

Modeling SVT Pixel Resolution in the Fast Simulation

John Walsh
INFN, Pisa

What is a resolution model?

- Track fitter takes position measurements (hits) and reconstructs track trajectory
- Hit positions and *uncertainties* are needed as input to the track fitter
- Resolution model is simply an estimate of the hit uncertainty, which may depend on detector properties (pitch, thickness, readout details) and (possibly) track properties (momentum, *angle*)

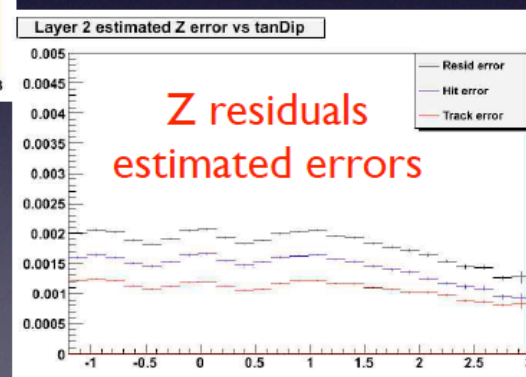
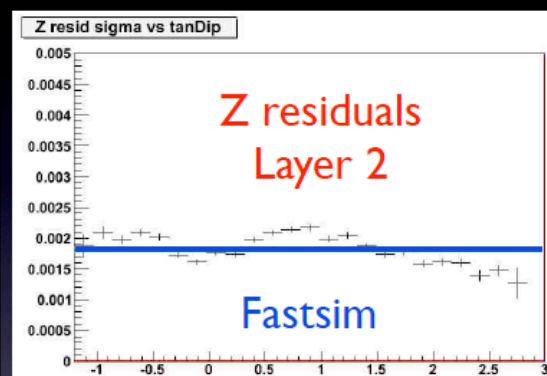
Currently in Fastsim

Based on results from Babar SVT

(Dave Brown, Paris, 2/09)

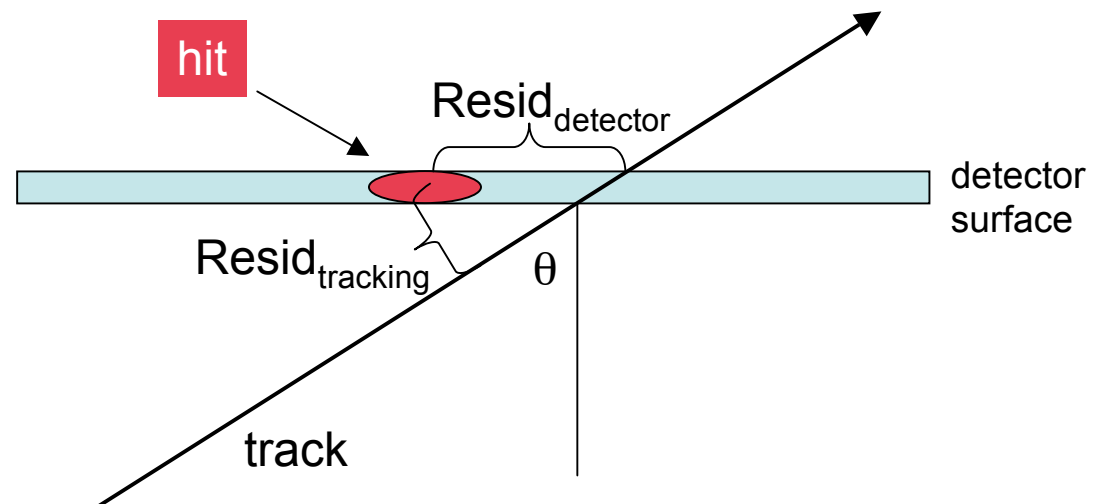
- Constant hit resolution - Gaussian with non-Gaussian tails
- Good for layers 1-5 of SuperB SVT
- Current baseline for Layer0 in SuperB: 50x50 μm^2 hybrid pixels
- Resolution significantly different than for BaBar-style strips
 1. no floating strips
 2. digital readout

Resolution vs $\tan(\text{dip})$



Definition of residual/resolution

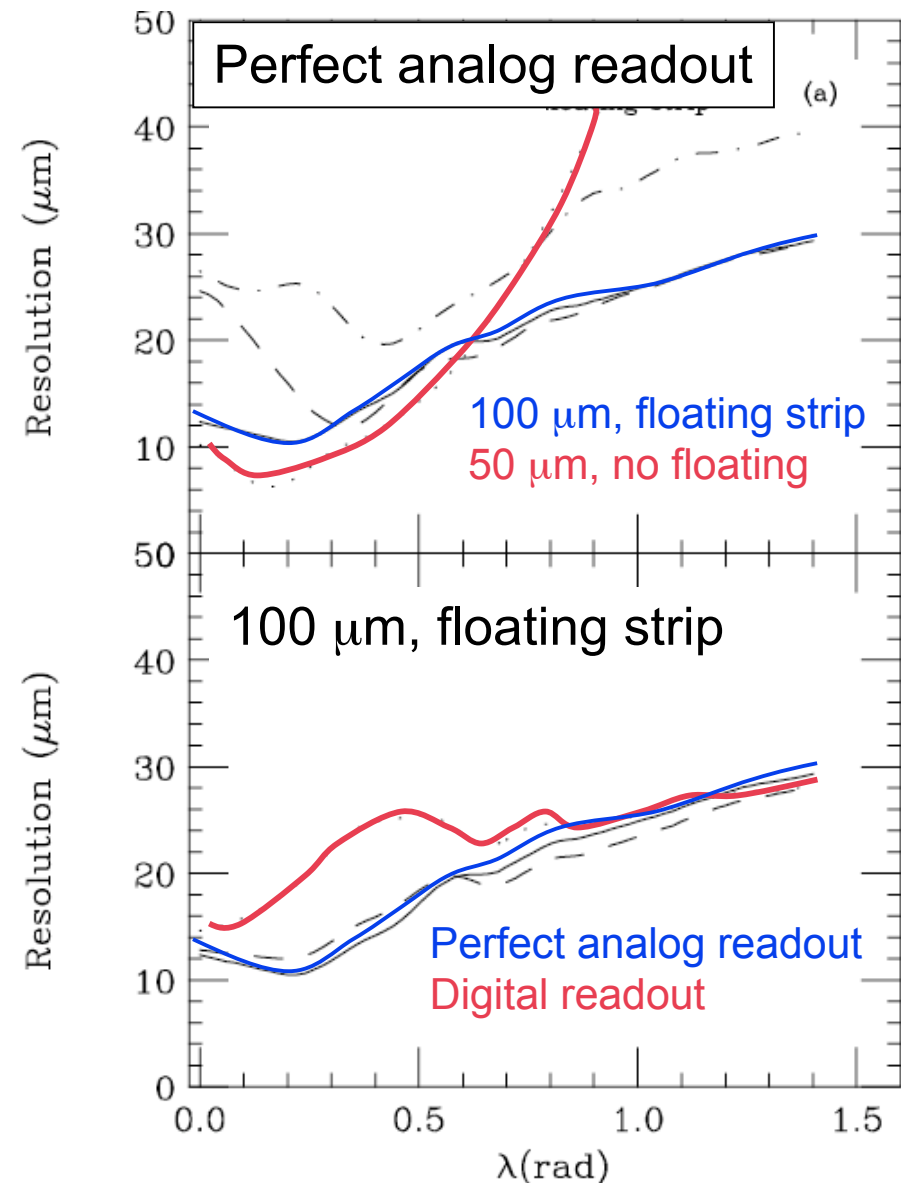
- Residual: measured distance between track and hit
- Resolution: intrinsic resolution of detector, subtracting out track extrapolation uncertainty
- Detector people: resolution measured in detector plane
- Tracking folks: resolution measured perpendicular to track
- Will refer to both types of residual/resolution in this talk



Angle: projected into measurement plane.

Expected resolution in Silicon detectors

- Based on simulations performed for Babar TDR
- Predict hit resolution in wafer plane for silicon strips for different pitch/readout configurations
- Expect similar results for pixels
- **Results:**
 - 50 μm pitch (no floating strips) shows large resolution at larger angles
 - digital readout gives worse resolution at moderate angles



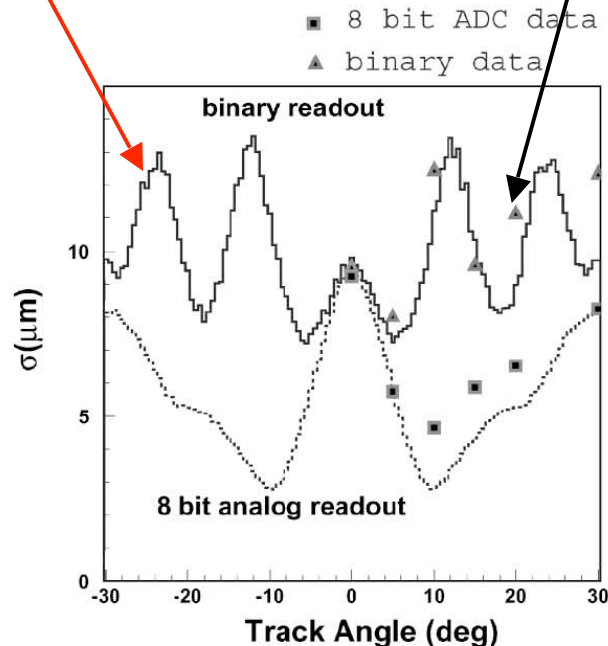
BTEV and Atlas beam tests: 400x50 μm pixels with digital readout

Resolution in wafer plane

BTEV

testbeam measurements

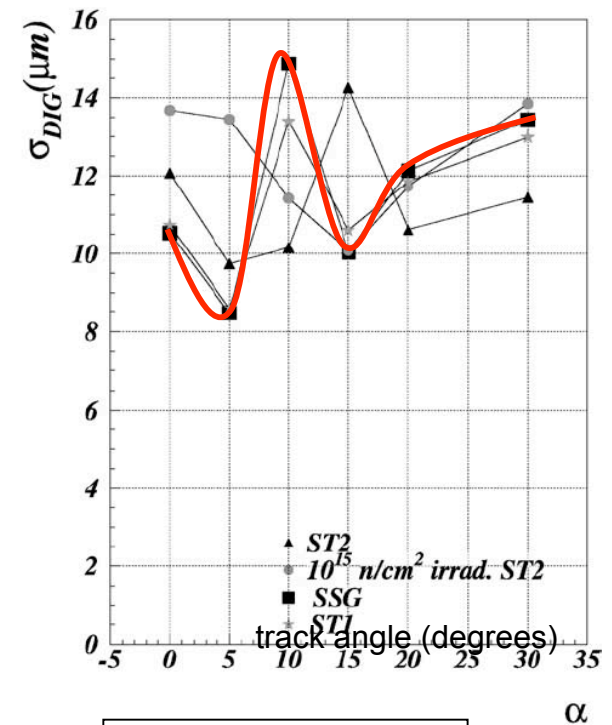
simulation with digital readout



NIM A 465 (2001) 115.

- Original BaBar simulations confirmed in Atlas and BTEV beam tests
- Curious oscillation in resolution due to digital readout
- Measurements only to 30 degrees -- in SuperB we will have angles up to ~ 75 degrees

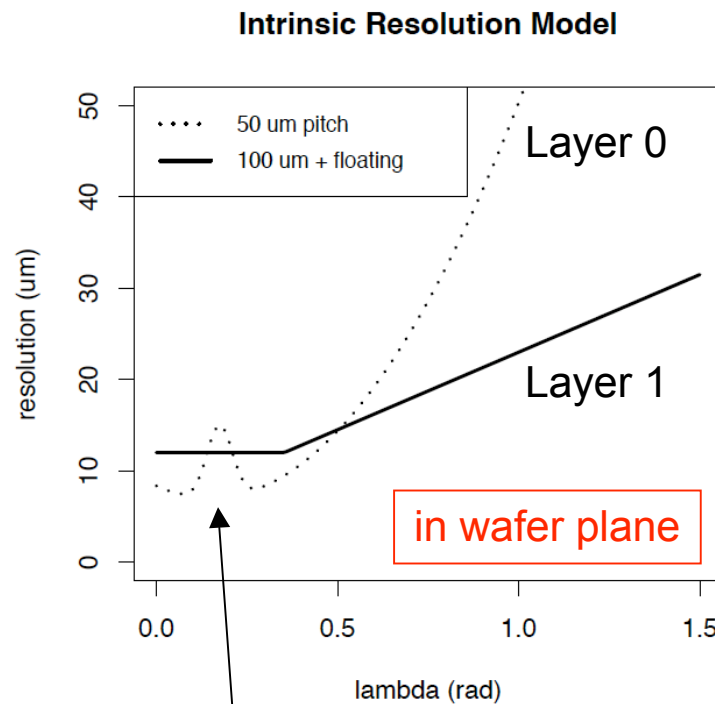
Atlas



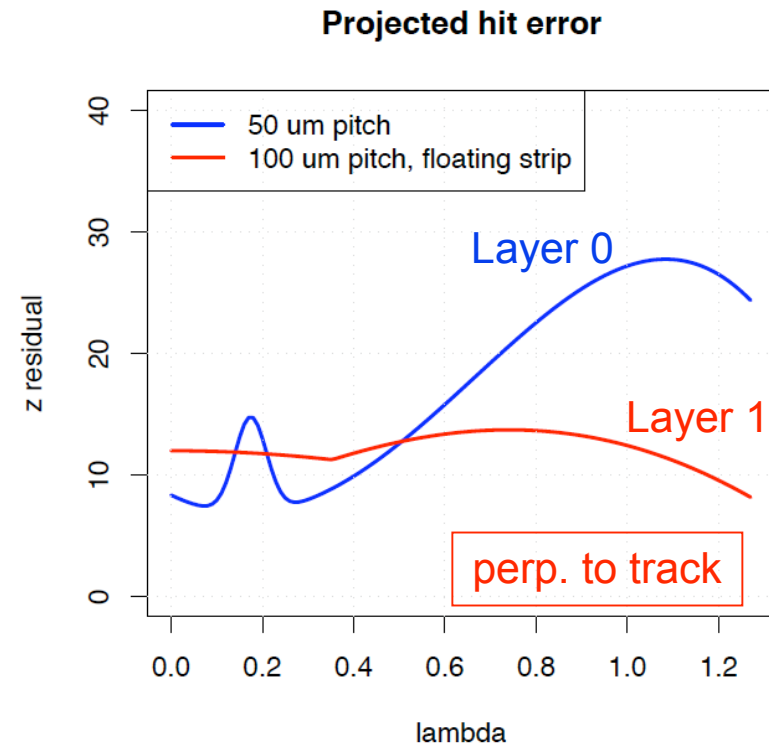
NIM A 481 (2002) 204.

Parametrization of resolution for Layer 0

- Parabolic form with gaussian “wobble”
- Layer 1 parametrization shown for reference

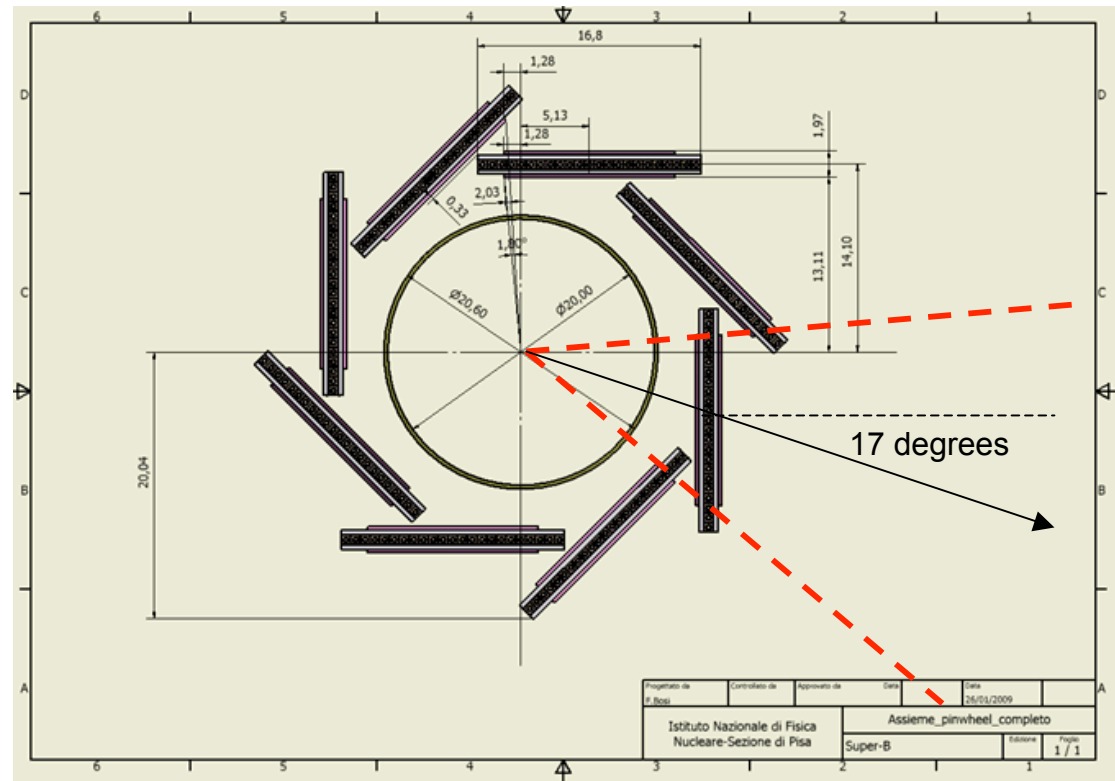


model only 1 wiggle



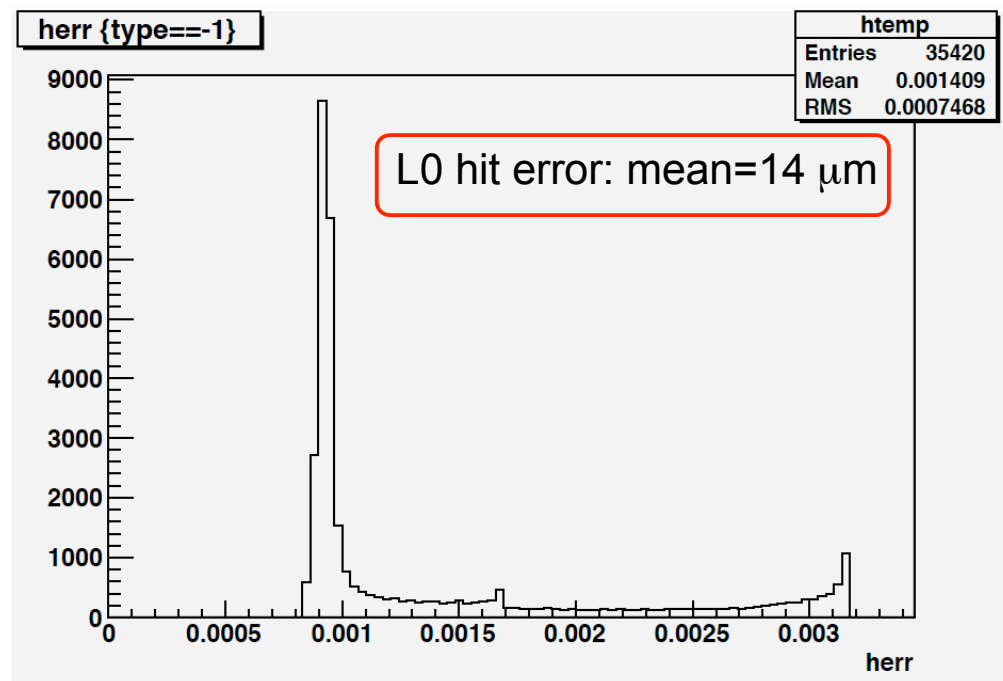
Phi side adjustment

- Due to cylindrical approximation in fastsim, average track incident angle is 0 degrees in phi projection
- The preliminary L0 layout shows a average phi incident angle of 17 degrees
- I add 17 degrees to angle phi-projected angle when calculating the hit error

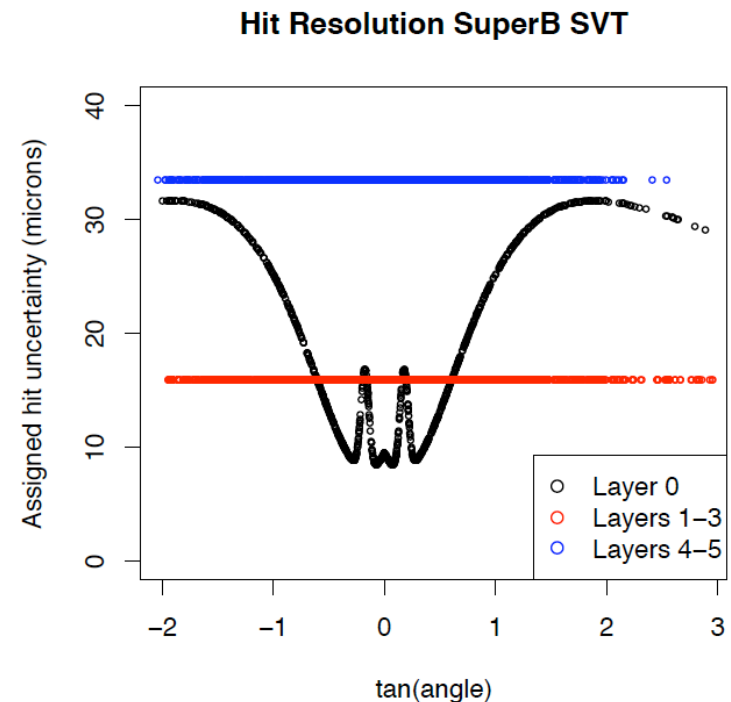


Overall effect on tracking

- Overall effect is very small on average (see next slide)
- With new model, average hit error for L0 = 14 μm , close to value used in old (constant) model

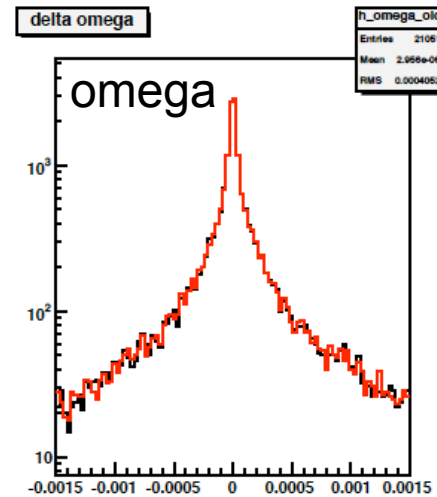
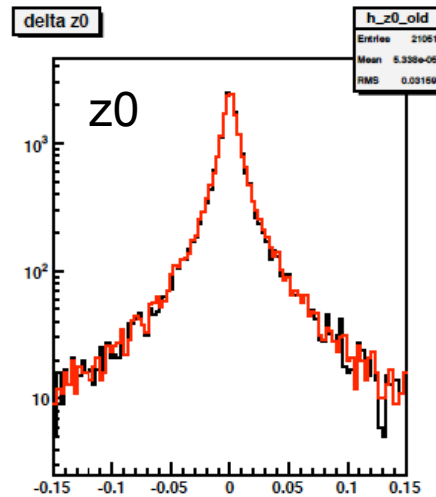
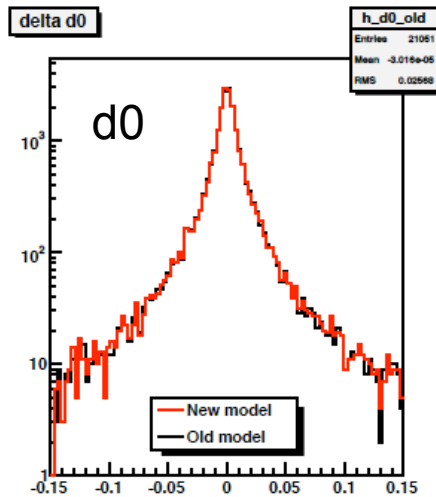


Sanity check: assigned hit error in fastsim:

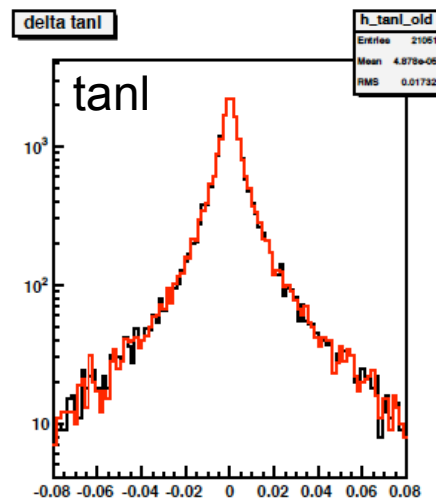
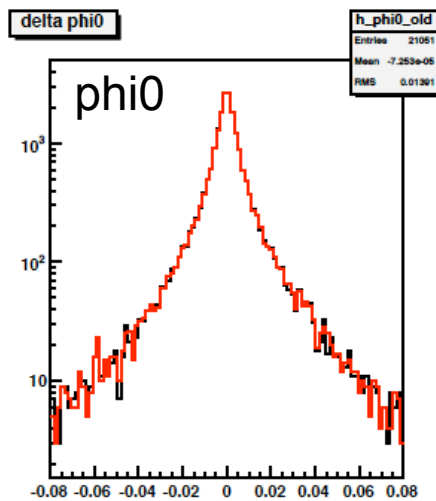


Track parameters

(2k generic BBbar events)



— Old model
— New model



RMS (Reco-MC)

Par	New	Old
d0	2.58e-2	2.57e-2
z0	3.18e-2	3.16e-2
omega	4.07e-4	4.05e-4
phi0	1.39e-2	1.39e-2
tanl	1.73e-2	1.73e-2

Implementation in Fastsim

- **New class:** `PacTrkResModelSi`
 - 2 models currently implemented (old and new). Can trivially add more models
 - parameters stored in `vector<double> _pars`
- `PacTrkHitViewSi` **modified to use new class**
- `PacDetector/PacMeasurementFactory.cc` **modified to correctly instantiate `PacTrkHitViewSi` objects**
- **Parameters describing model passed for detector configuration:**
`Si_SuperB.xml`
- `PacEnv/EdmlParser.cc` **modified to recognize new pars**

Conclusions

- Hit resolutions for the pixel detector of Layer 0 cannot be extrapolated from Babar data
- Implemented resolution model for 50x50 μm pixels with digital readout
- Average effect on tracking is very small -- in part because the average hit error in the new model is close to the constant hit error used in the old model
- However, we now have a better description of tracking performance as function of track angle
- Easy to implement new models or update parameters as more information becomes available (via testbeam, etc).