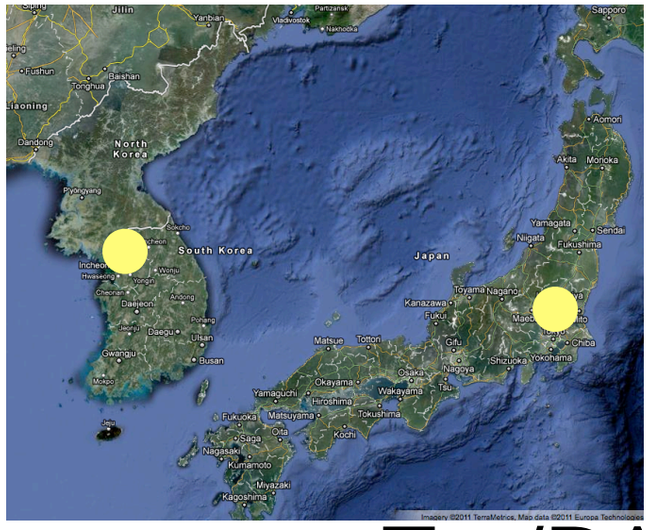
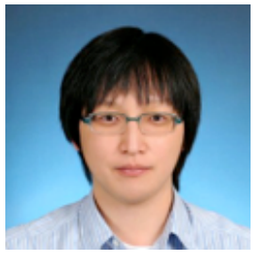
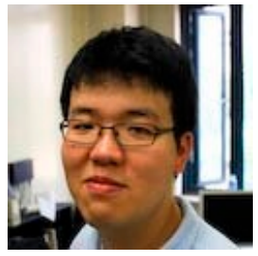
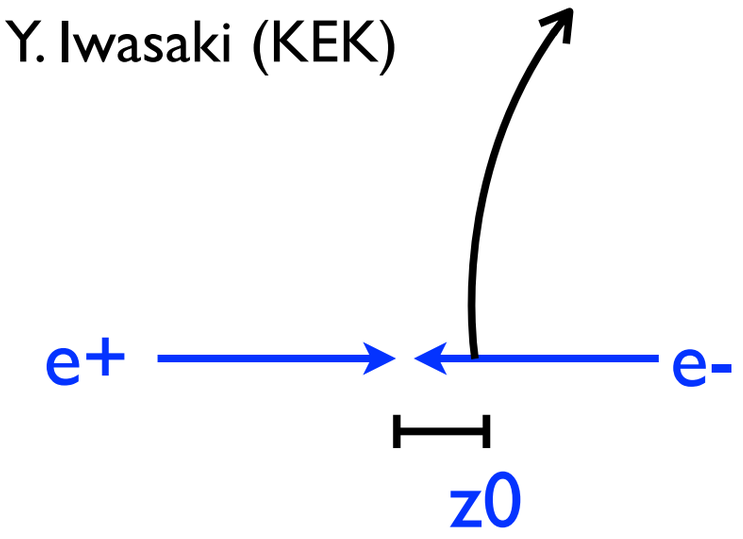
Status of a conventional 3D track trigger

Eunil Won for Jaebak Kim, Kyungtae Kim (KU) and Y. Iwasaki (KEK)









Korea University

Trig/DAQ workshop 2012 1. 16-18 U of Hawaii

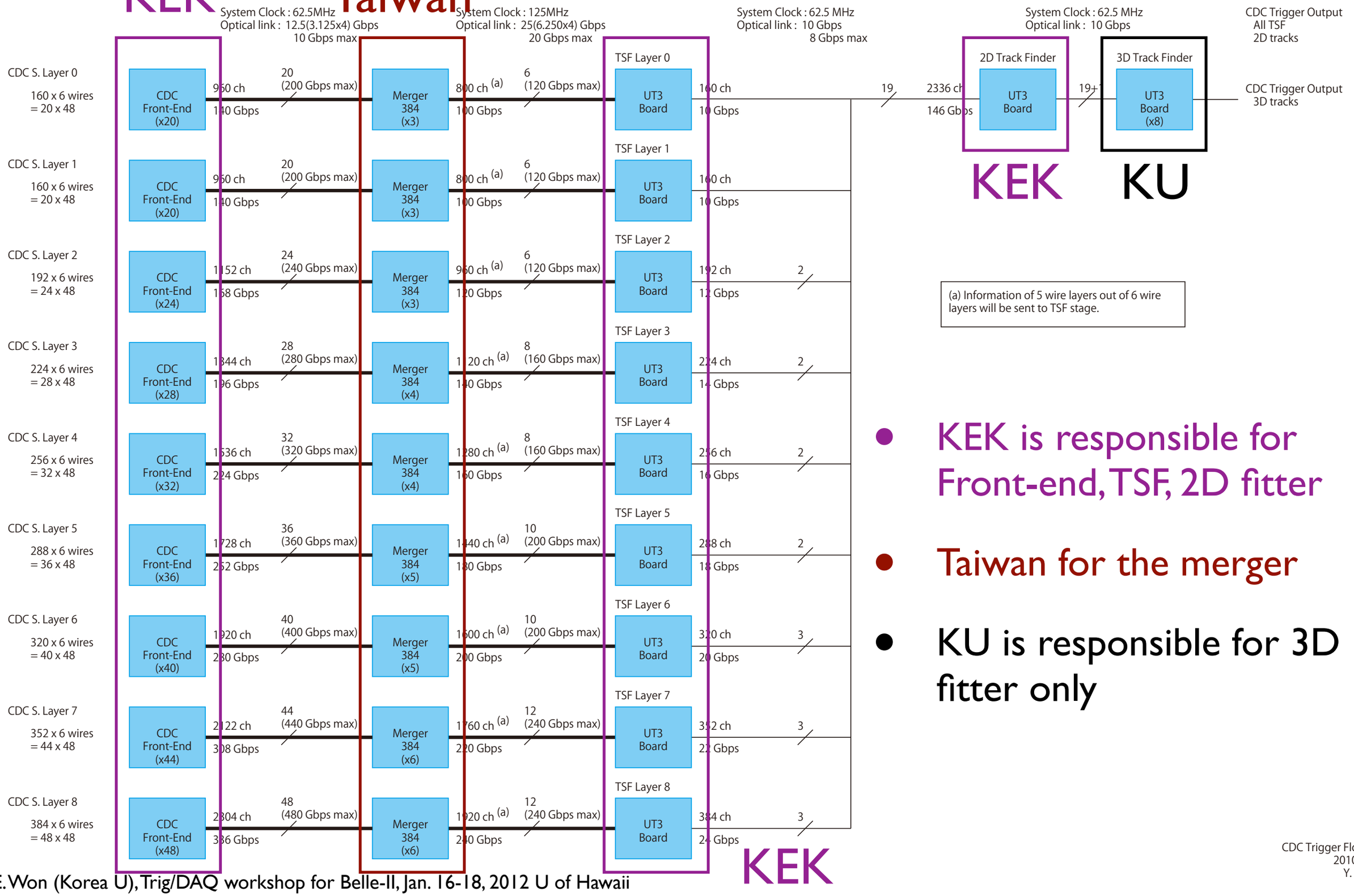
δz0: error on the closest z value of a track from IP

Outline

- Present system of track triggers
- A brief history of developments
- 3D fitter status
- Recent progress and status
- Plan

Track Trigger System

KEK Taiwan



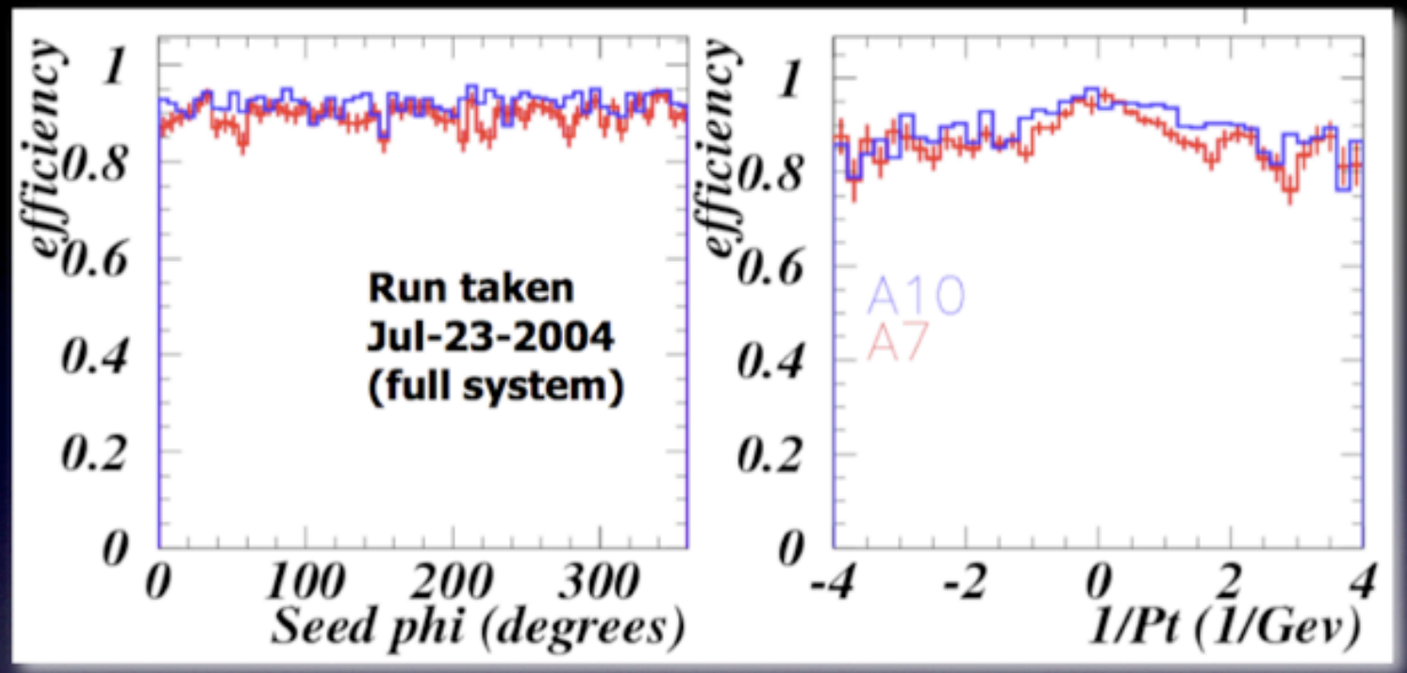
(a) Information of 5 wire layers out of 6 wire layers will be sent to TSF stage.

- KEK is responsible for Front-end, TSF, 2D fitter
- Taiwan for the merger
- KU is responsible for 3D fitter only

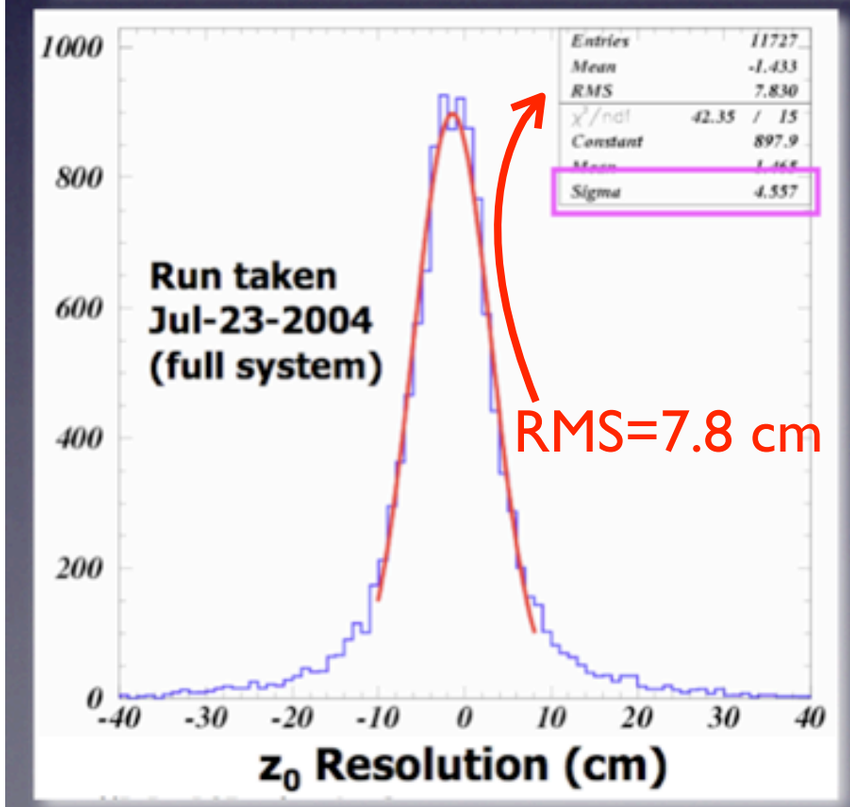
History

- I proposed to have 3D track trigger for Belle-II in 2008 at the level 1 stage
- It was based on my experience with BaBar
  (IEEE NS2, 1288 (2003))
- They achieved $\sigma_{Z0} \sim 4.6$ cm
- This was powerful in rejecting beam-induced background

Performance of BaBar 3D Track Trigger



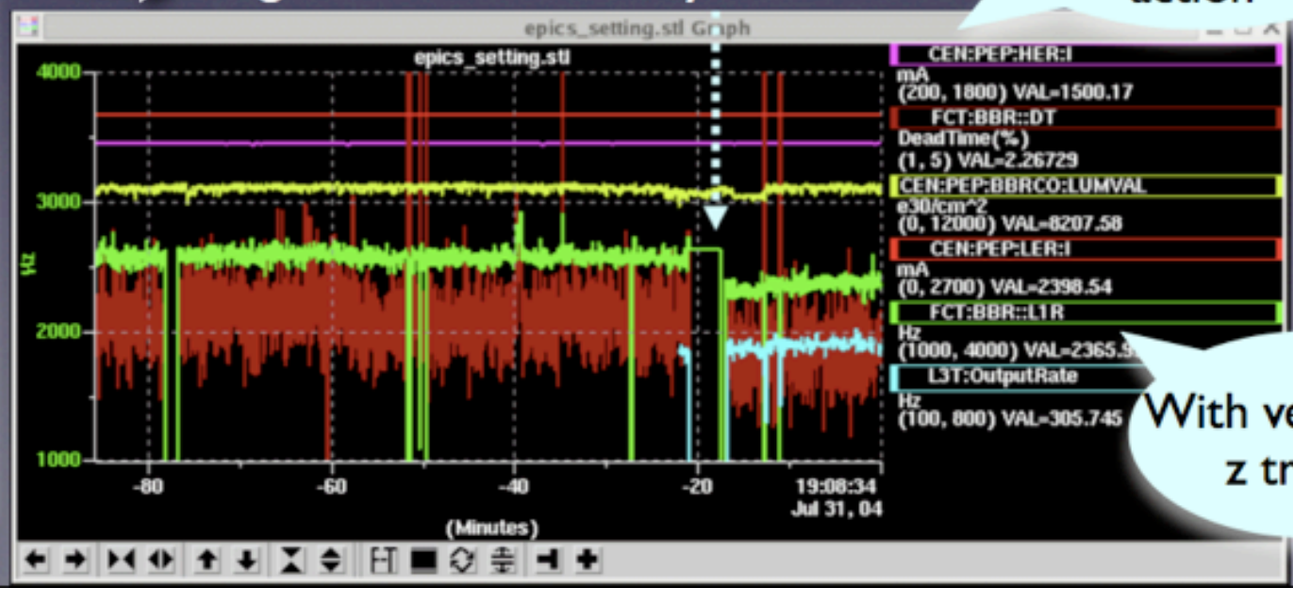
High efficiency (~90%) over full 360° and wide range of pT



The z0 resolution is ~ 4.6 cm and is in good agreement with their simulation

This is precise enough to select signal tracks exclusively

EPICS snapshot at the trigger in action



With very loose z trigger

Belle vs BaBar

- Difference between Belle-II and BaBar

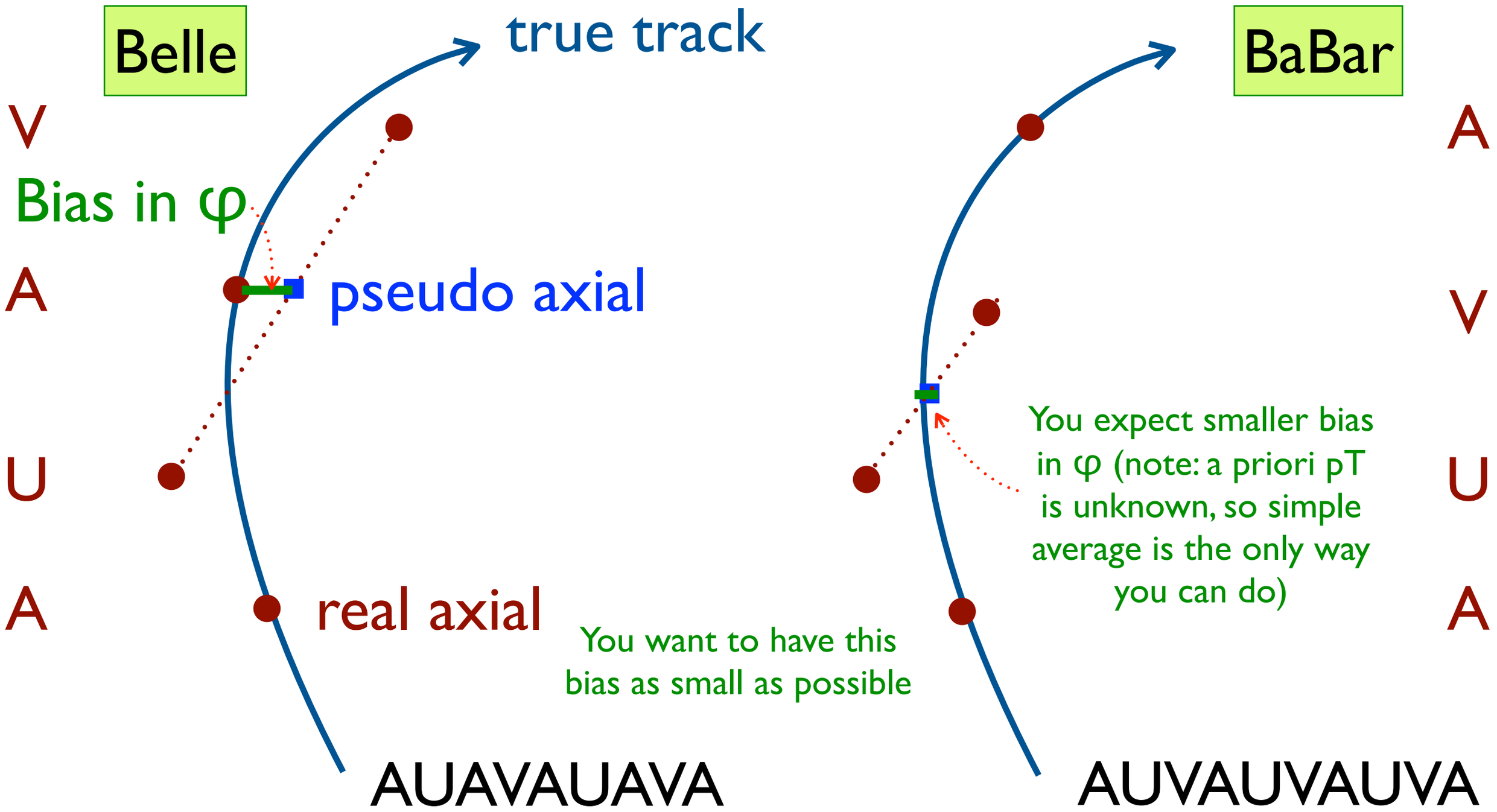
  - CDC geometry

  - Number of axial and stereo super-layers

    AUAVAUAVA : 5 axial and 4 stereo (Belle II)

    AUVAUVAUVA : 4 axial and 6 stereo (BaBar)

Pseudo Axial



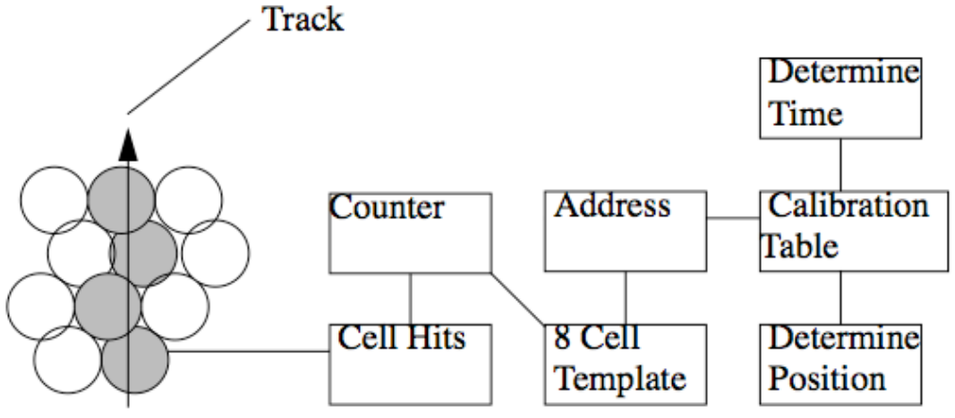
- Study showed AUVA is better in 3D fit

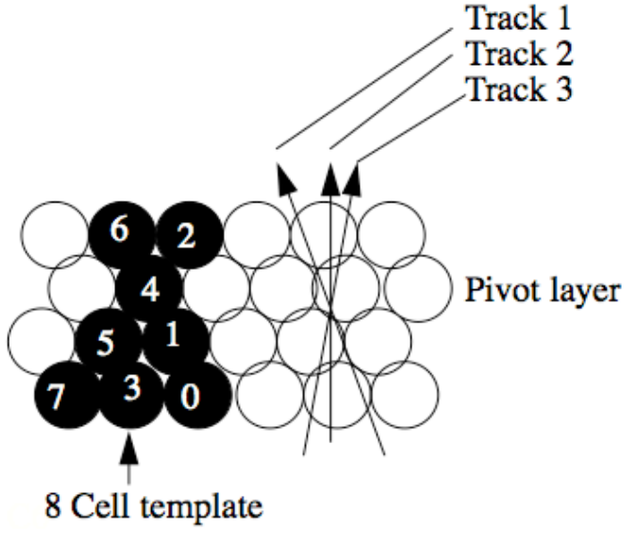
Belle vs BaBar

- Track Segment Finder (TSF)
  - Belle-I : send one bit of TS (the center wire ID)
  - BaBar : send “fine” location of phi

5.2.2 Counter-Based Segment Finder

Here we use two bits per cell to give drift time information for the track, instead of just using one to report whether the cell was hit or not. The address reflects how much time has passed since the wire was hit. By incrementing a counter every 269 ns (64 beam crossings) after a hit, we can thus obtain an approximation of the drift time. This is shown on the lower right hand corner of Figure 5.4. From the drift time and the relationship shown in Figure 2.5, we can thus obtain a better axial (ϕ) position resolution, as well as a better time-of-event resolution than in the one-shot case. This approach produces 65536 addresses instead of 256. There are 62721 valid 3/4 segment patterns and 41715 valid 4/4 segment patterns. Of those 4/4 patterns, 367 report a calibrated address as defined in the next section.



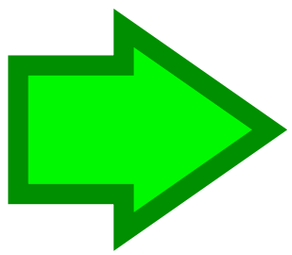


Cell/T	T0	1	2	3	4	T0	1	2	3	4	T0	1	2	3	4
6	0	0	0	**1**	2	0	1	2	**3**	0	0	0	0	**1**	2
4	0	0	0	**1**	2	0	0	0	**1**	2	0	0	1	**2**	3
5	0	0	0	**1**	2	0	0	0	**1**	2	0	0	1	**2**	3
3	0	0	0	**1**	2	0	1	2	**3**	0	0	0	0	**1**	2
	Track 1					Track 2					Track 3				

There's some more interesting items that I left out...

We asked...

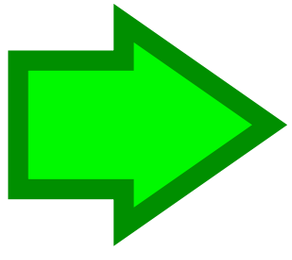
- Re-design CDC with AUVA for better z0 resolution



Uno said NO

But he changed the stereo angle of innermost wires to be larger

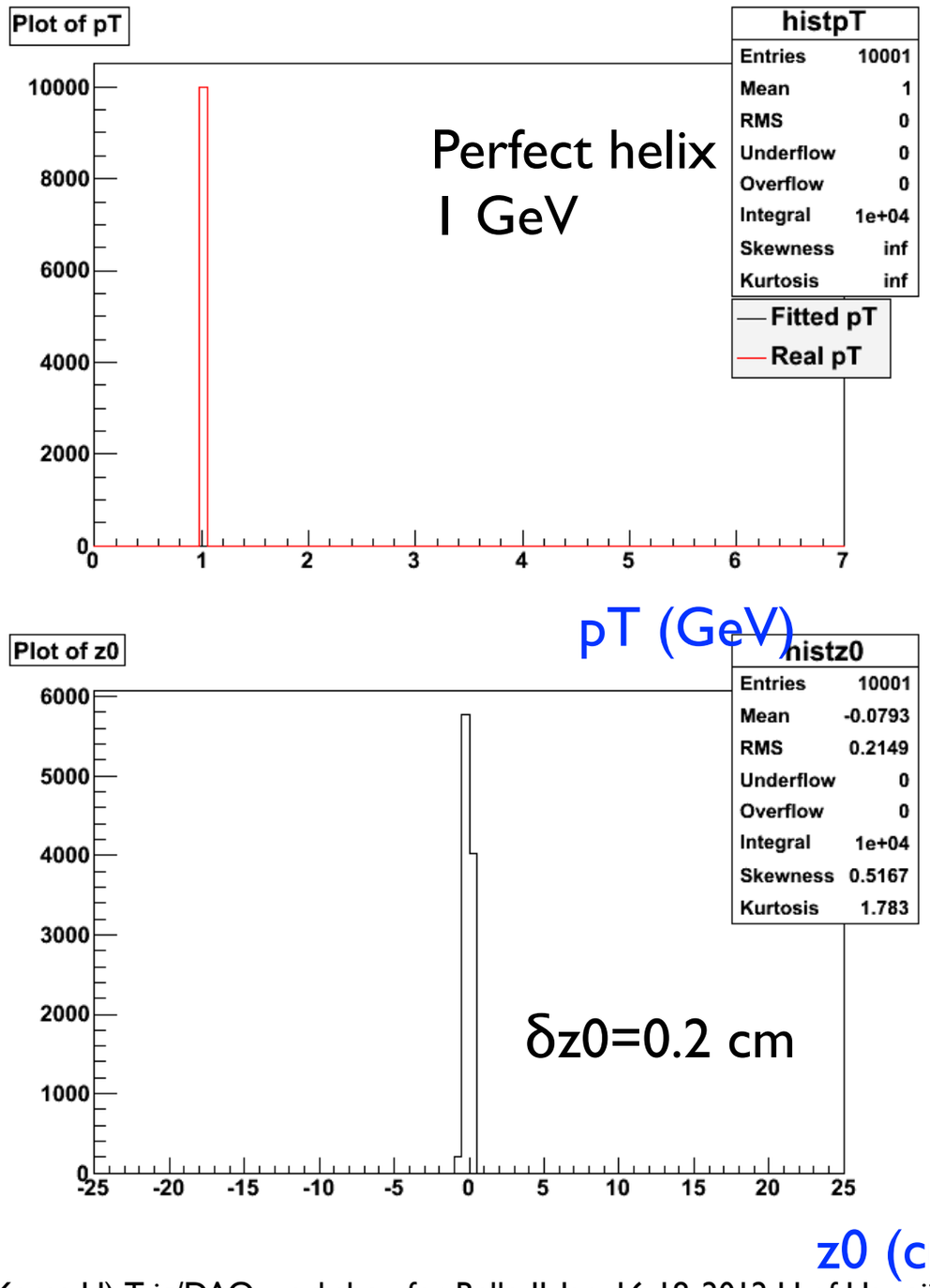
- Improved information from TSF



Y. Iwasaki promised to implement something

3D Fitter Status

- Standalone simulation (w/o basf2)

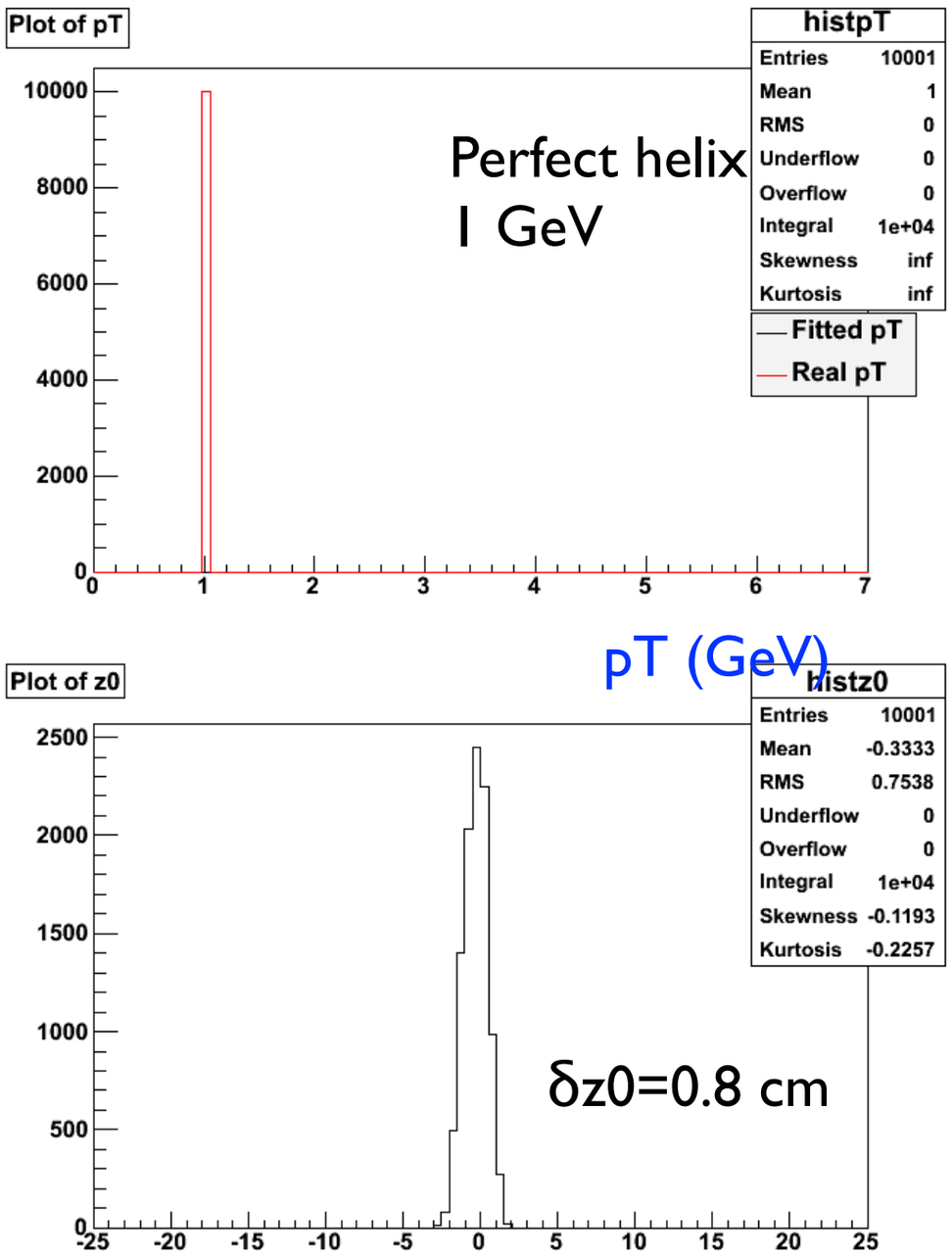


- Linear regression for the fit
- C++ w/ floating point numbers
- LUTs for sin()/cos() etc
- r-z fit approximated as a linear fit

z0 resolution: 0.2 cm

3D Fitter Status

- Standalone simulation (w/o basf2): real 2 integer

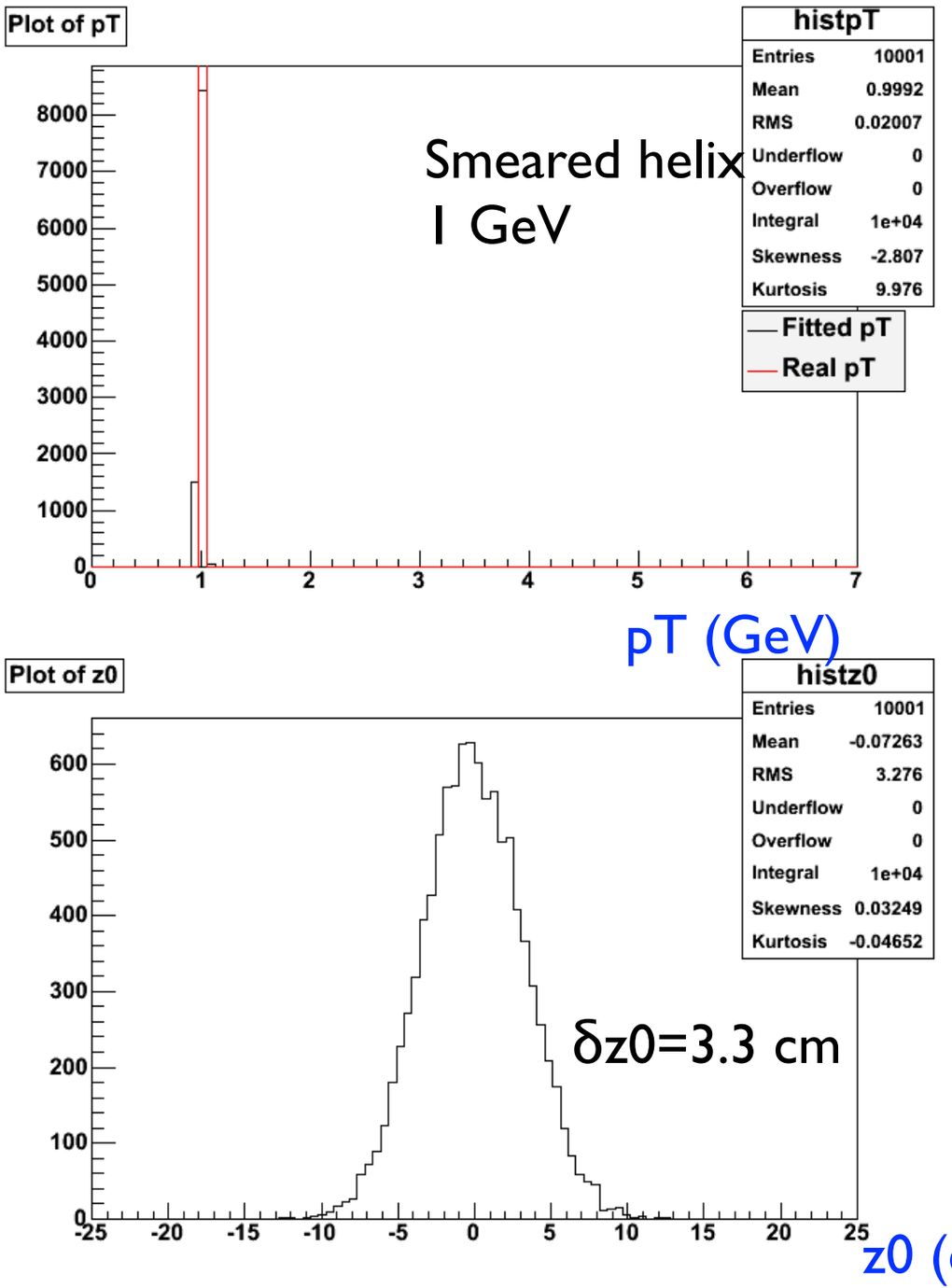


- If you move to integer space (for vhdl)

z0 resolution: 0.2 cm → 0.8 cm

3D Fitter Status

- Standalone simulation (w/o basf2) + smearing



- If you smear your data

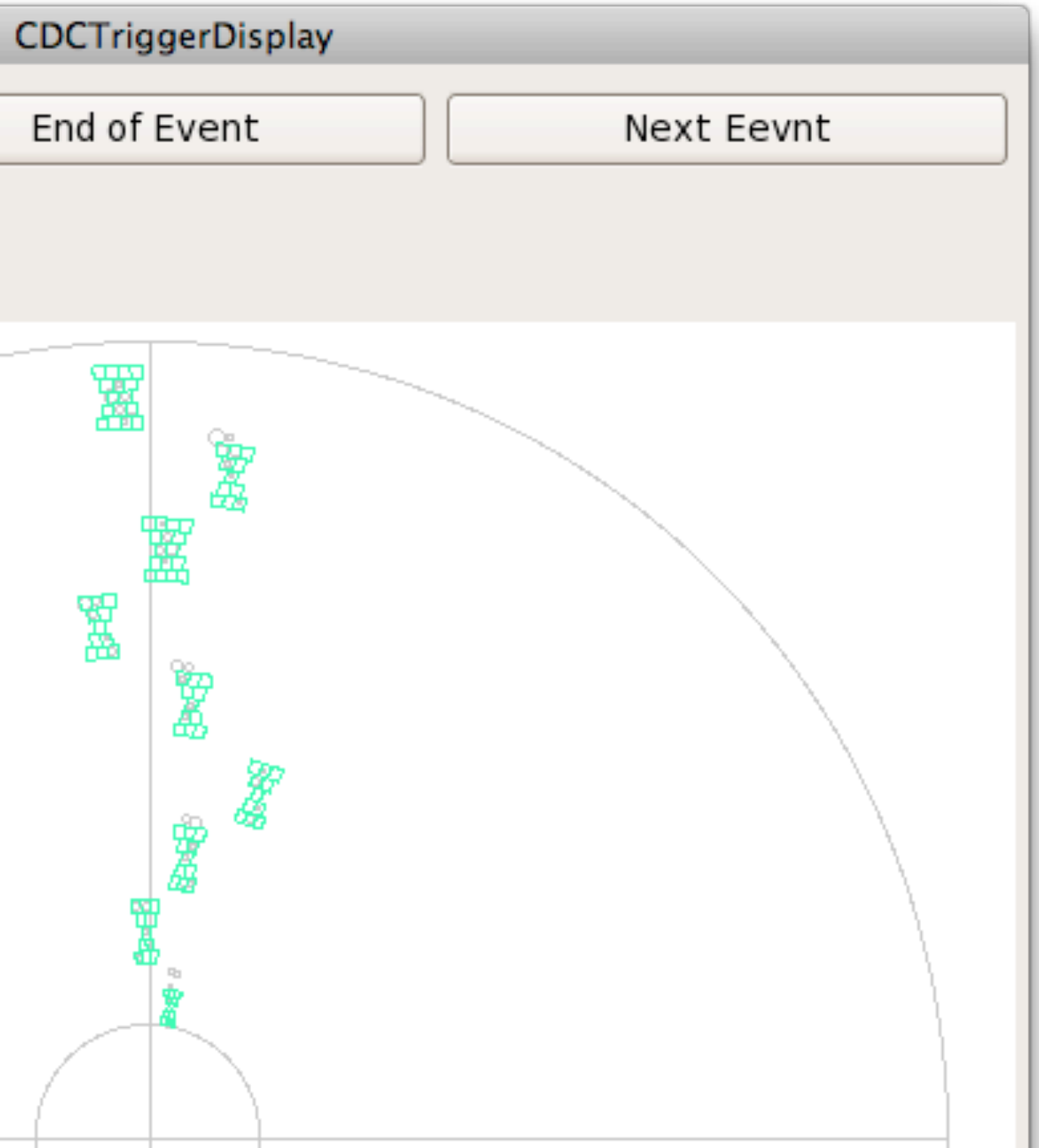
z0 resolution: 0.2 cm → 3.3 cm

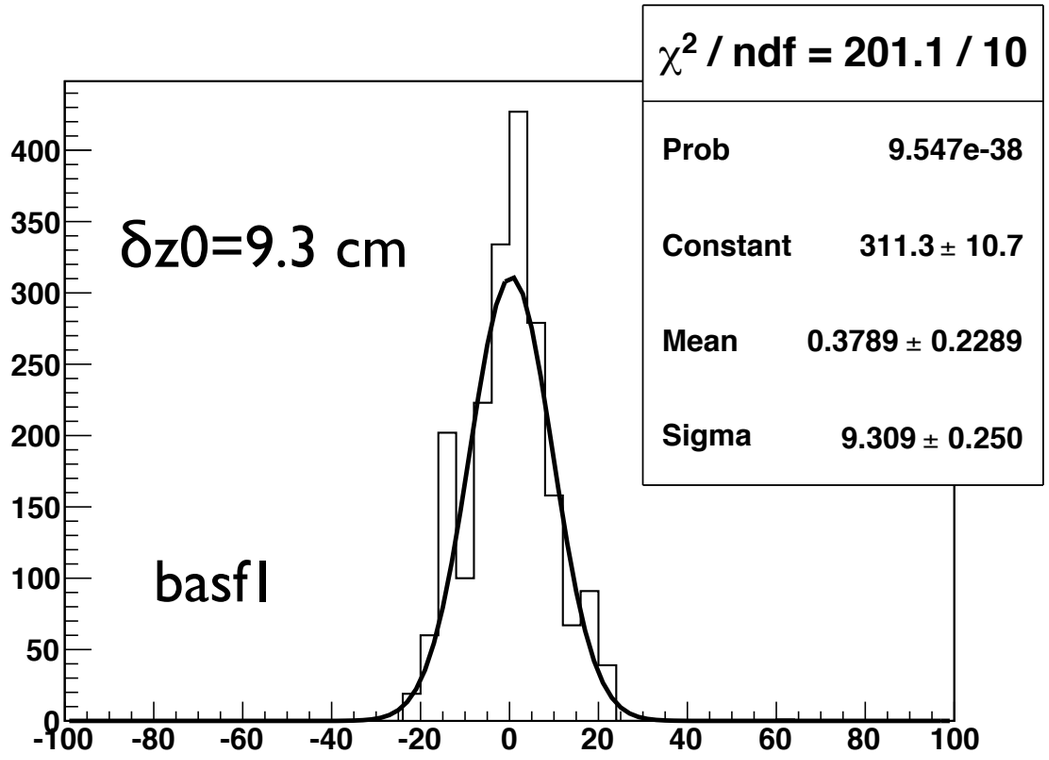
At present, we have a working 3D fitter in vhdl/C++ (in tsim)

(Basically KU task is done...)

3D Fitter Status

- With TSIM data: center wire position of TS used





z0 resolution: ~ O(10) cm at the moment

We need to use timing information from TS level to improve δz0 ➔

New algorithm for TSF firmware needed

Summary of 3D Fitter Status

- VHDL for the 3D fitter exists (need further tuning)
- A new TSF algorithm utilizing timing information of wire hits is needed for better δz0

For better TS location

- We learned that having precise TS location is the most important component for smallest δz0.

- Present strategy : two choices

  1) Compute the fastest time and compute TS location more precisely
     - seems easier
     - Preliminary study of timing resolution (TSIM) seems promising

  2) Do something similar to what BaBar did
     - somewhat complicated (I did not fully understood)
     - will try if 1) fails

Recent progress

Y. Iwasaki in KU

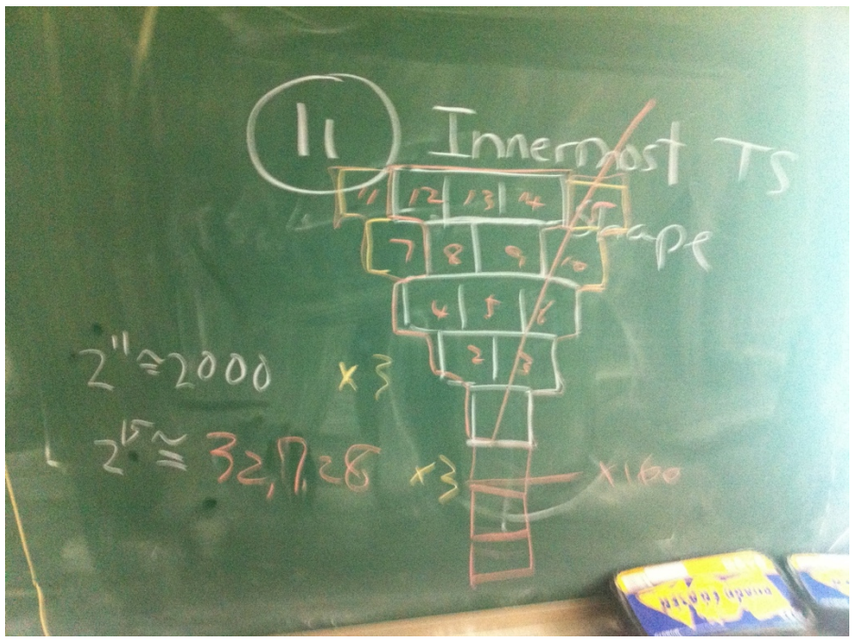
- Y. Iwasaki visited KU (Dec. 18-22, 2011)
  - We discussed various issues for 3 full days

Who does what
1. TSF work: time, LR ambiguity, LUT
2. Front-end: "fine-phi" if needed
3. 2D finder event timing
4. 2D finder/fitter implementation to vhdl
5. Matching stereo TS to 2D track
6. Bias in 3D fit
7. Communication w/ VME cpu
8. 2D finder for multi-tracks
9. TSIM for front-end

(There were various issues I was ignorant deliberately but this time I actually discussed with YI)

Color coding:
Blue: undecided
Black: KU (new commitments)
Brown: Taiwan (TBC)
Purple: KEK

Y. Iwasaki in KU



- Decision on TS for innermost super-layer

- Decision on outputs (subject to change)

TSF
4: time
14: ID
2: LR

2D fitter
8: pT
11: z(i)
4: δz(i)
1: charge
8: time

3D fitter
8: pT
11: φ0
4: z0
1: δz0
8: pz

Equations

- By using Taylor series expansion,

$$z = z_0 + 2\frac{P_z}{P_T}Rsin^{-1}(\frac{r}{2R})$$

We have been ignoring this term

$$= z_0 + 2\frac{P_z}{P_T}R[(\frac{r}{2R}) + \frac{1}{6}(\frac{r}{2R})^3 + \frac{7}{30}(\frac{r}{2R})^5 + \dots]$$

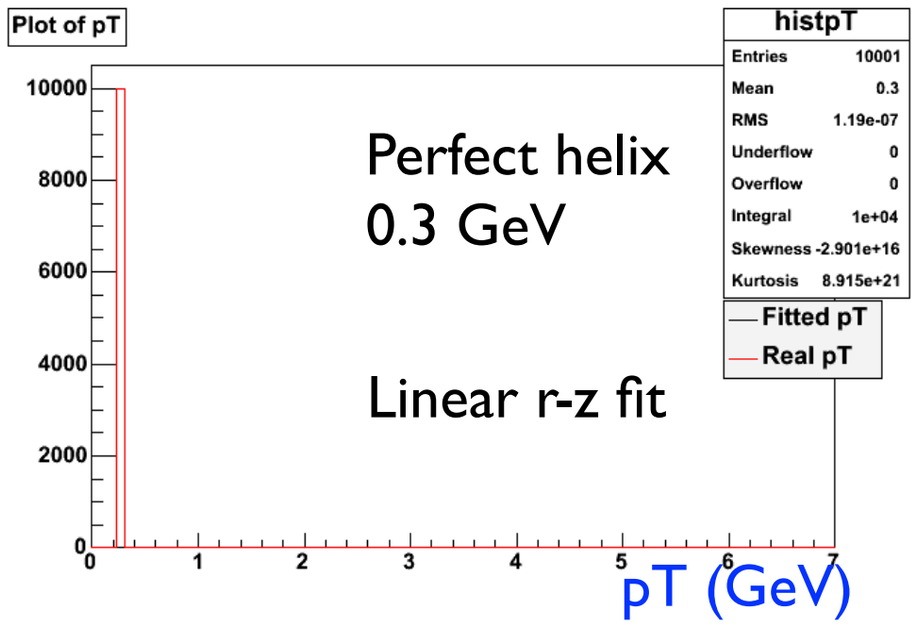
- There is such a relation between R and P_T

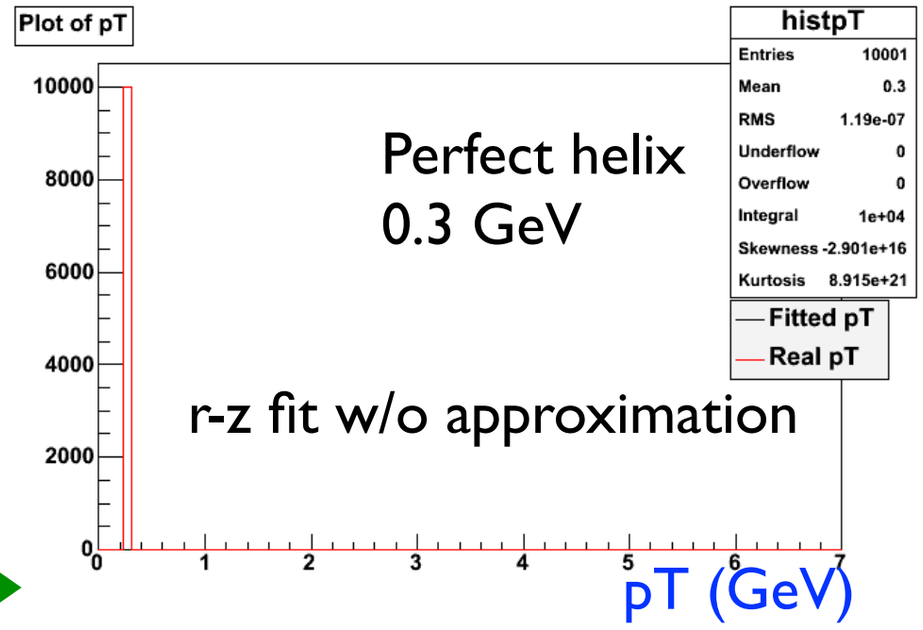
$$R = \frac{P_T}{Bq}$$

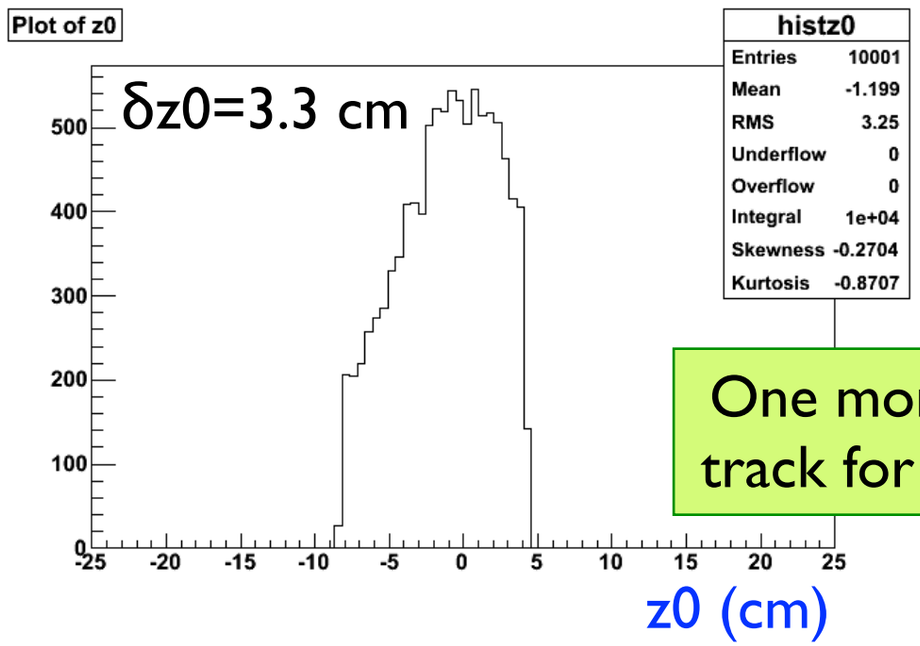
Jaebak Kim

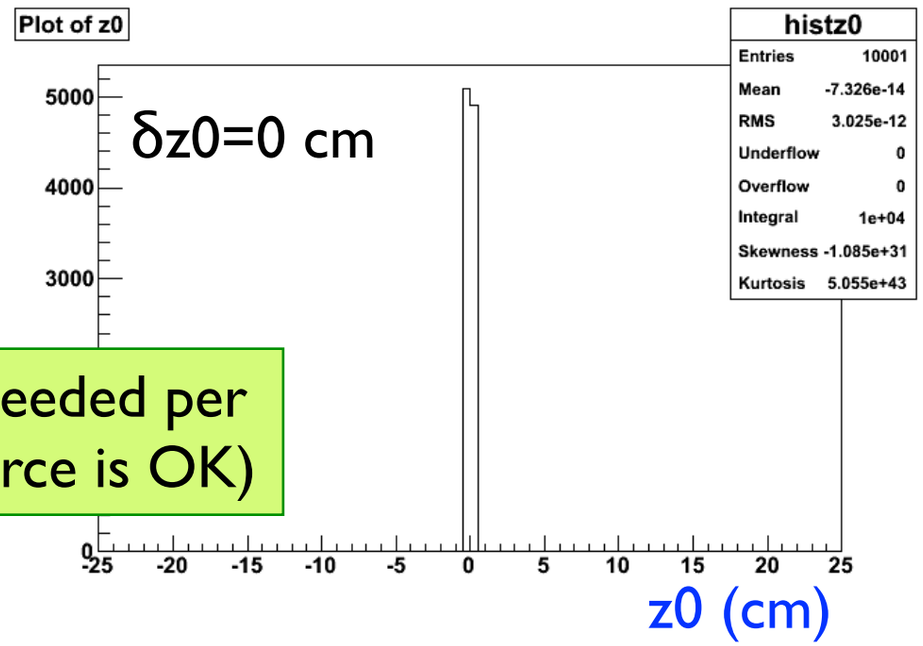
Fitter improvements

- r-z fit : removing approximation of line fit
- Standalone simulation (w/o basf2)







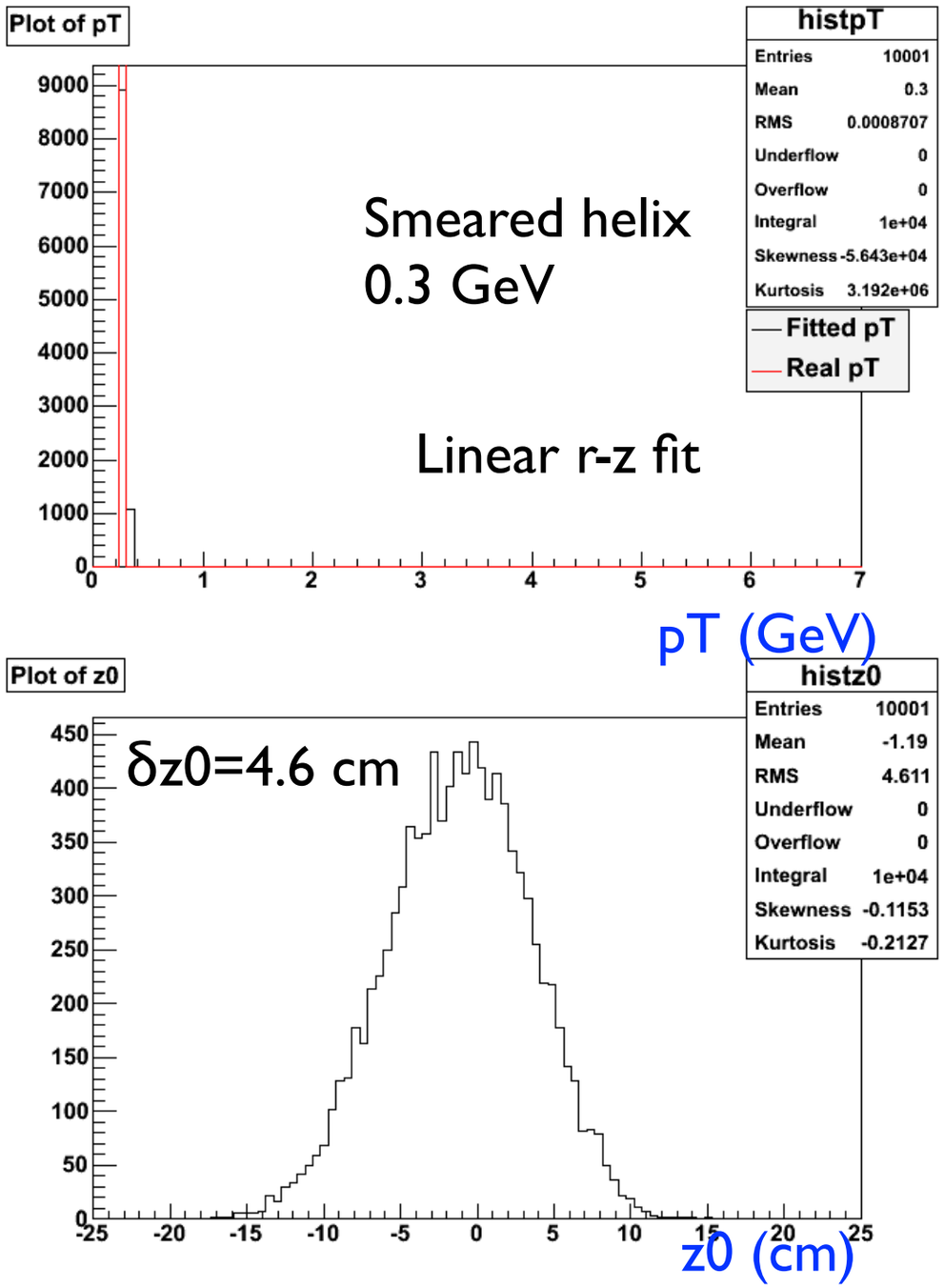


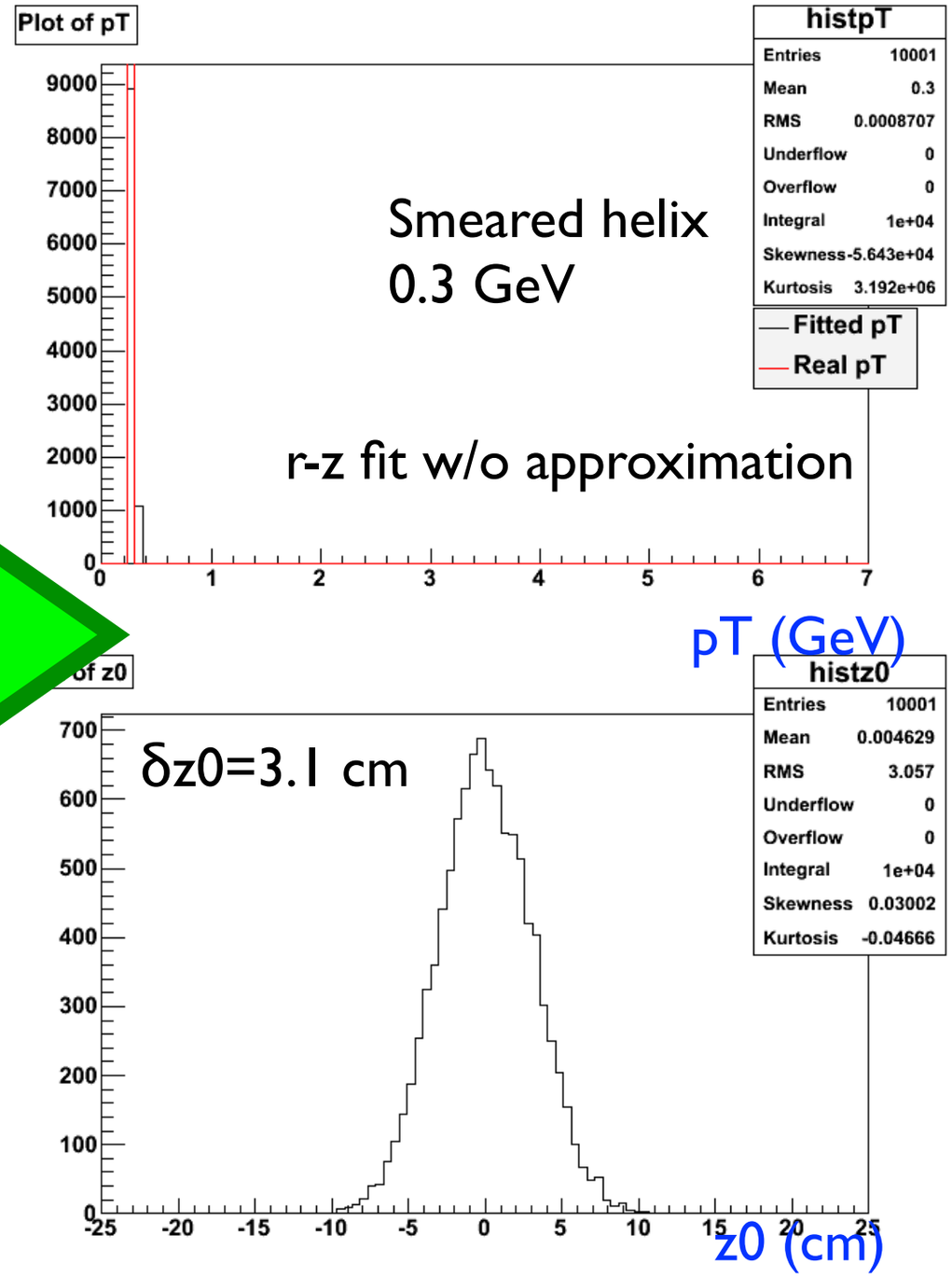
One more LUT is needed per track for this (resource is OK)

Jaebak Kim

Fitter improvements

- Standalone, smeared simulation (w/o basf2)

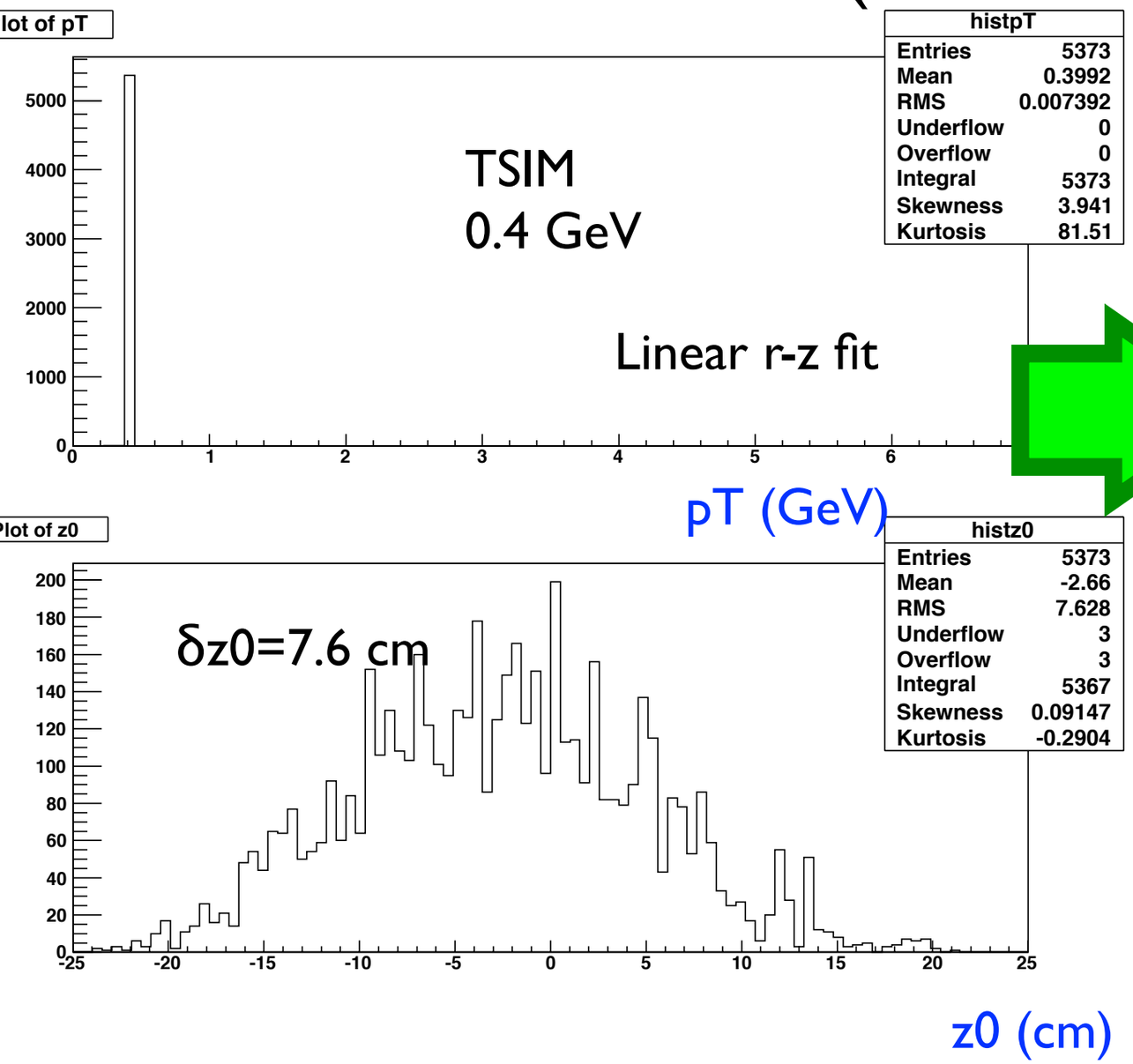


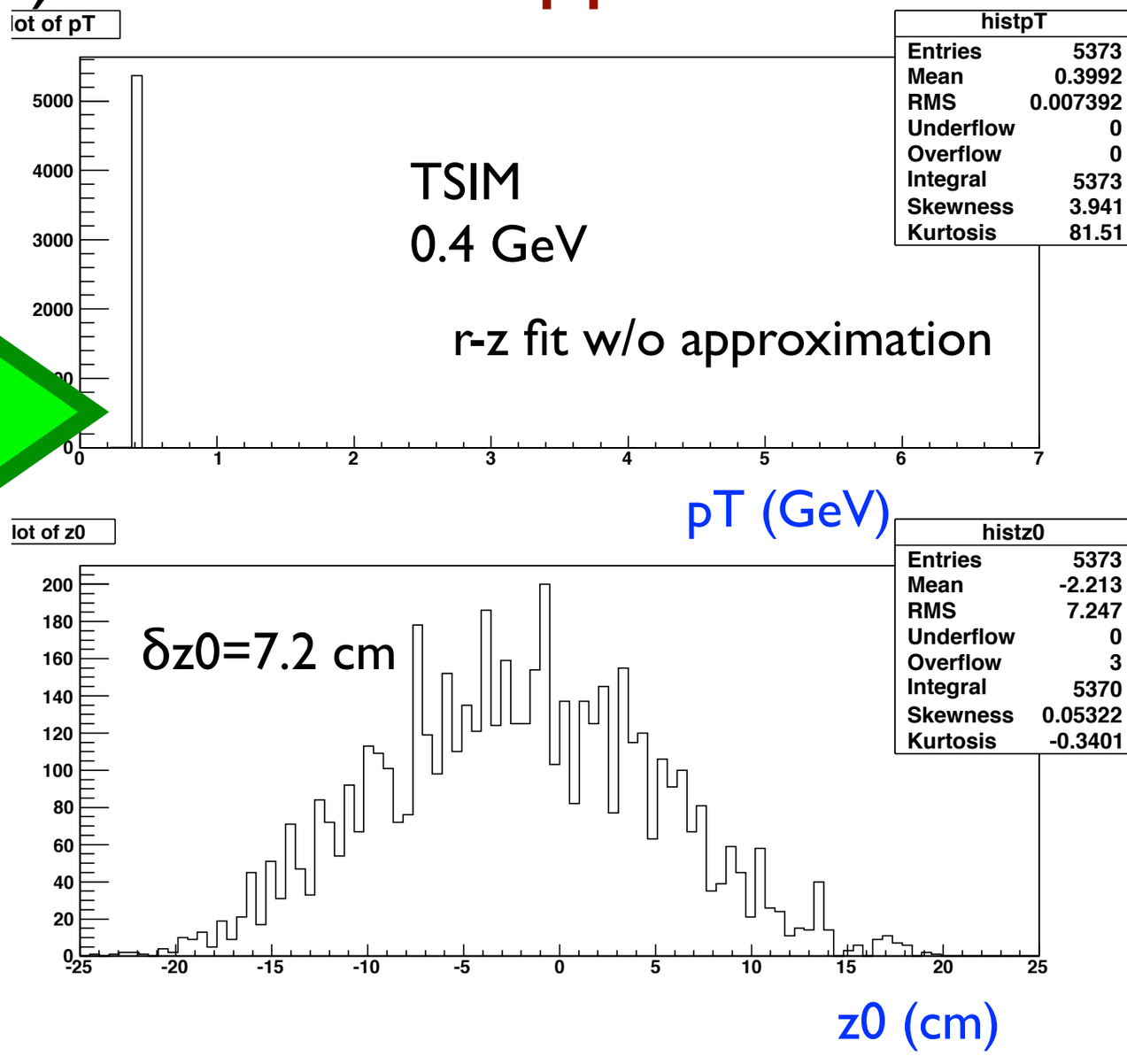


Jaebak Kim

Fitter improvements(?)

- TSIM simulation (w/ basf2): r-z fit w/o approximation



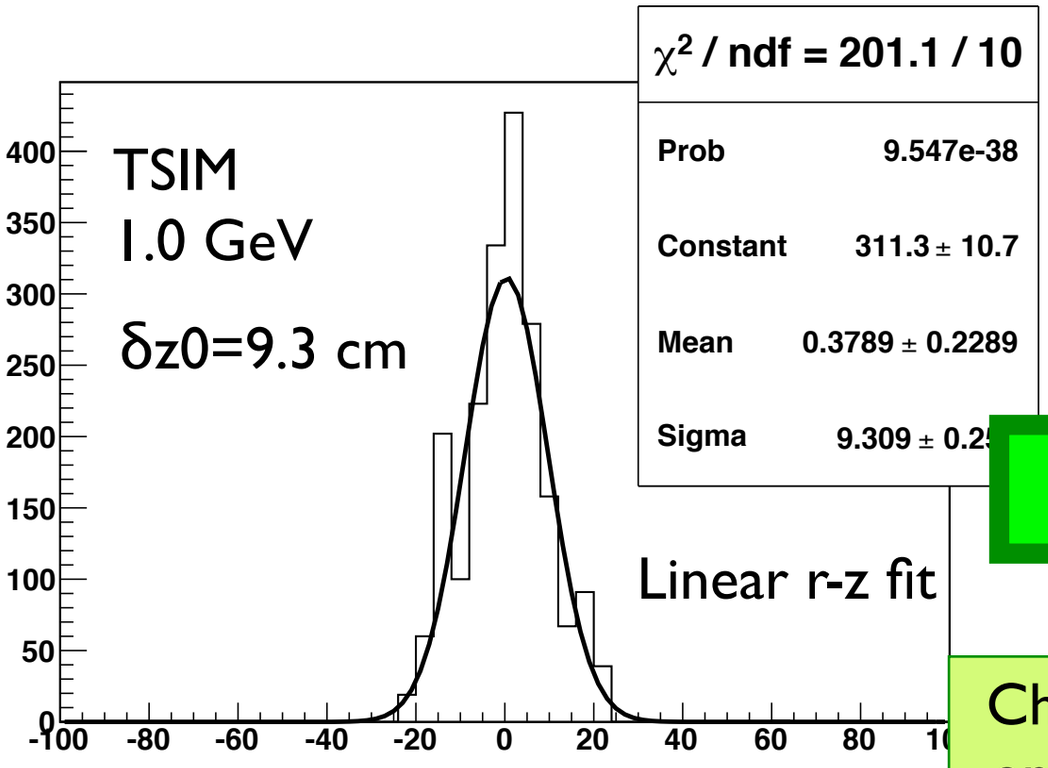


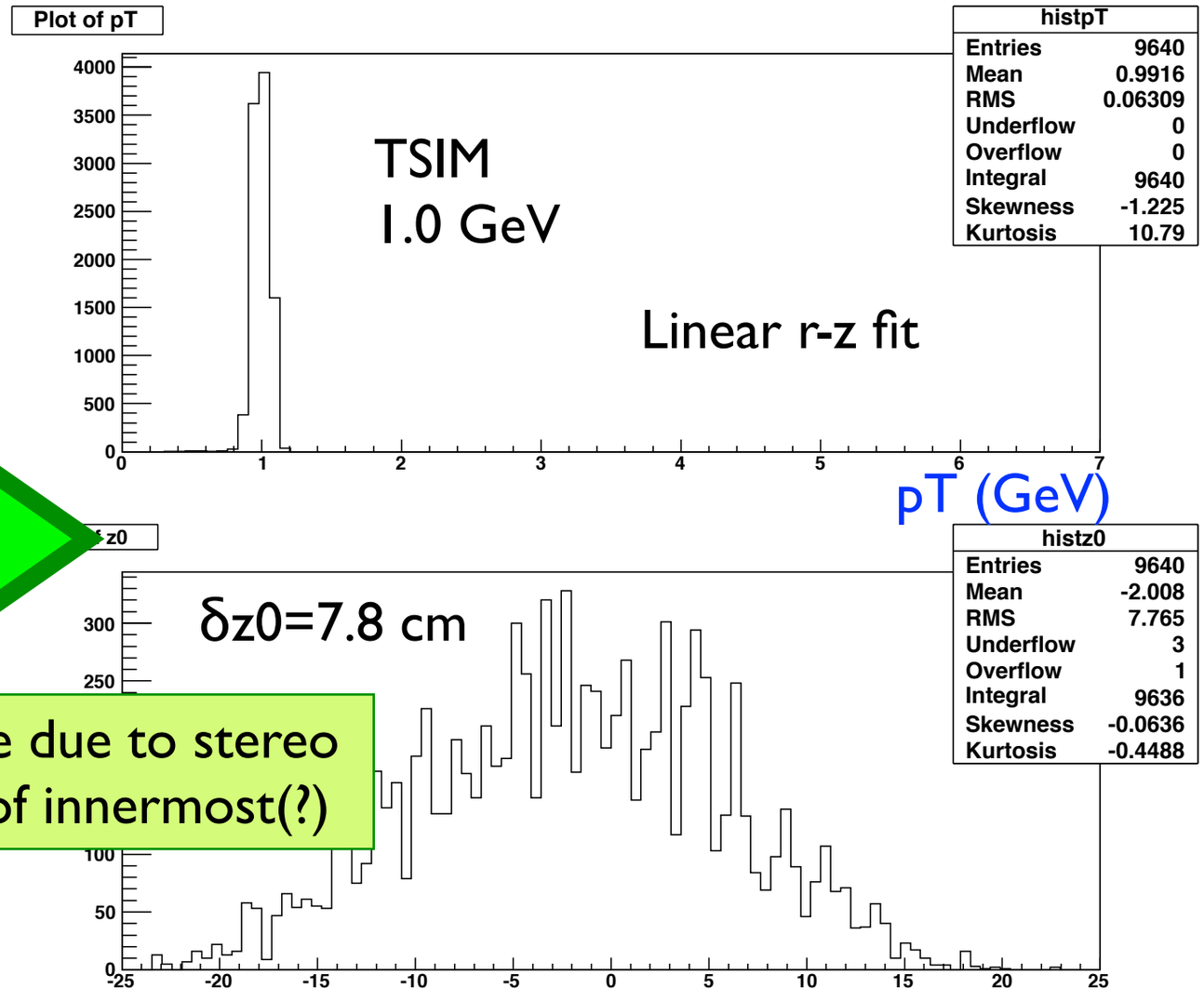
- Some change between with and without r-z linear fit approximation
- We are not sure about the source of ~ -2 cm bias in z0

Jaebak Kim

Fitter improvements(?)

- TSIM simulation (w/ basf2): 19% improvement





Change due to stereo angle of innermost(?)

There are several changes between two..

- stereo angle change for innermost
- basf to basf2 (geometry change)
- algorithm changes

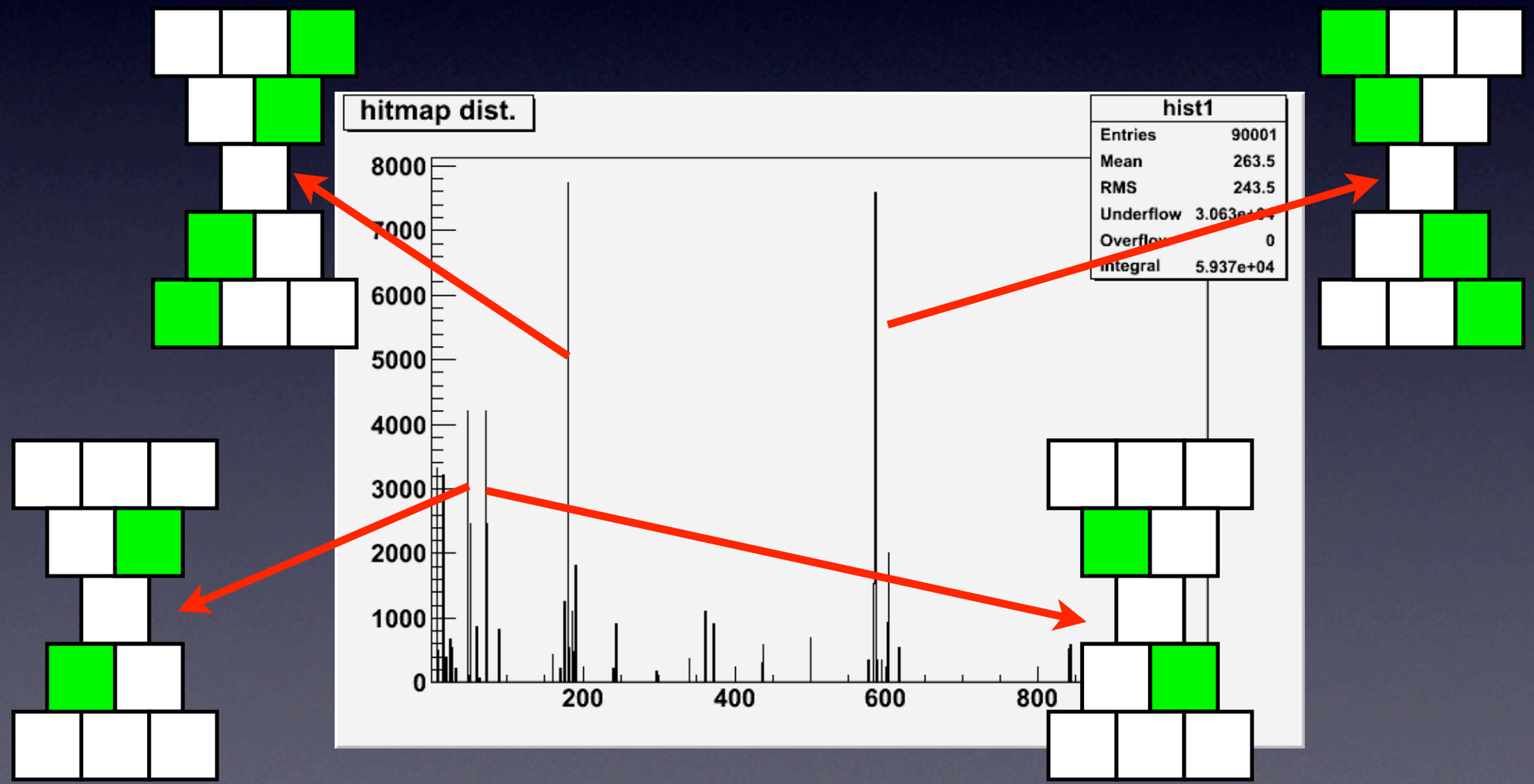
Jaebak's old study expected 18% improvement for 5 holes.

http://ekpbelle2.physik.uni-karlsruhe.de/~twiki/pub/Detector/Trigger/JaeBakSlides3D/Study_on_z0_RMS_by_changing_stereo_angle.pdf

HitMap Distribution

LR ambiguity issue: work in progress

- For low pt, there are two peaks.



Plan on this winter

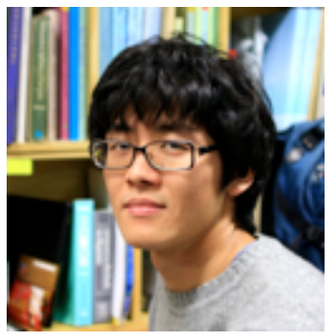
Who does what
1. TSF work: time, LR ambiguity, LUT
2. Front-end: "fine-phi" if needed
3. 2D finder event timing
4. 2D finder/fitter implementation to vhdl
5. Matching stereo TS to 2D track
6. Bias in 3D fit
7. Communication w/ VME cpu
8. 2D finde for multi-tracks
9. TSIM for front-end

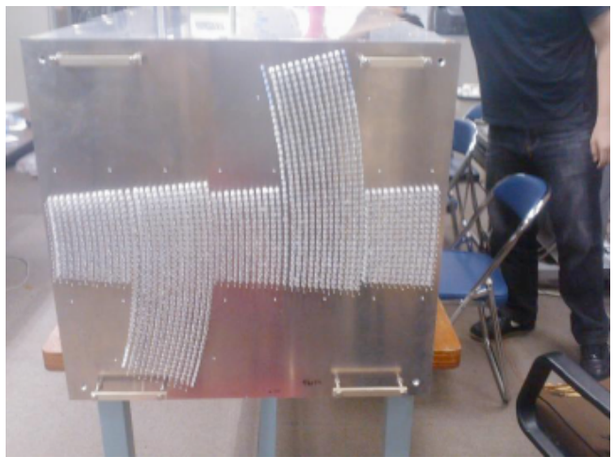
- Jaebak Kim and Kyungtae Kim in KEK to do some of these.

(This is a new commitment by KU)

- 3D test chamber
  - Hyunki Moon and Donghyun Kim will be in KEK from 25 Jan.
  - Finish the cabling (almost done)
  - Timing measurement (important for TSF algorithm development)

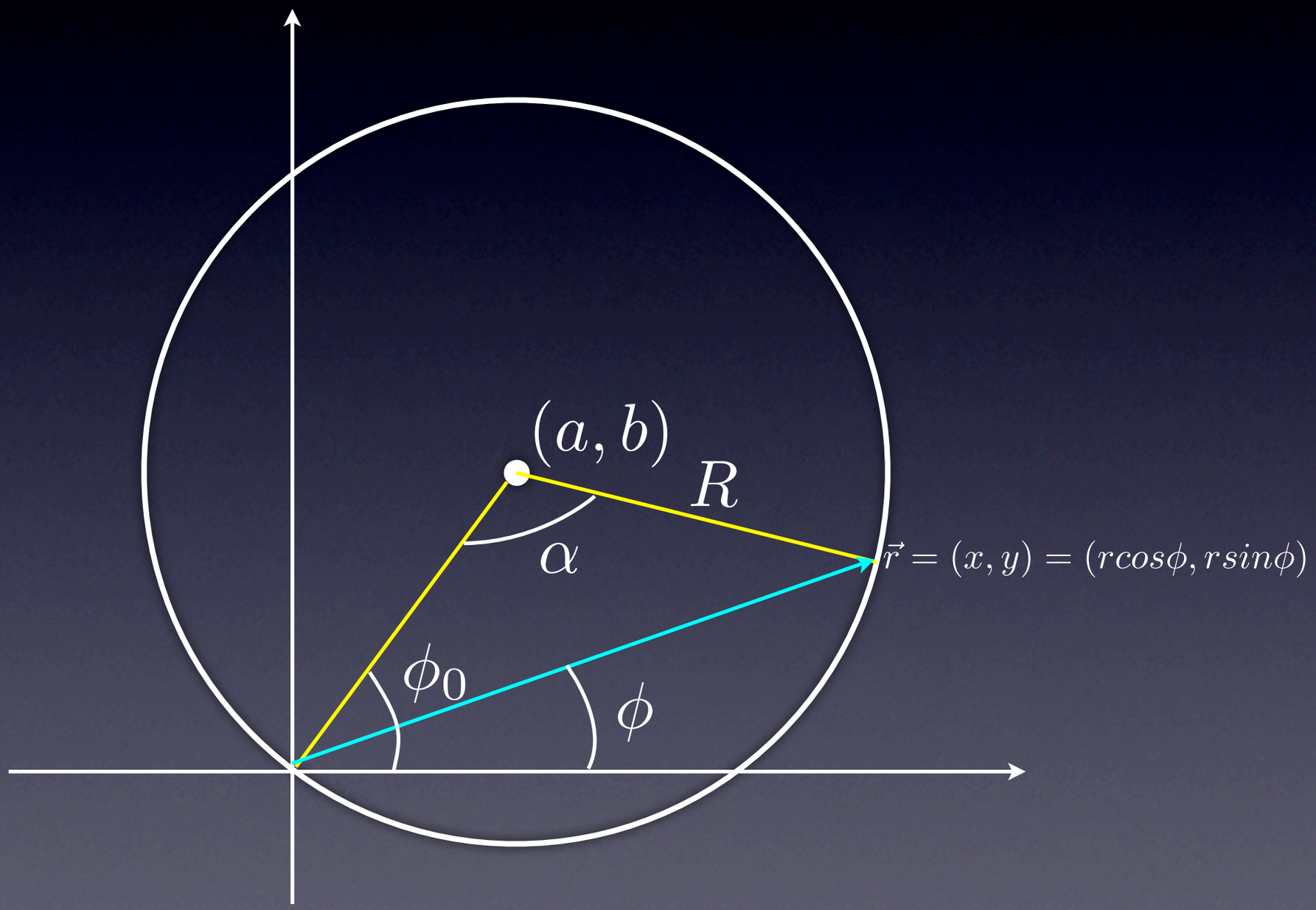
I'm sending 6 students to KEK this winter...





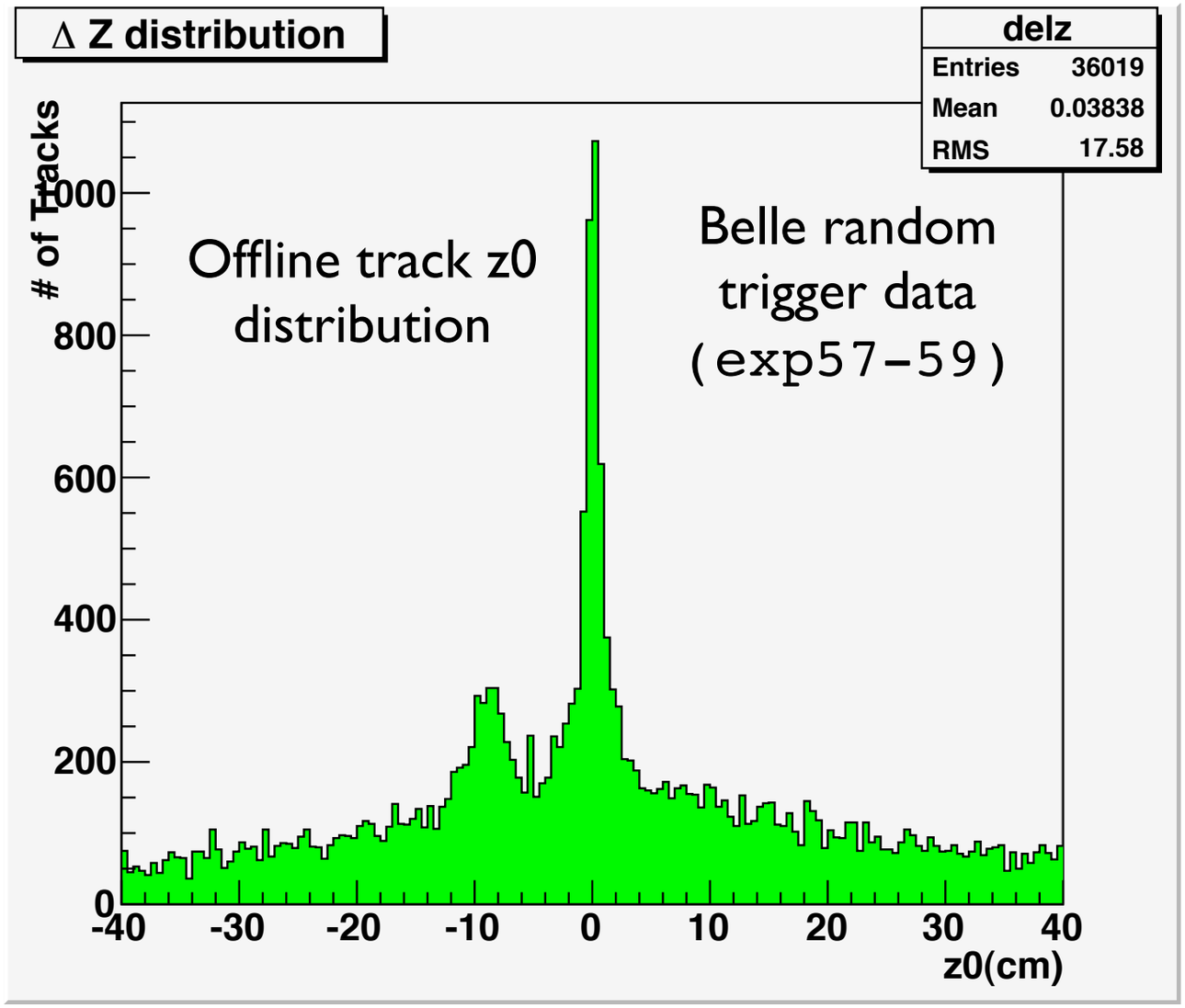
There is no summary, that's it.

Parameters

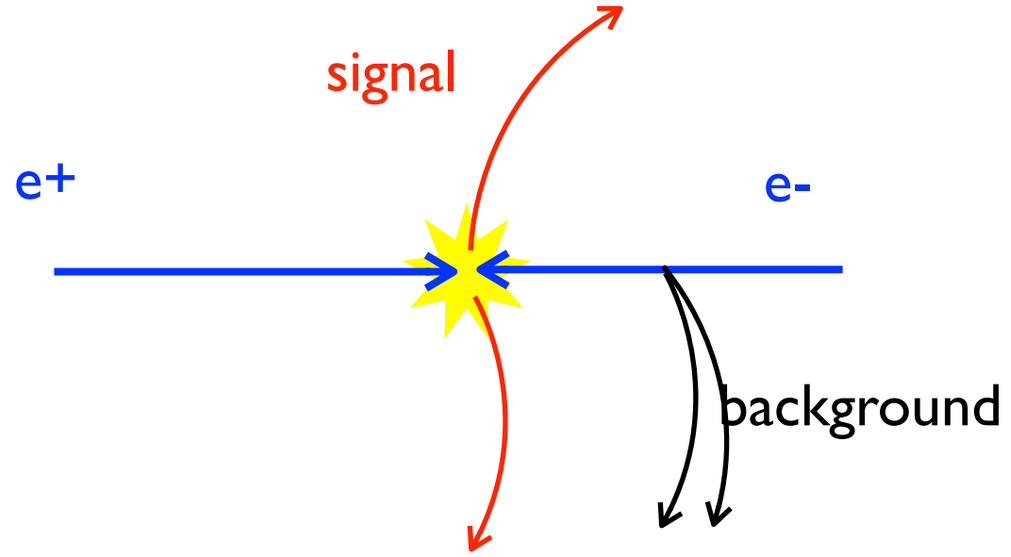


Introduction

- Level 1 reconstruction of 3D track parameters may help in reducing beam induced background



z0 : closest z coordinate of a track to the beam pipe



- This idea was demonstrated by BaBar: resolution of z0 ~ 4 cm

(Nucl. Instr. and Meth. A 518, 544 (2004).)

BaBar Document says...

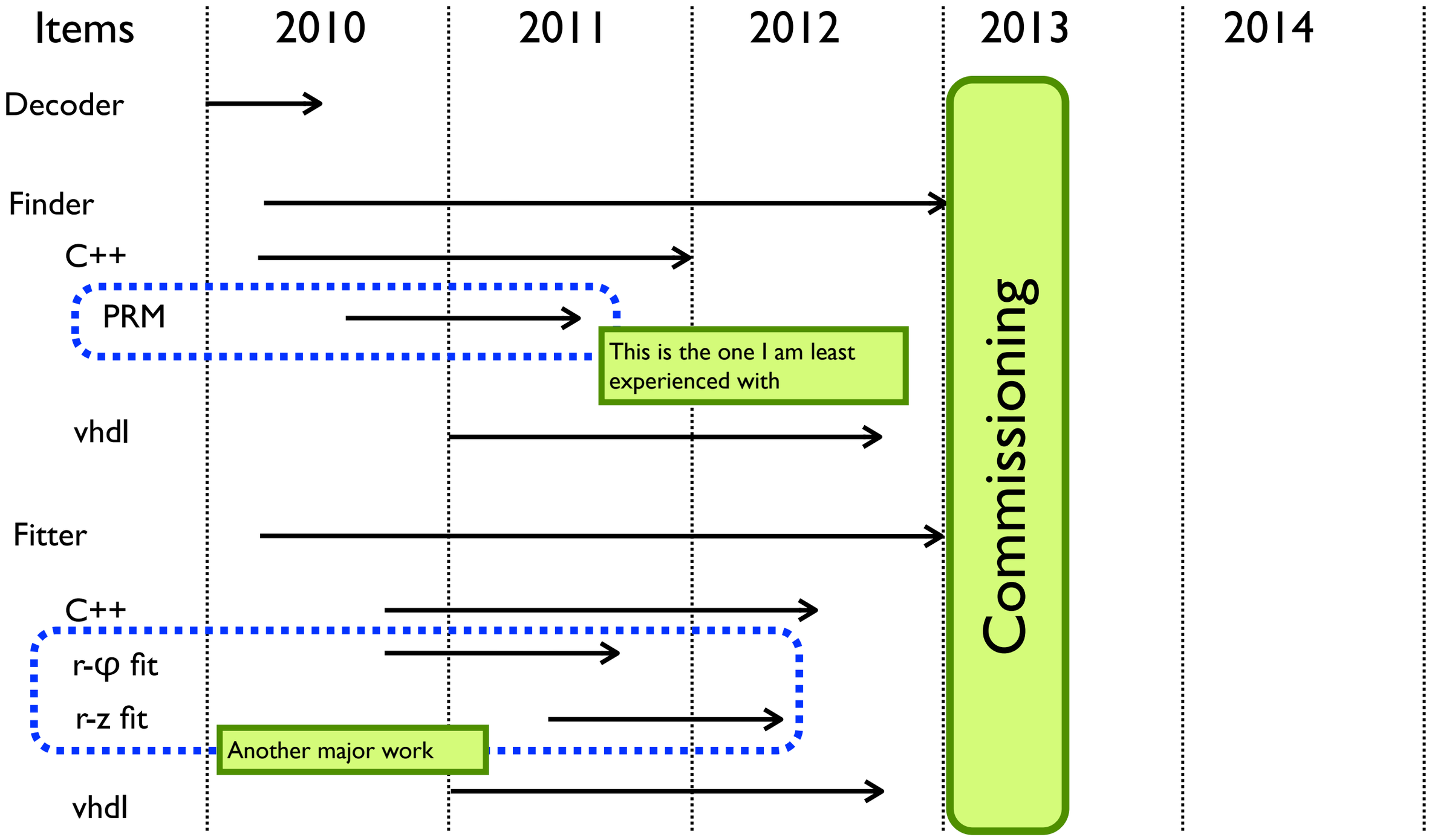
Stefan Gerhig Thesis (http://www.slac.stanford.edu/BFROOT/www/Detector/Trigger/, open to public)

Track Segment Finder

Segment finding is used frequently as an initial part of drift chamber trigger algorithms (see e.g. [ZEUS 92]). For *BABAR*, we also use crude drift time information. Hence different drift times for different wires have to be taken into account, which is done in a calibration step. The track segment finder receives the raw drift chamber hits as input. These hits start counters that count for a full drift time to produce at least one tick where all hits corresponding to one event are on. There are 1776 segment finders producing segments for every pivot cell (third cell within a superlayer) in the drift chamber. Segments that have adjacent hits in 3 or 4 out of 4 layers per superlayer are assumed to have a track traversing. These segments are sampled every 3.7 MHz. The other components of the segment finder run at the trigger clock speed, 7.4 MHz. The segment finder waits to report the "best" segment found for one drift time (typically the segment where all hits were on). The timing of these segments is latency adjusted and stretched to ensure coincidences in further processing.

So they do use timing information, but at TSF stage,
as I (we) guessed!

Plan toward Belle-II startup



Result from 3D fits

- There is ~ 15-28 % improvement in z0 resolution by moving from AUAVAUAVA to AUVAUVAUVA (from March B2GM)

φ=all, cosθ=0		Belle-II	Belle-II UV9	Belle-II UV10	Improvement in % (Belle-II vs VU10)
		z0 (σ_z0) cm = offset (resolution)			
pT = 0.8 GeV	w/o pseudo axial	0.86 (9.58)	0.17 (7.46)	2.12 (6.87)	28%
	with pseudo axial	-1.25 (8.70)	-2.24 (6.99)	-0.23 (6.60)	24%
pT = 1.0 GeV	w/o pseudo axial	0.45 (8.01)	-0.10 (5.97)	1.48 (6.43)	20%
	with pseudo axial	-2.00 (7.44)	-2.56 (6.03)	-1.01 (6.30)	15%
pT = 2.0 GeV	w/o pseudo axial	-0.12 (11.82)	-0.52 (8.89)	0.93 (9.33)	21%
	with pseudo axial	-3.55 (10.48)	-3.26 (8.31)	-1.69 (8.59)	18%
pT = 4.0 GeV	w/o pseudo axial	-0.57 (14.40)	-0.50 (10.99)	0.92 (10.89)	24%
	with pseudo axial	-4.37 (12.87)	-3.36 (10.27)	-1.72 (9.89)	23%

Algorithm Design Strategy

After several iterations of discussions with Y. Iwasaki, we agreed on

- YI develops his 2D finder (based on Hough transformation) and KU use it