

# Track Fit Algorithm Status

Wendy Taylor  
STT Meeting  
Fermilab  
January 19, 2001

# Matrix Lookup Table Algorithm

$$\phi(r) = \frac{b}{r} + \kappa r + \phi_0, \quad \vec{p} = (b, \kappa, \phi_0)$$

♠ 3 parameters ( $p_i$ )

♠ 2 CFT hits + 4(3) SMT hits ( $r_j, \phi_j$ )

$$p_i = \sum_j M_{ij} \phi_j, \quad M_{ij} = f_i(r_j, \sigma_j)$$

♠ Use 160  $\phi$  slices (SuperRoads) where  $r_j \approx r_{\text{layer}}$  to limit number of matrices  $M_{ij}$

# Matrix Lookup Table Status

- ♠ Floating point version exists for 3x6 matrix (nb: doubles overflow memory)
- ♠ Consists of 12 LUTs (one for each TFC, necessary to minimize LUT size)
- ♠ Small LUT converts  $\varphi_H$ /global Road ID into TFC ID and local Road ID (ignores overlap)



# MInv LUT To Do List

- ♠ Convert to 3x5 matrix (for rotated  $\phi$  vectors, cuts LUT size in half)
- ♠ Create integer version
- ♠ Pack 2 16-bit elements into 32-bit word (cuts LUT size in half)

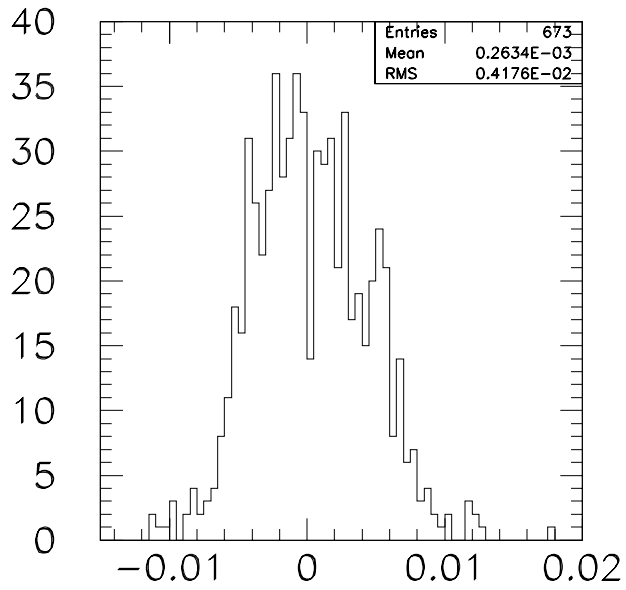
# CFT/SMT Coordinate LUTs

Radius (50 $\mu\text{m}$ )	Barrel	Layer	Hit $\phi$	Field
27..22	21..19	18..16	15..0	Bits

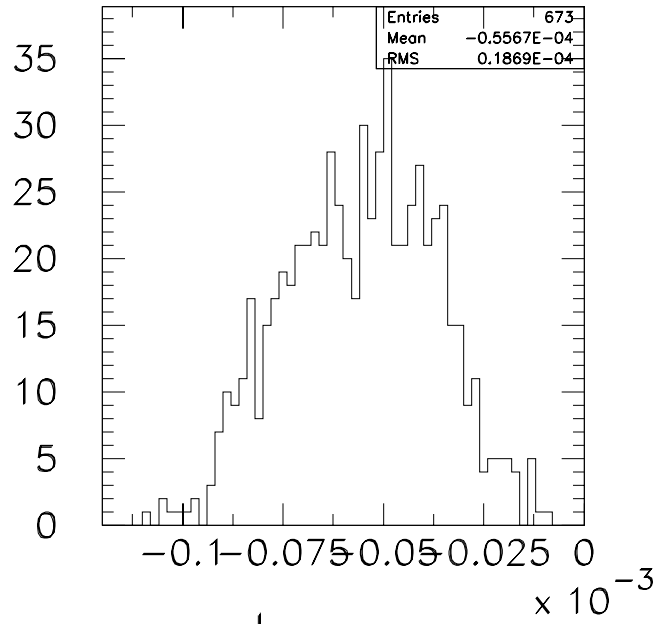
- ♠ Integer quantities now available
- ♠ Store  $\phi$  range per TFC (precision, small LUT provides reference  $\phi$  per TFC)
- ♠ SMT LUT stores residual radius (precision, small LUT yields reference radius per layer)

# Integer Fit Algorithm Status

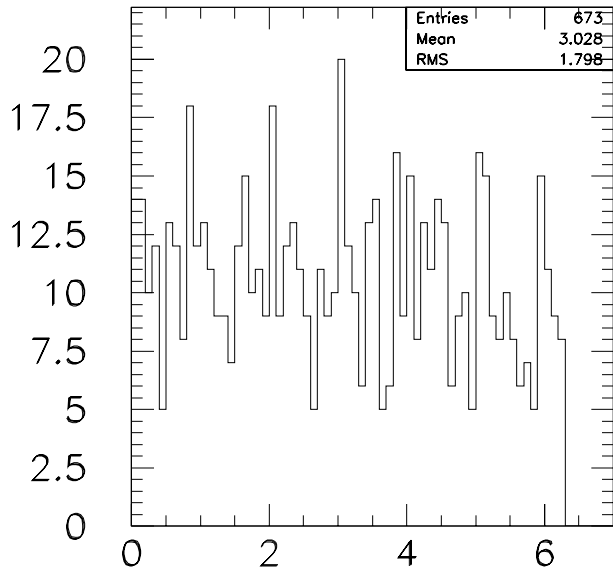
- ♠ Double matrix computed from double coordinates, scaled to integer
- ♠ Hit selection, track fit and  $\chi^2$  calculation performed with 16-bit integers
- ♠ Worst SMT hit is dropped and track is refit with 3 remaining SMT hits
- ♠ Track parameters and  $\chi^2$  agree well with 3xN result



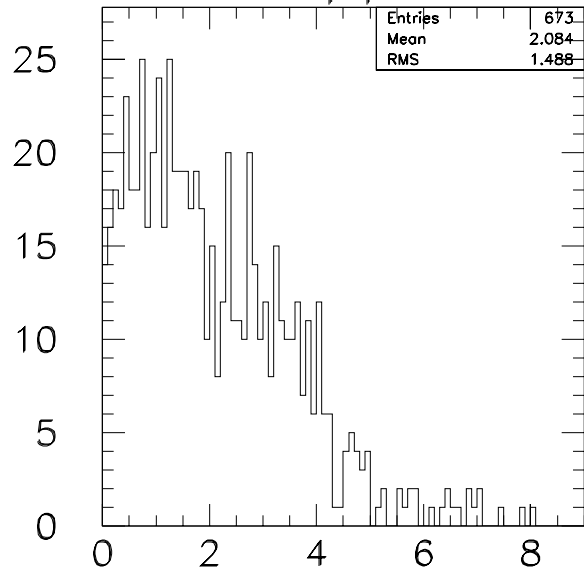
$b$



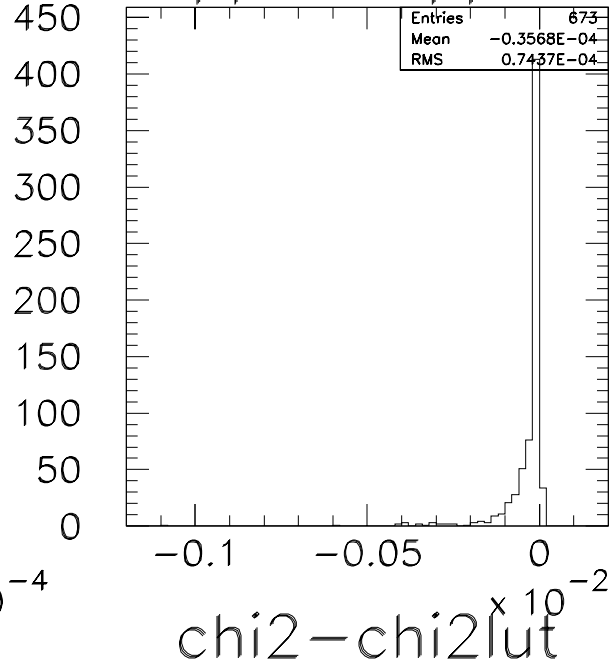
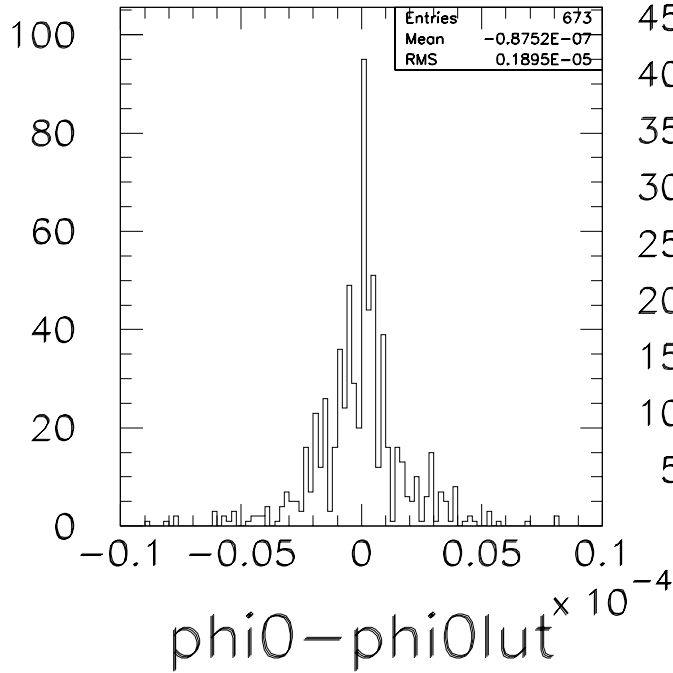
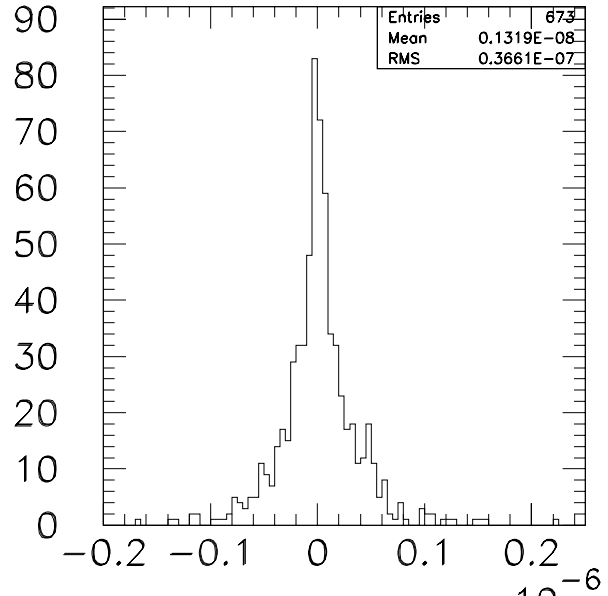
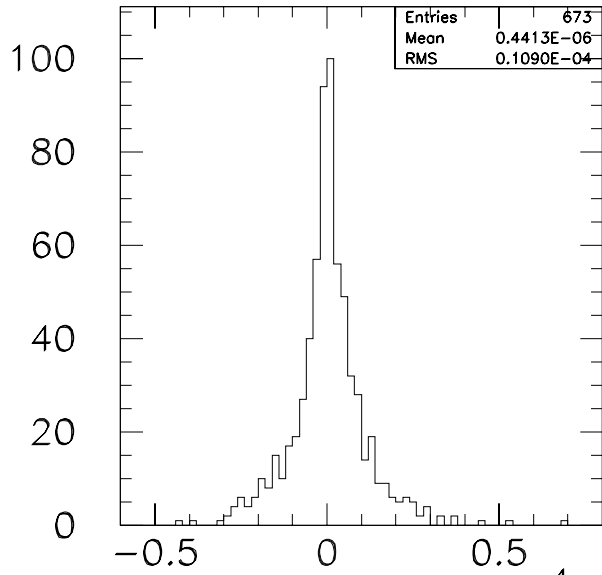
$\kappa$



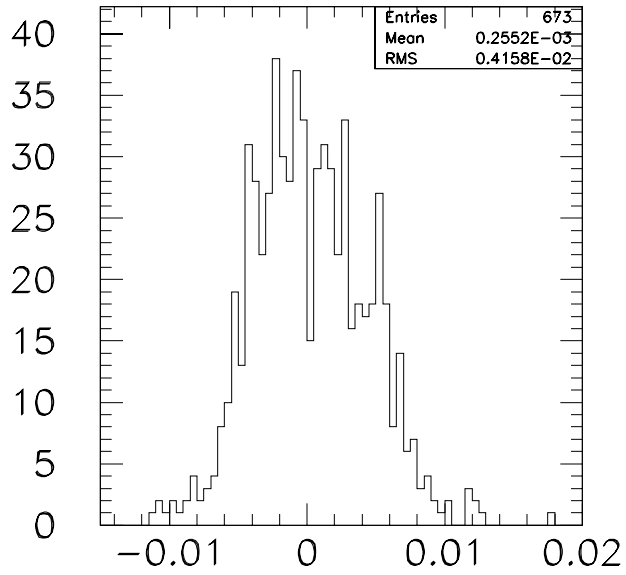
$\phi_0$



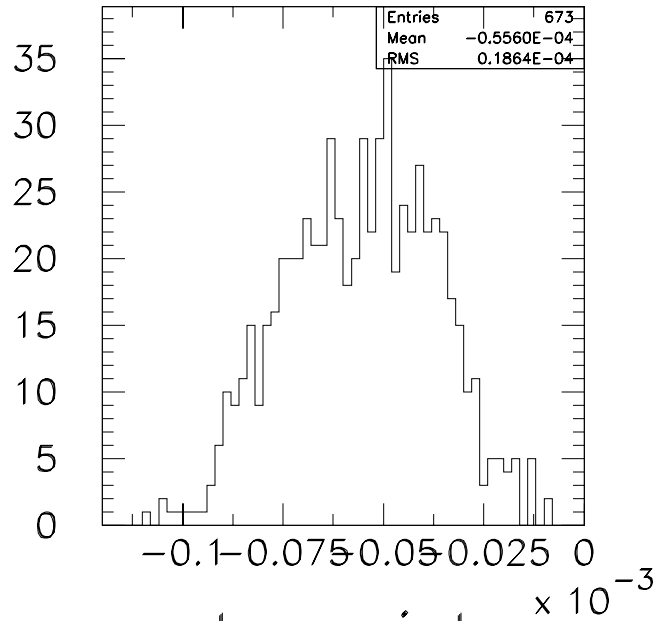
$\chi^2$



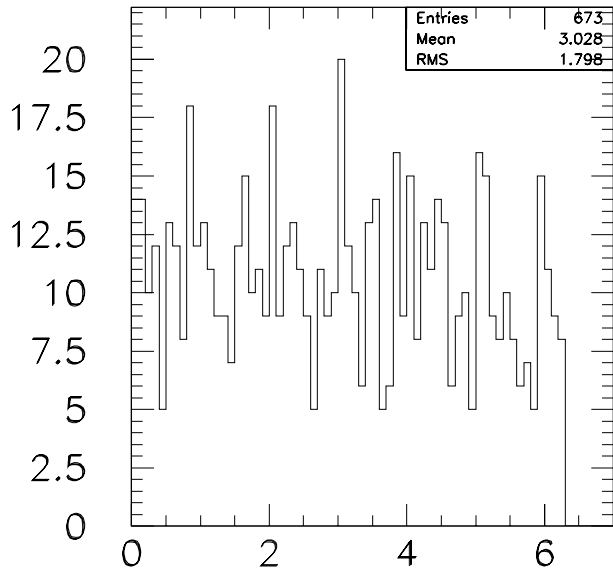




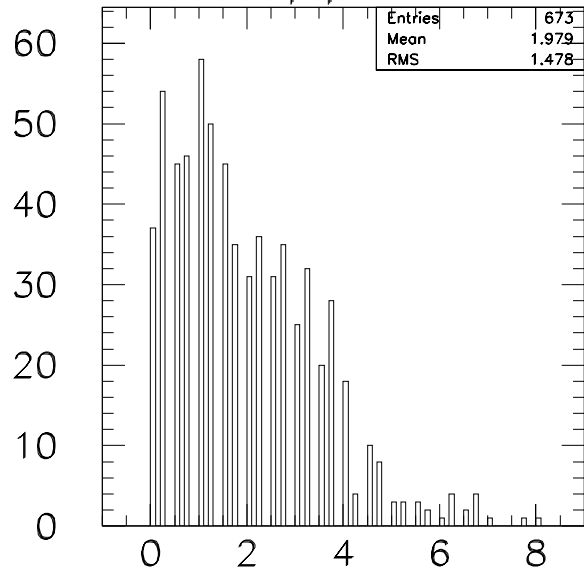
`bint`



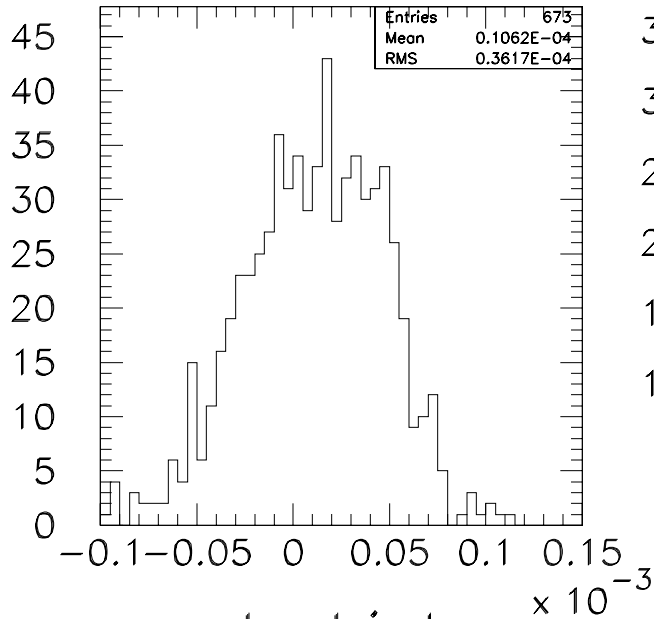
`kappaoint`



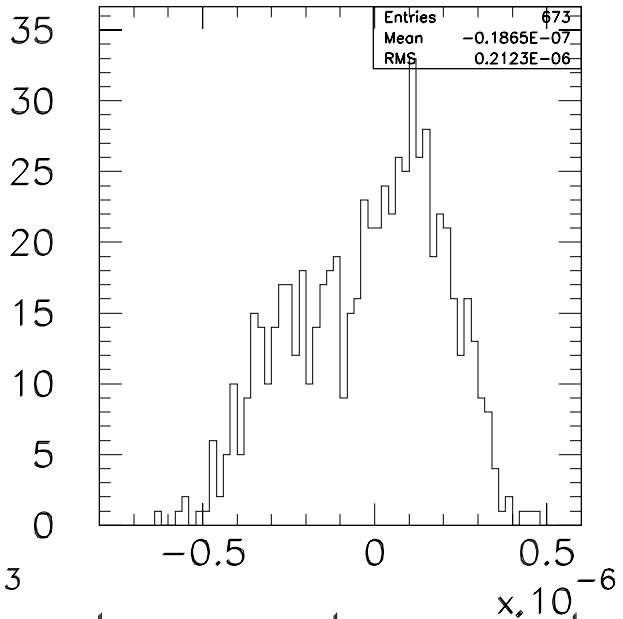
`phi0int`



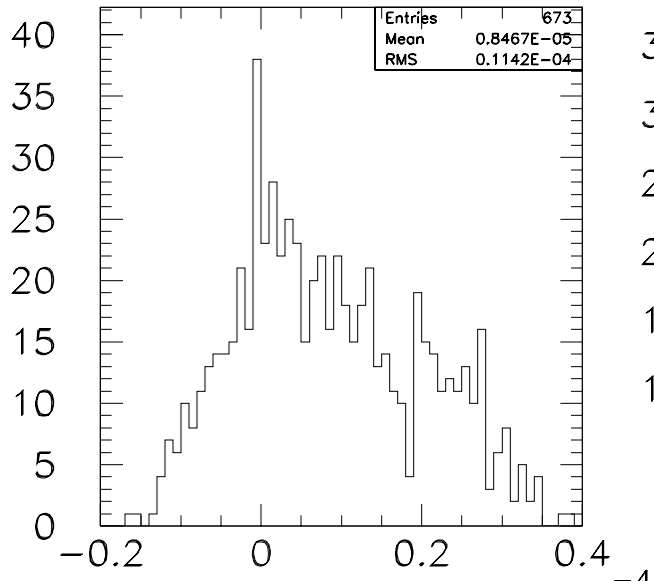
`chi2int`



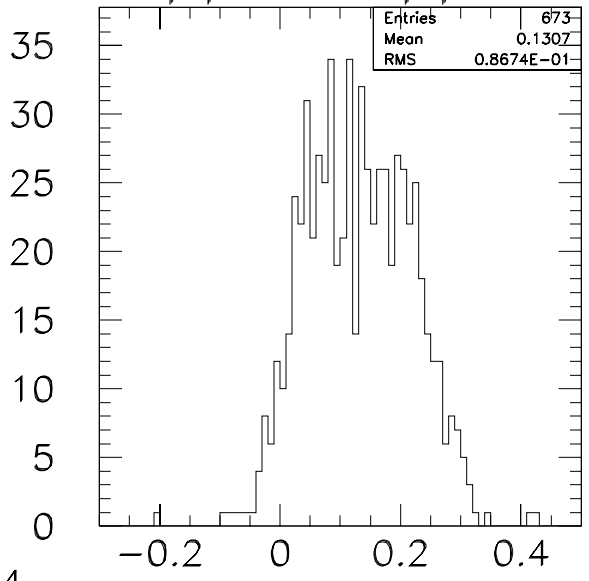
$b - b_{int}$



$\kappa - \kappa_{int}$



$\phi_0 - \phi_{0int}$



$\chi^2 - \chi^2_{int}$

# Integer Fit Algorithm Tasks

- ♠ Use matrix lookup table (integer form)
- ♠ Convert divisions (scaling) to bit-shifts and decide how to compute  $\chi^2/\text{dof}$
- ♠ Output integer quantities (i.e., plots and correct output record format)
- ♠ Compare all results to 3xN algorithm
- ♠ Clean up C++ code for easy maintenance
- ♠ Convert to C for DSP