} & \frac{\partial P_z}{\partial d_0} & \frac{\partial E}{\partial d_0} \\ \frac{\partial P_x}{\partial z_0} & \frac{\partial P_y}{\partial z_0} & \frac{\partial P_z}{\partial z_0} & \frac{\partial E}{\partial z_0} \\ \frac{\partial P_x}{\partial \phi} & \frac{\partial P_y}{\partial \phi} & \frac{\partial P_z}{\partial \phi} & \frac{\partial E}{\partial \phi} \end{bmatrix} = \frac{-1}{\Omega} \begin{bmatrix} 0 & 0 & -\Omega P_T & -\frac{P_z^2 \,\Omega}{E\, tan\lambda} \\ P_x & P_y & P_z & \frac{P^2}{E} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ P_y\,\Omega & -P_x\,\Omega & 0 & 0 \end{bmatrix}$$

$\longrightarrow$ $\Sigma'_i = \mathfrak{J}^T\,\Sigma_i\,\mathfrak{J}$, covariance matrix in momenta space.

($\Sigma'_i = \mathfrak{J}\,\Sigma_i\,\mathfrak{J}^T$ if you define jacobian as the transposed of quoted above)