Upgrade studies with the "Fast-Estimation" tool

29th of May, 2011

- ITS Upgrade Plenary Meeting -

Outline

1. Code Updates & Extensions

- Extension to “solution for lower pt” & “ITS standalone case”
- Implementation of Kalman version (previously Billoir)
- Calculation of “Track-finding Efficiency” was extended
- Updated QED calculations

2. Reminder on previously obtained results

- Performance comparison to “Real Data” and “full MC”
- General statements & properties of a first layer
- “Double-sided” Pixel detectors

3. Different design options and performances

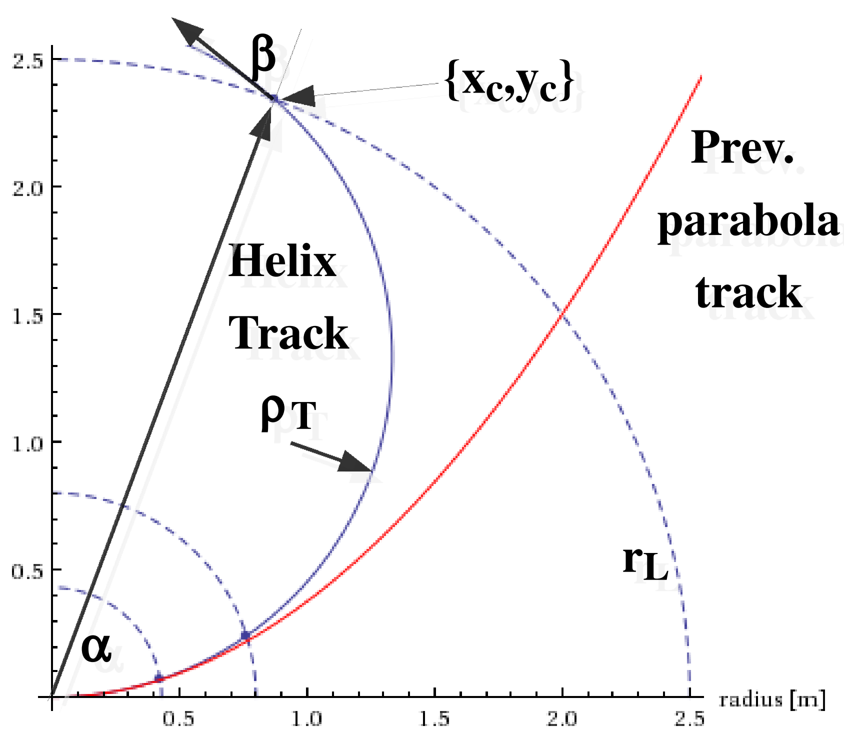
- General considerations
- Different designs with optimized layer positions

1. CODE UPDATES & EXTENSIONS

Extension to "solution for lower pt"

- The Billoir propagation matrix uses a "parabola" like track model which has problems if the curvature is large (low pt)
- An extension of the **"ideal cluster position"** within the code **using the exact intersection points of a "helix track"** with the layer solves the problem.

  → it emulates a "rotation into the tracking plane"



For "ITS+TPC" tracking, we can now go to pt~200MeV (compared to prev. pt~400MeV)

α... Angle of radial vector (or cluster position)
β... Track inclination in respect to layer normal

$$\alpha = ATan\left[\frac{y_c}{x_c}\right] \equiv \beta = ATan\left[\frac{p_y}{p_x}\right]_{rot.Frame}$$

$$a = \left[\frac{p_y}{p_x}\right]_{r.F.} = \tan[\beta] = \frac{r_L^2}{\sqrt{4 r_L^2 \rho_T^2 - r_L^4}} \quad with \quad \rho_t \geq r_L/2$$

$$b = \tan[\lambda]/\cos[ATan[a]]$$

1. CODE UPDATES & EXTENSIONS

Extension to "ITS standalone case"

- *Previous problem was the "Initialization Matrix":*

  → could bias the calculation if the number of layers is low

  → **Honest approach:** Start the calculation with "extremely large errors"

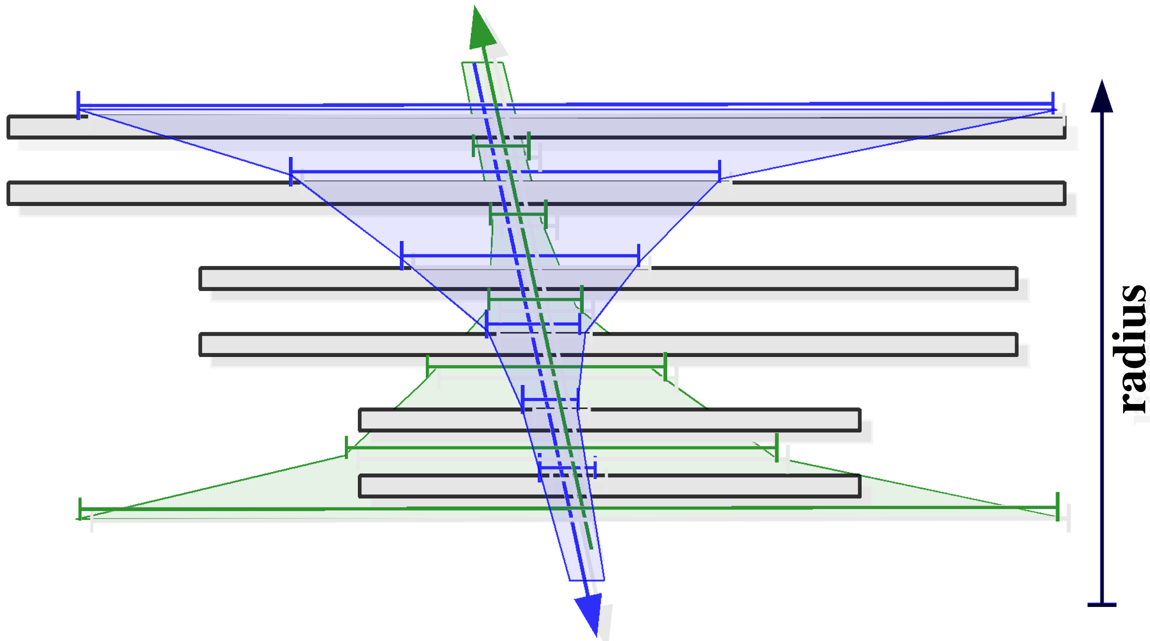
- *Efficiency calculation:*

  → Up to now we only used "forward fitting" (towards the vertex). For ITS standalone, we also need the "backward fitting" (starts from the vertex).

  → Efficiency calculation uses the "weighted estimate of the errors" from the forward and backward fitting

$$\Delta_w = \frac{1}{(\Delta_{fw}^{-1} + \Delta_{bw}^{-1})}$$

This approach allows a reliable Efficiency calculation for the "ITS standalone case"



1. CODE UPDATES & EXTENSIONS

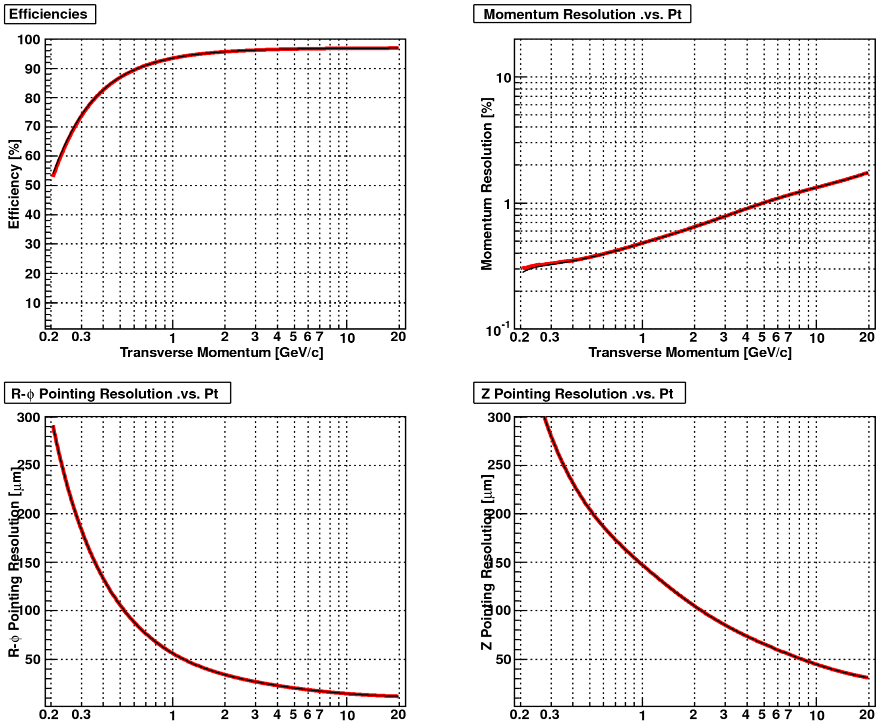
Kalman-version instead of Billoir

- The base code , e.g. “simple geometry” using cylindrical layers, is the same but the solving algorithm was exchanged with the “Kalman version“ within AliRoot …

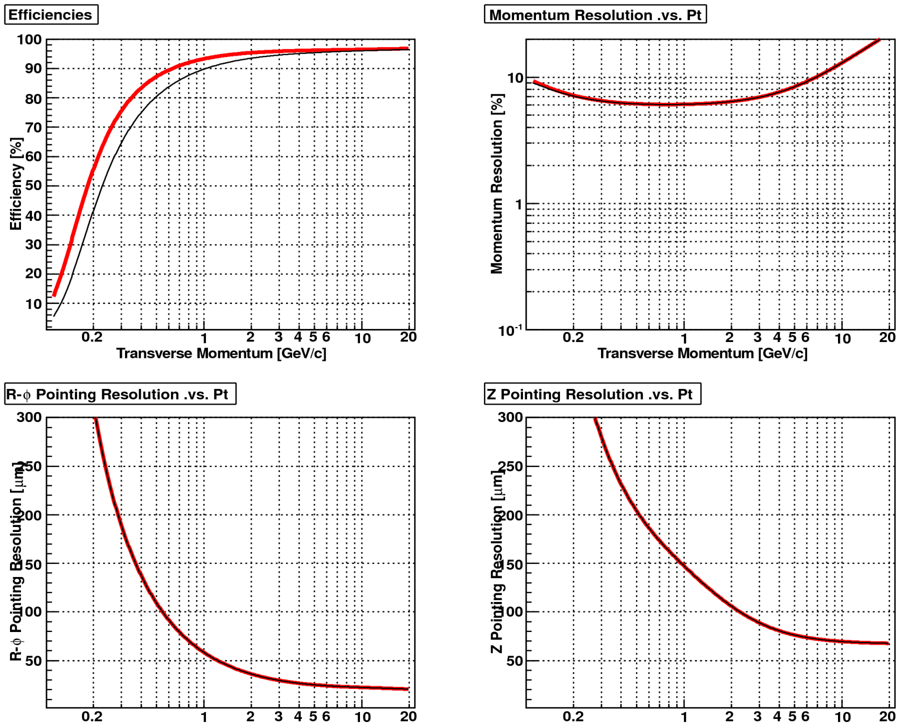
→ **Excellent agreement** for the “TPC+ITS” case in general

→ Excellent agreement for the ITS standalone case except for the “efficiency calculation” The AliKalman algorithm seems to “converge faster” ...

“TPC+ITS” for a Pion (current ITS conf.)



“ITS stand alone” for a Pion (current ITS conf.)



Red: Kalman version
Black: Billior version

1. CODE UPDATES & EXTENSIONS

Calculation of "Track-finding Efficiency" was extended

- Extended version proposed by Ruben Shahoyan:

  → includes "chi2" cuts on the cluster level (e.g. 3σ of confidence)

  → per layer efficiency of "showing the correct hit", (e.g. noise problems?)

Old calculation

$$P_{good} = \frac{1}{1+2\pi\rho\sigma_x\sigma_y}$$

(equivalent to 100% efficient layer and "inf. σ" of confidence)

New calculation

γ ... is fraction of good hits lost due to $\chi 2$ cut

ε_L … is layer hit efficiency (of showing the correct hit)

Match to correct candidate

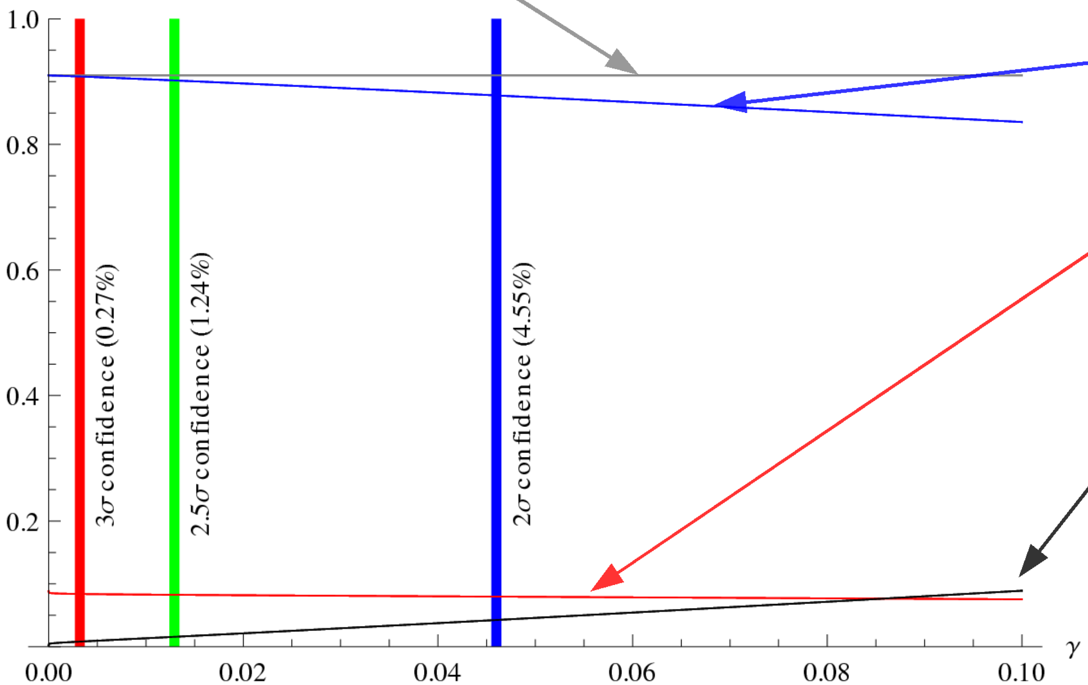
$$P_{good} = \epsilon_L \frac{1-\gamma^{1+2\pi\rho\sigma_x\sigma_y}}{1+2\pi\rho\sigma_x\sigma_y}$$

Match to a fake candidate

$$P_{fake} = 1 - P_{null} - P_{good}$$

No match at all

$$P_{null} = (1-\epsilon_L+\epsilon_L\gamma)\gamma^{2\pi\rho\sigma_x\sigma_y}$$



If the chi2 cut is "soft", the efficiencies do not change a lot

1. CODE UPDATES & EXTENSIONS

Updated QED calculations

- QED (or UPC electrons) can be a crucial part of the background

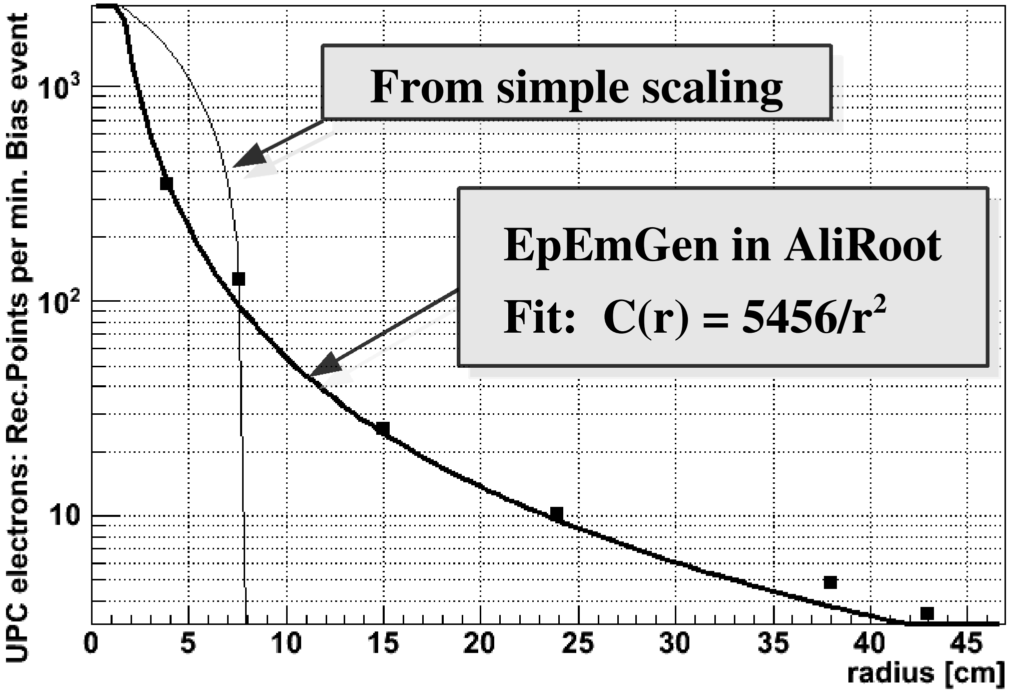
  → So far, we just scaled Star-Simulations to Alice expectations (factor 2.5 [1])

- We have a **Particle Generator** in AliRoot for exactly such processes [2]. Recalculations were performed using the current ALICE geometry ...

At a radius of r=2.2 cm, we expect approx. 1100 clus/min.BiasEv/eta

→ This means, so far we have overestimated this effect by roughly a factor of 2 (at a radius of 2.2 cm)

→ we were on the safe side



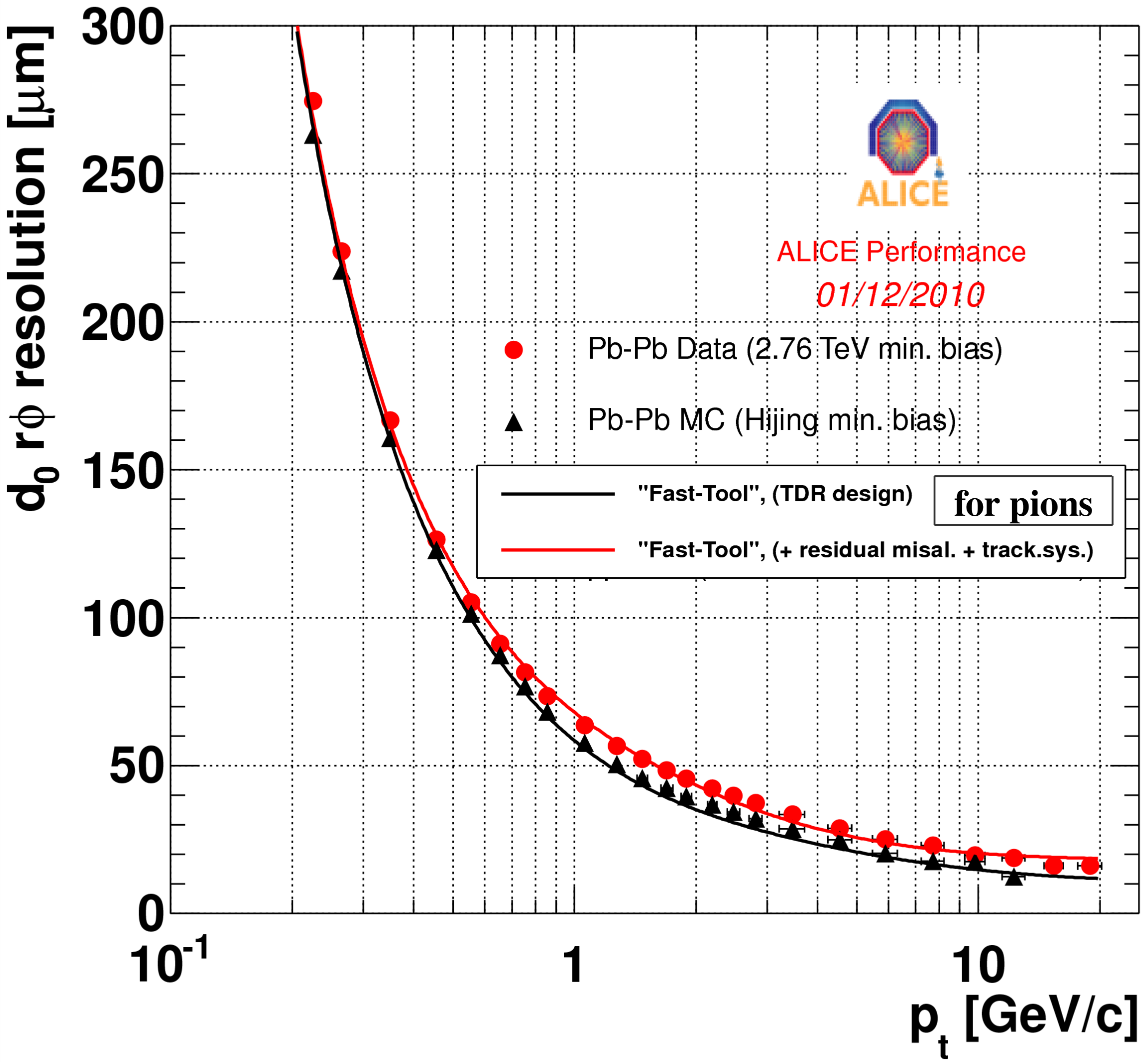
Detailed presentation can be found here: https://indico.cern.ch/conferenceDisplay.py?confId=131407

*[1] K. Hencken, et.al., **Production of QED pairs at small impact parameter in relativistic heavy ion collisions**, Physical Review C, vol. 69, Issue 5, id. 054902 (2004) doi = 10.1103/PhysRevC.69.054902*
*[2] S.Sadovsky,K.Hencken, Yu.Kharlov. **Generator for e+e- pairs in PbPb collisions at LHC**, ALICE-INT-2002-27*

2. Reminder on previously obtained results

Performance comparison to “Real Data” and “full MC”

Excellent agreement to “Full MC” and “Real Data”



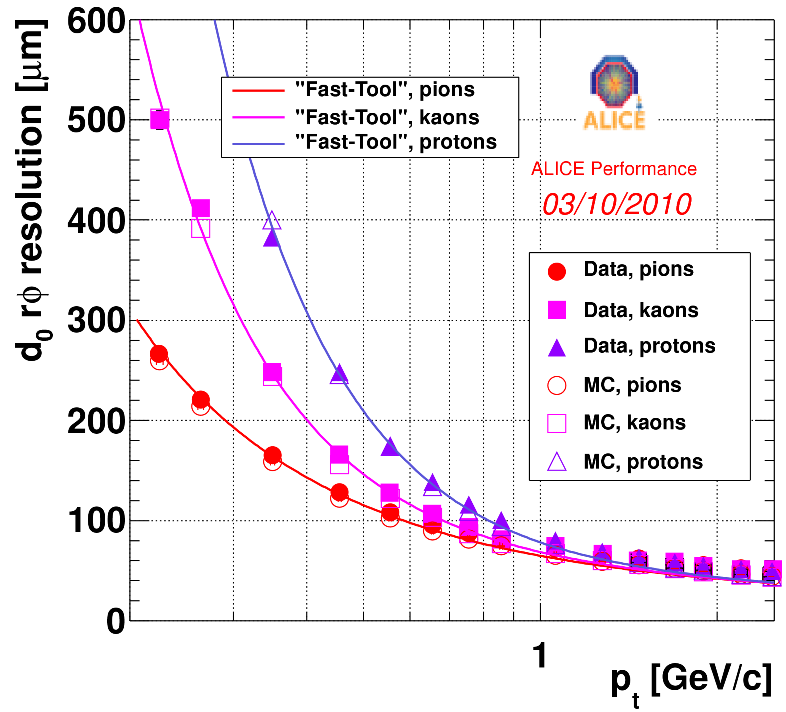


Table 1: Current ITS layout

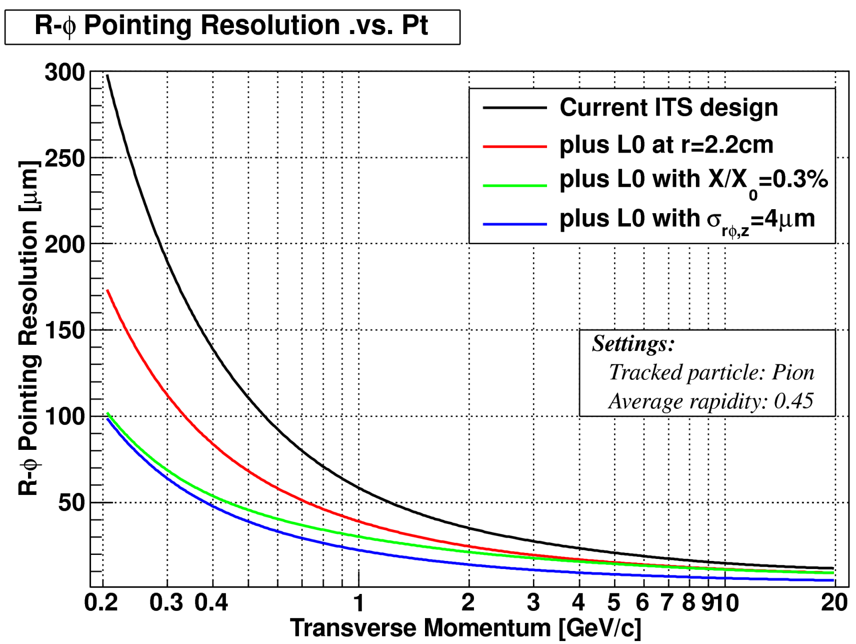
Name	radius [cm]	Material budget [%]	Resolution in $r\phi$ [μm]	Resolution in z [μm]
Beam pipe	2.94	0.22	–	–
SPD1	3.90	1.14	12	130
SPD2	7.60	1.14	12	130
Thermal shield 1	11.50	0.65	–	–
SDD1	15.00	1.13	35	25
SDD2	23.90	1.26	35	25
Thermal shield 2	31.00	0.65	–	–
SSD1	38.00	0.83	20	830
SSD1	43.00	0.86	20	830

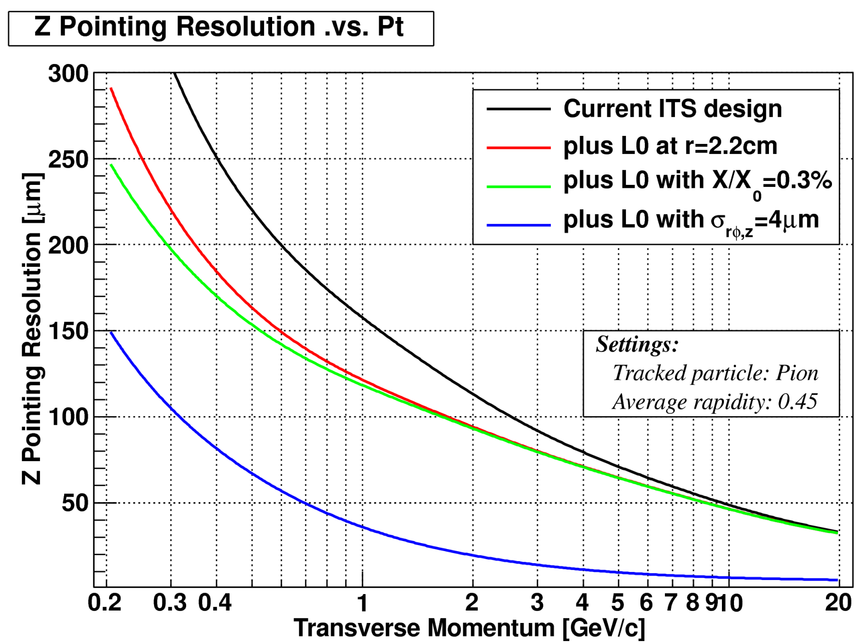
2. Reminder on previously obtained results

General statements & properties of a first layer

- **Pointing resolution to the vertex** depends mostly on the properties of the **first 2 layers**
  → Material thickness (X/X_0), radial position (R), intrinsic resolution ($\sigma_{r\phi}$,σ_z)
- Only at **high pt** (>2 GeV), the ITS **layers further out** (plus the TPC) become important …
- **Other layers** in between the first Pixel detectors and the TPC are **important for the Track-Finding efficiency** (otherwise the distances, and therefore the extrapolation errors, increase)

That can be obtained with one single improved layer (L0) close to the beam pipe ...





Note: Current SPD1, r = 3.9 cm, X/X_0=1.14 %, (σ_r, σ_z)=(12,130) μm

2. Reminder on previously obtained results

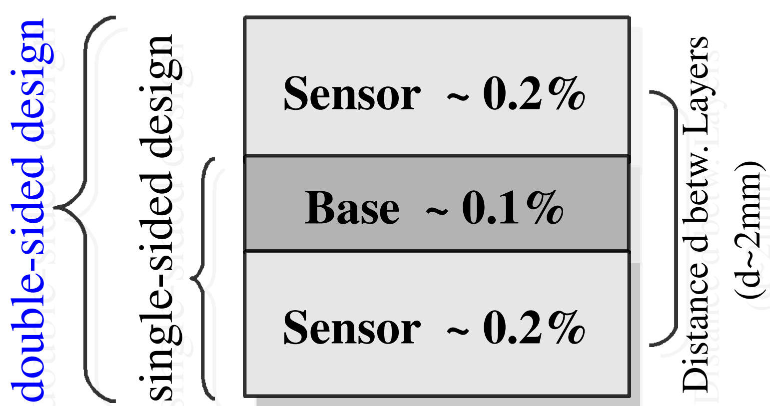
"Double-sided" Pixel detectors

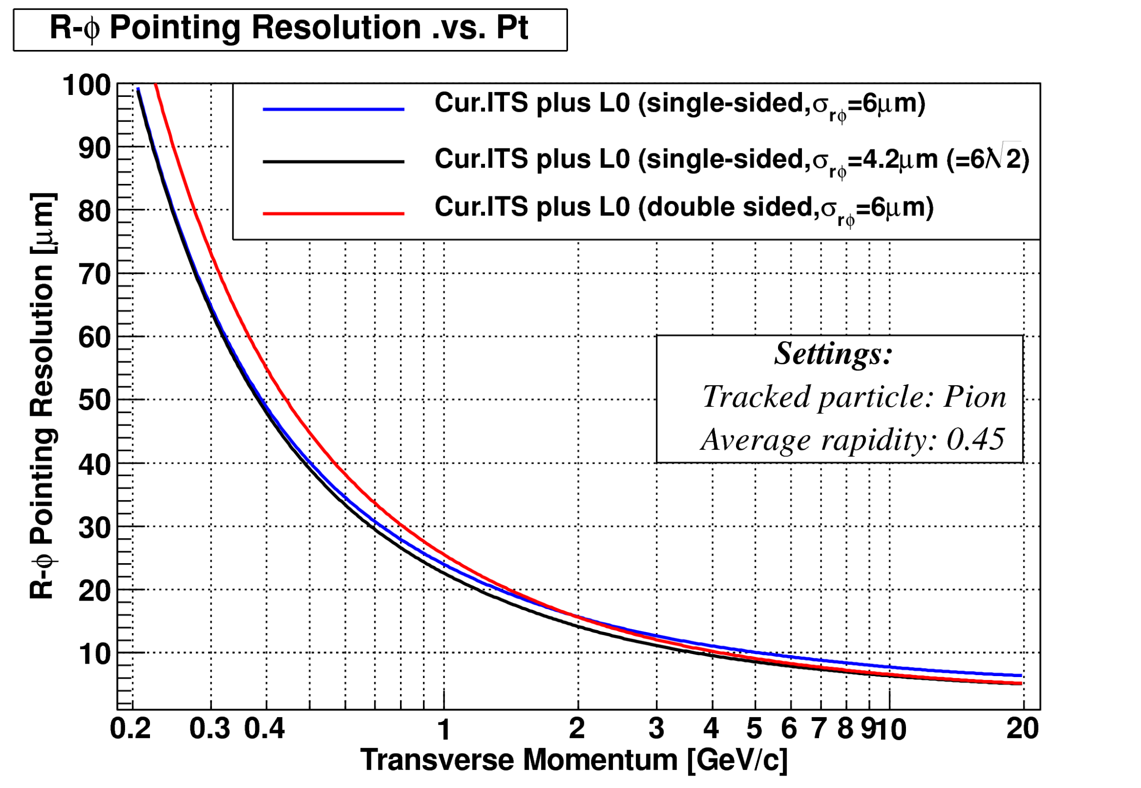
... would be an opportunity to improve the detector resolution by a factor of sqrt(2)
But, the track resolution only gets better at high pt (due to the additional material, it gets worse at low pt)

Simulation details:

Current ITS plus a layer zero (L0)

- Radius: r = 2.2 cm
- Resolution: $(\sigma_{r\phi}, \sigma_z)=(6,6)\ \mu m$
- Assumed material budgets:





- Studies by Serhiy Senyukov indicate a possible usage of "mini-vectors" in order to improve the tracking efficiency for such layers!

(Details can be found here: https://indico.cern.ch/materialDisplay.py?contribId=7&materialId=slides&confId=131406)

3. Different design options and performances

General considerations

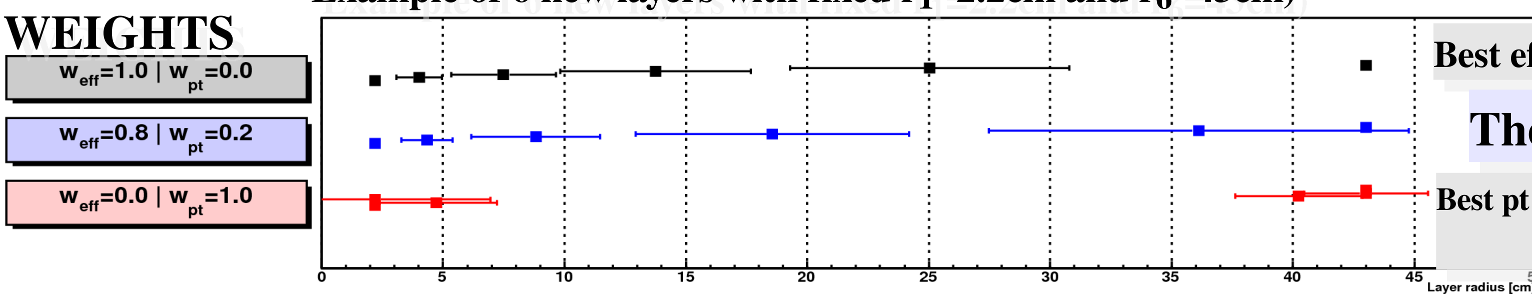
General boundaries of a new design

- Radius of "**outermost" layer r_{Ln}~ 43 cm** (= current)
- Radius of **"innermost" layer** depends on beam-pipe radius; **r_{L1}~ 2.2 cm** is likely (currently 3.9cm)

The general performance (e.g. efficiency, pt resolution) can be optimized in dependence of …

- The layer properties, e.g. find the optimal radial position
- Total number of layers a.s.o.

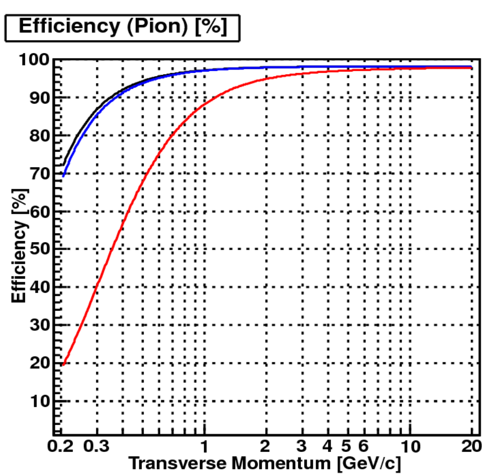
Example of 6 new layers with fixed r_1=2.2cm and r_6=43cm)

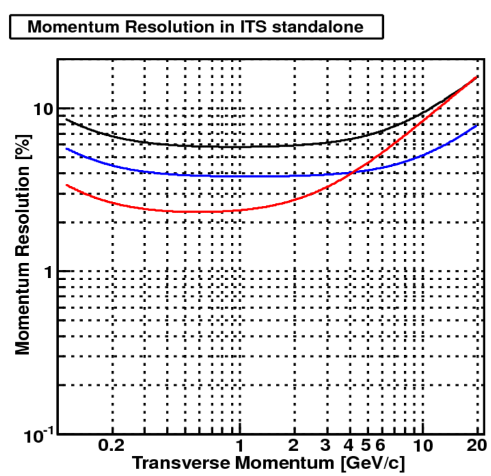


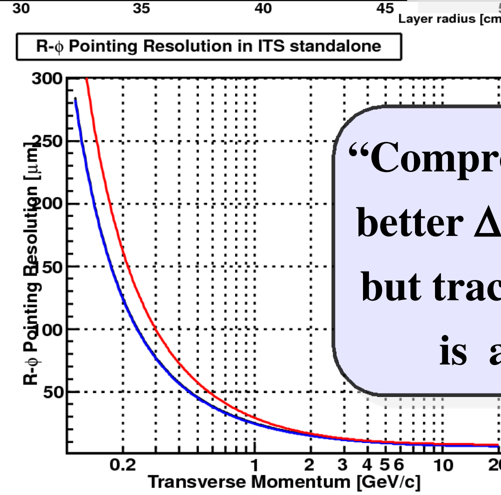
Best eff. In TPC+ITS mode

The compromise

Best pt resolution (at 750 MeV) for ITS standalone







"Compromise (blue)" delivers better Δpt in ITS standalone, but track resolution and eff. is almost untouched

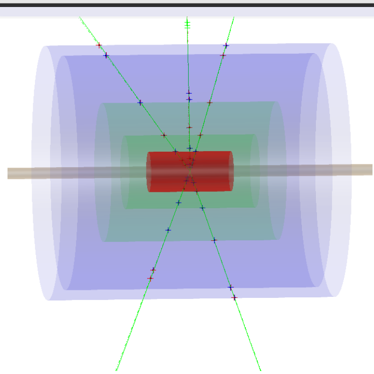
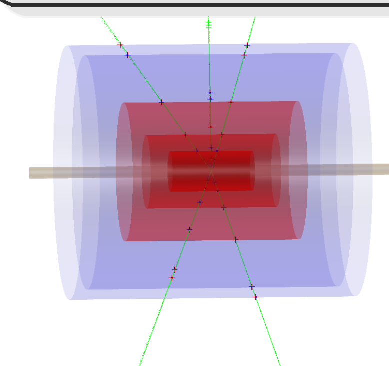
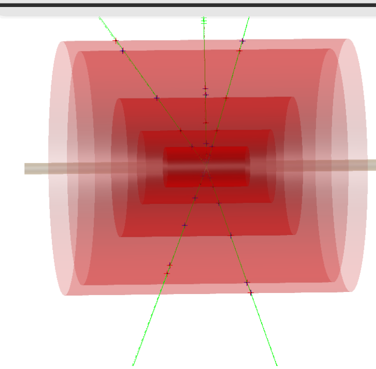
3. Different design options and performances

Different designs with optimized layer positions

NEW Pixel layers *(e.g. Hybrid like)*

$X/X_0 = 0.5\%$; $(\sigma_{r\phi}, \sigma_z) = (6,6)\ \mu m$

Fixed innermost layer: $R_1 \sim 2.2$ cm

Exchange SPDs	Only keep SSDs	Exchange everything ALL NEW
		

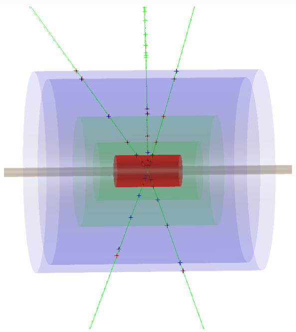
Note in advance: *There is essentially* ***no difference for the Pointing Resolutions****, but the impact on the* ***efficiency*** *and the* ***pt resolution*** *can be* ***dramatic*** *...*

3. Different design options and performances

Different designs with optimized layer positions

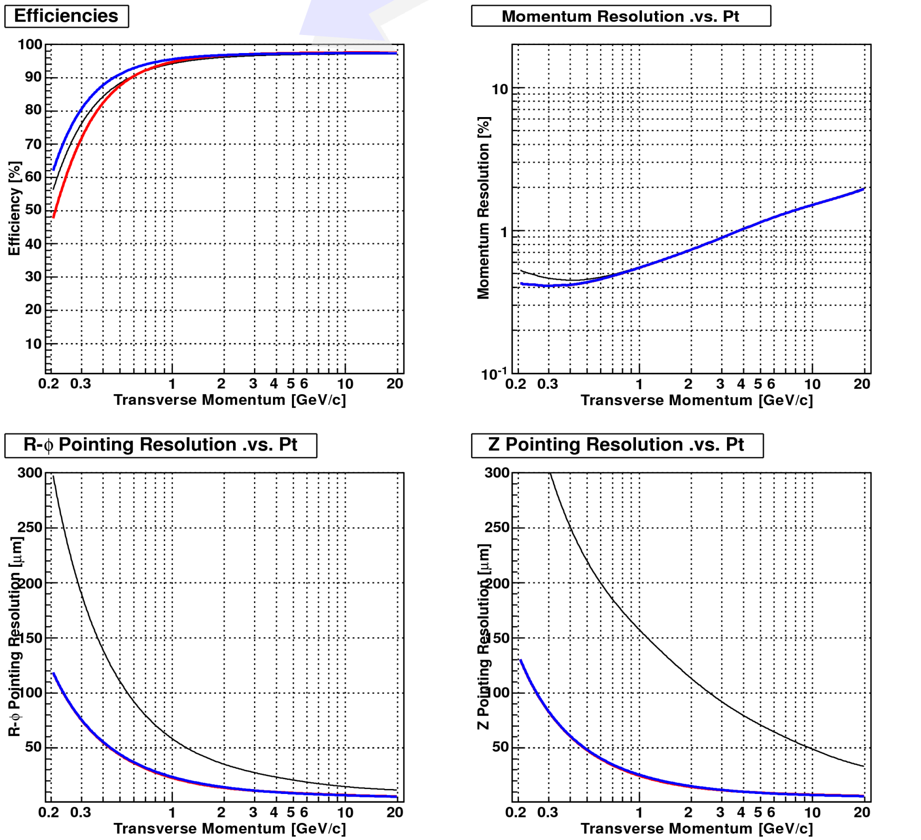
Exchange SPDs

The options are : 2 or 3 new layers … ?
→ 3 LAYER option is BETTER!
→ Efficiencies are improved!

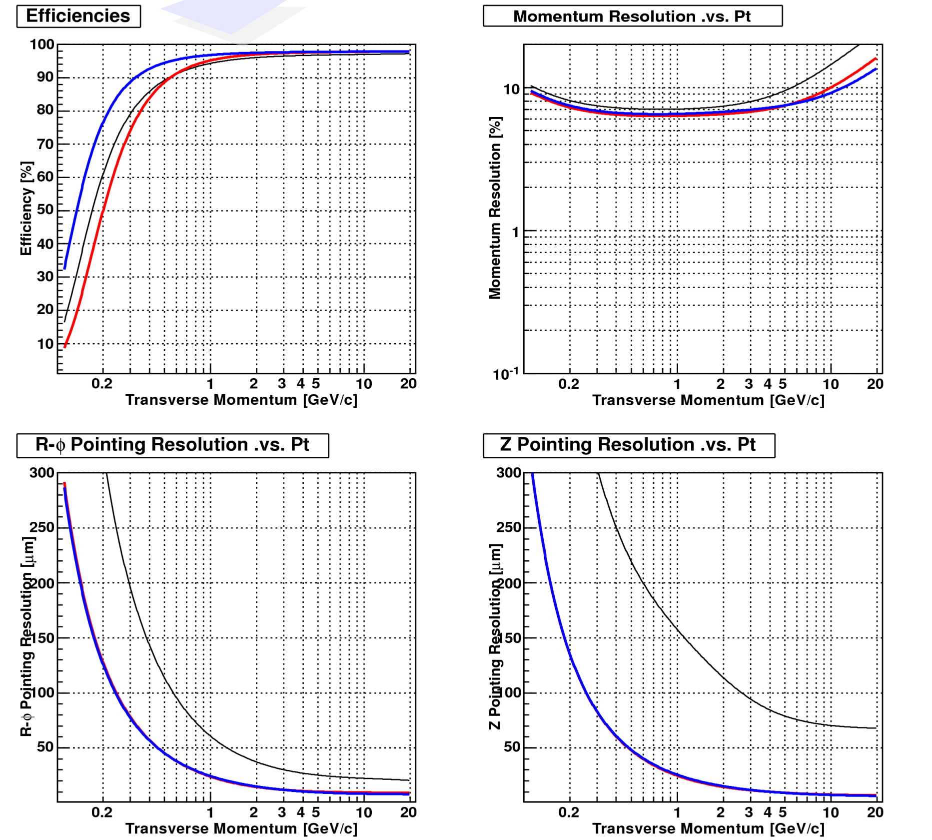


- current design
- 2 new layers r=(2.2, 6.6) cm
- 3 new layers r=(2.2, 4.8, 9.1) cm

"TPC+ITS" for a Pion



"ITS stand alone" for a Pion



3. Different design options and performances

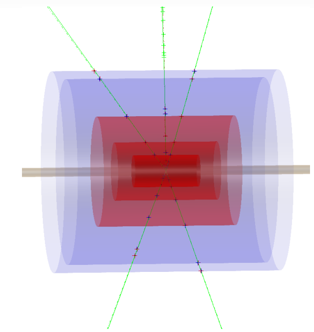
Different designs with optimized layer positions

Only keep SSDs

The options are : 4 or 5 new layers … ?

→ Better pt resolution in ITS standalone!

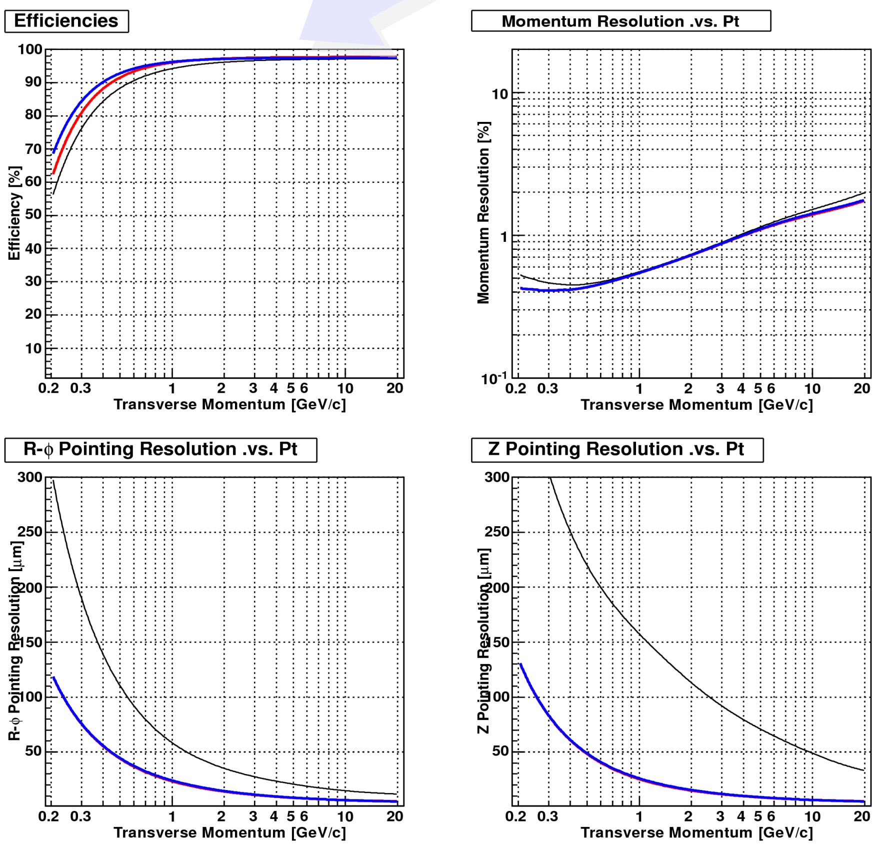
→ 5 Layer option even better efficiencies!

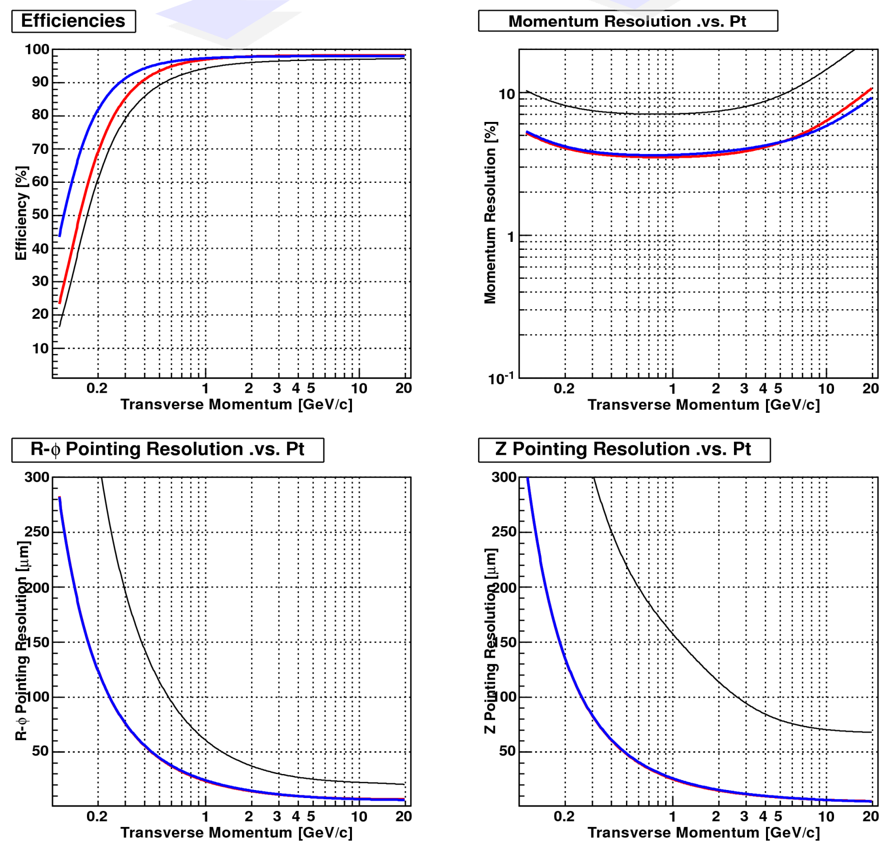


“TPC+ITS” for a Pion

- **current design**
- **4 new layers r=(2.2, 5.3, 12.9, 26.5) cm**
- **5 new layers r=(2.2, 4.3, 8.8, 18.2, 31.4) cm**

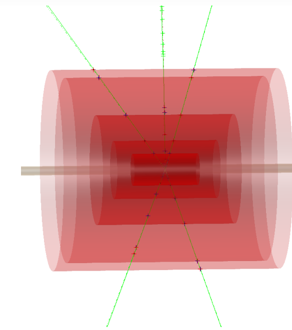
“ITS stand alone” for a Pion





3. Different design options and performances

Different designs with optimized layer positions

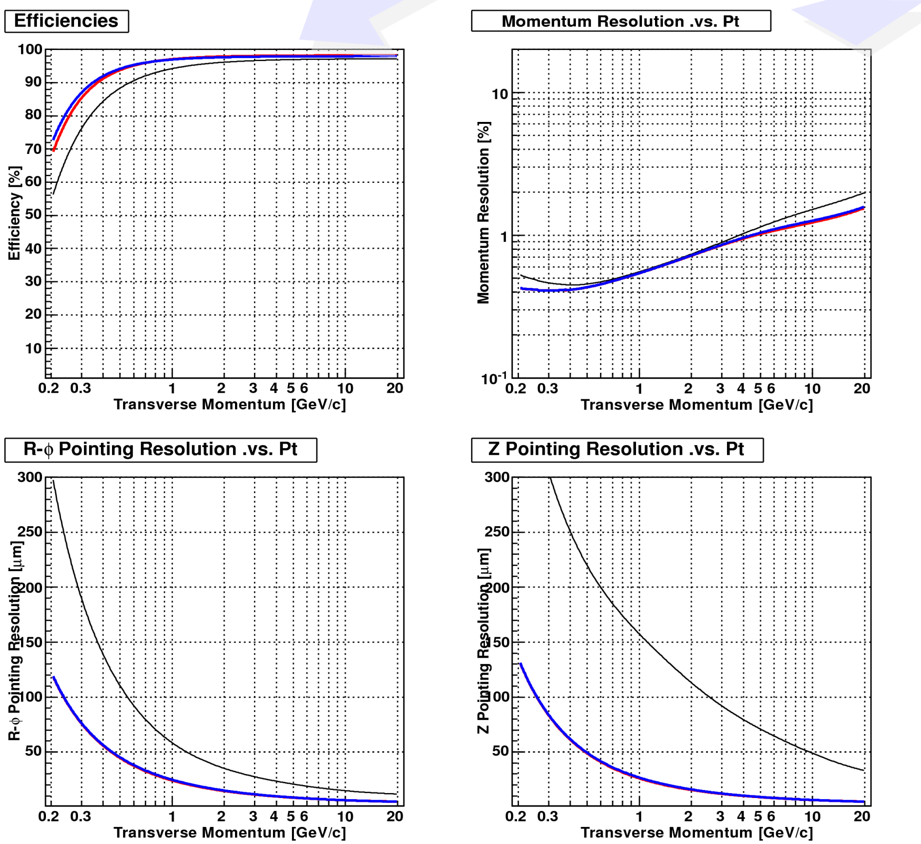


Exchange everything
ALL NEW

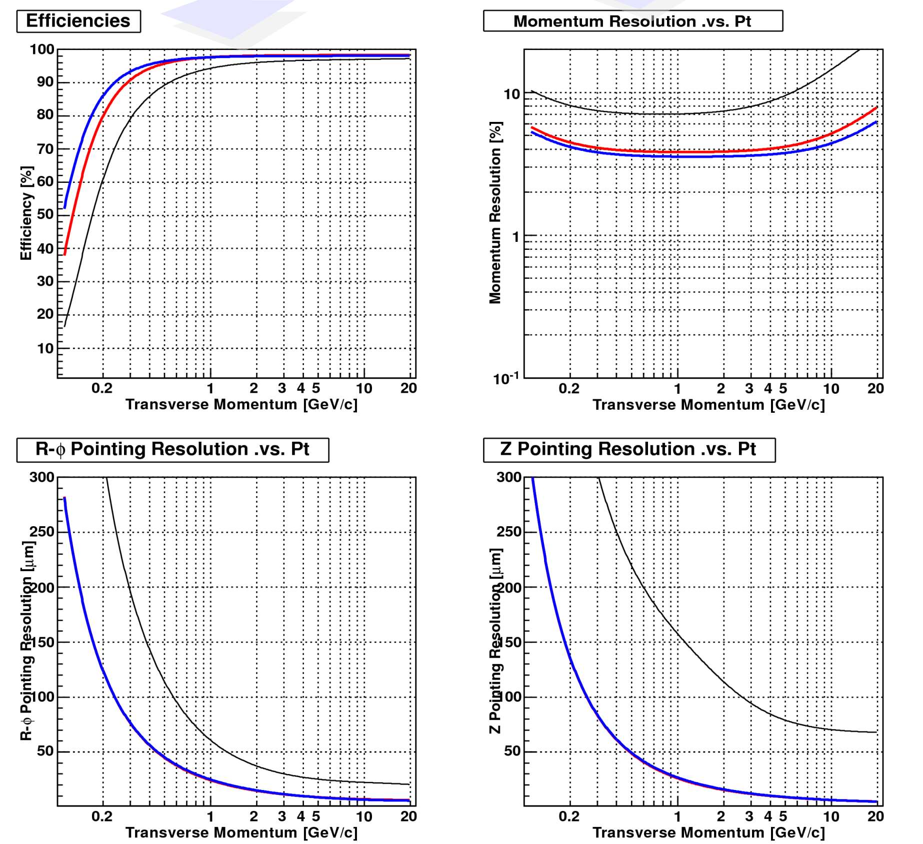
The options are : 6 or 7 new layers … ?
→ 7 LAYER option even BETTER!
→ Efficiencies and pt resolutions are further improved!

- current design
- 6 new layers r=(2.2, 4.3, 8.8, 18.6, 36.2, 43.0)cm
- 7 new layers r=(2.2,3.8,6.9,12.5,24.0,40.1,43)cm

"TPC+ITS" for a Pion



"ITS stand alone" for a Pion



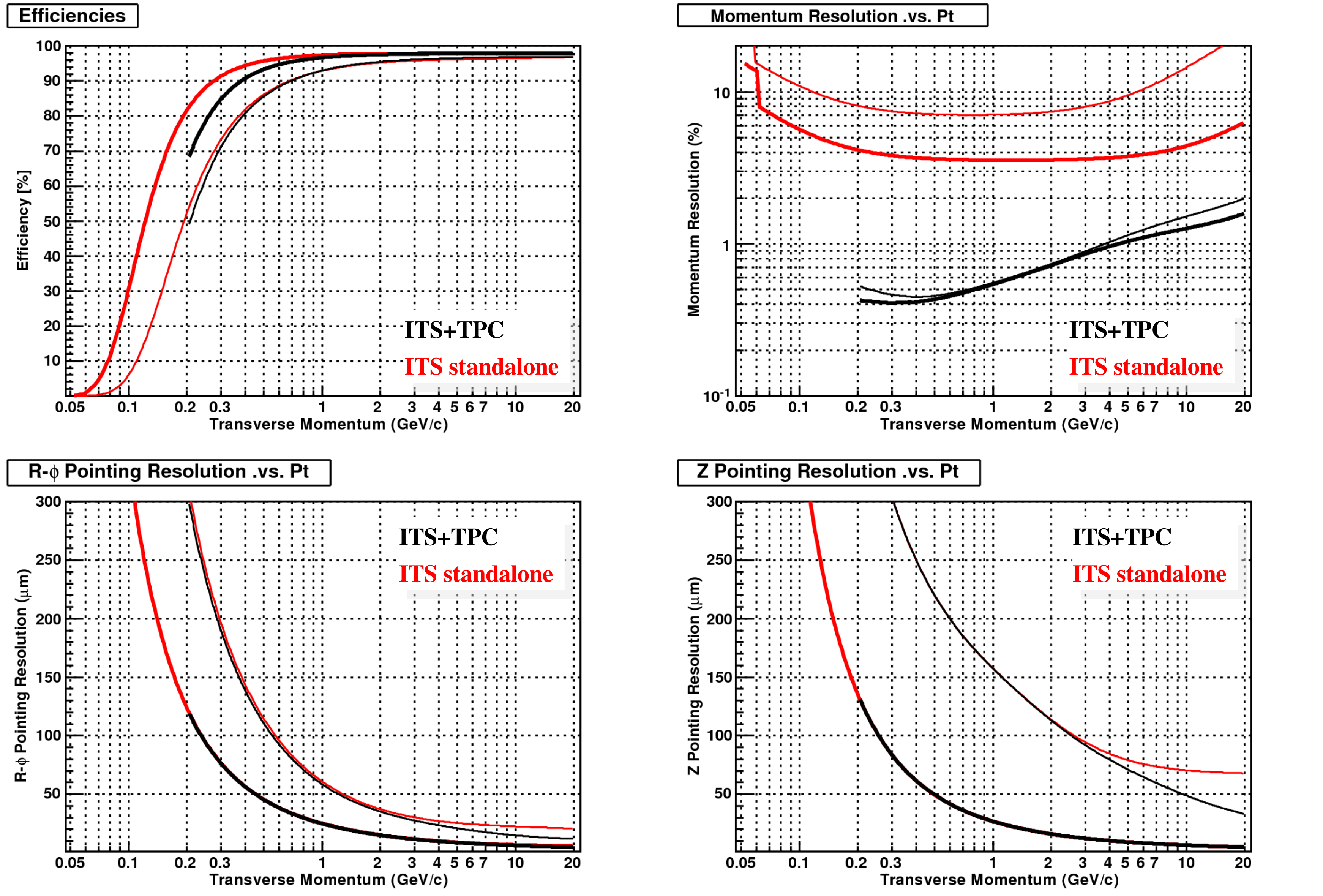
How performance plots could look ...

“ALL NEW” with Hybrid-like pixels ($X/X_0 = 0.5\%$; $(\sigma_{r\phi}, \sigma_z) = (6,6)$ μm)

All plots for PIONS

Thin lines → current ITS setup

Thick lines → “All New” with 7 layers



How performance plots could look ...

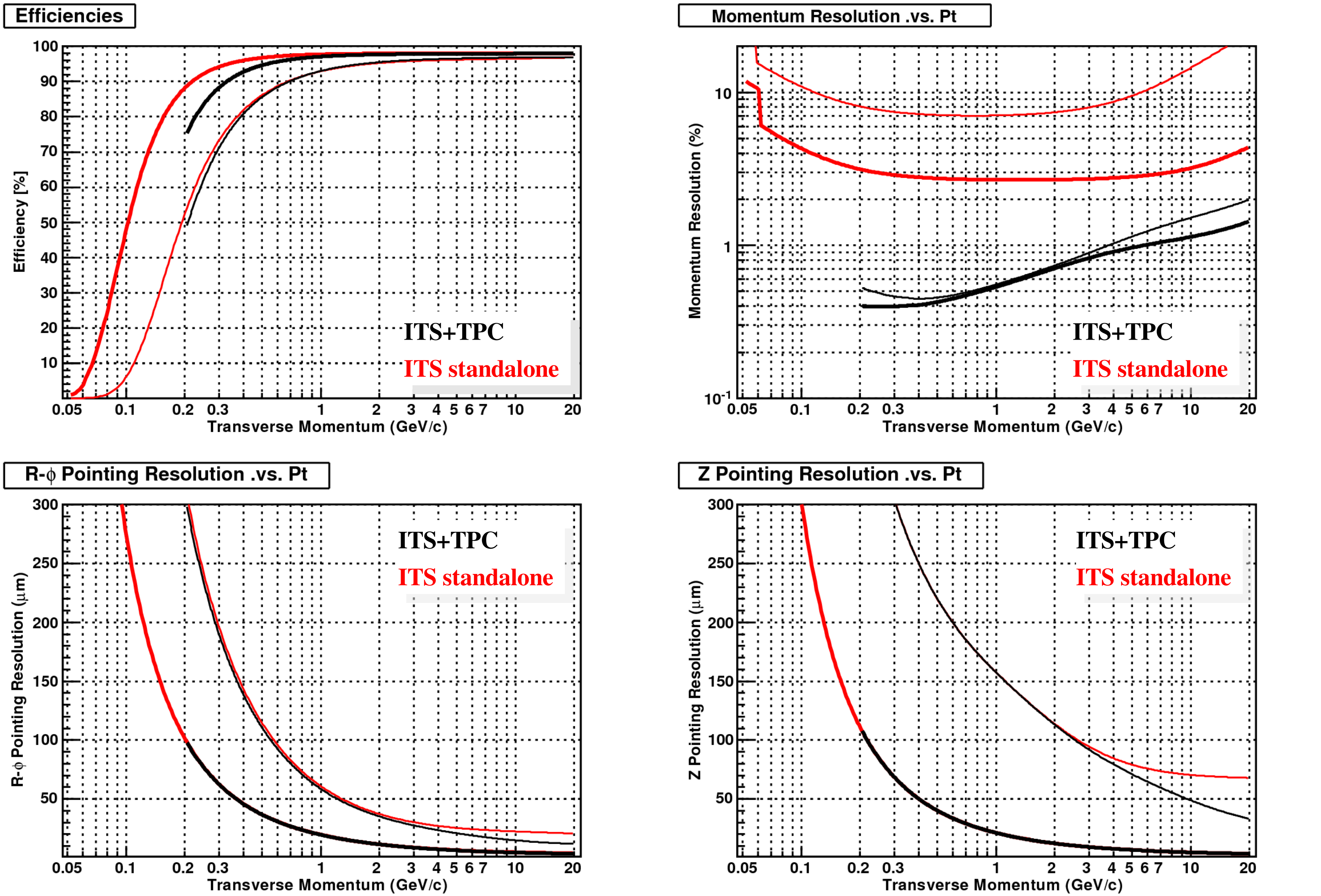
"ALL NEW" with Monolithic-like pixels ($X/X_0 = 0.3\%$; $(\sigma_{r\phi}, \sigma_z) = (4,4)\ \mu m$)

→ improved Material budget and resolution

All plots for PIONS

Thin lines → current ITS setup

Thick lines → "All New" with 7 layers



Conclusion & Open Points

- **"Fast-Estimation tool"** (which is based on the tool of Jim Thomas), is able to give a fast feedback on the influence of detector properties and the possible performance improvements due to different design options ...
- It was used extensively to produce the necessary **inputs for particle dependent performance improvements** which are currently used for the analysis of physics benchmark channels (e.g. D^0 in "hybrid" and "smearing" approaches)
- It can not only be used as a guideline but even provide fast feedback on questions for "optimizations" (e.g. radial positions of the layers)

OPEN POINTS:

- Ongoing discussions to extend to tool to the "Forward Region"