Bundesministerium für Bildung und Forschung

BMBF Forschungsschwerpunkt ALICE Experiment 202

Online track fit for the ALICE TPC detector in Online-Offline framework for the Run 3

Sergey Gorbunov
for the ALICE collaboration

Frankfurt Institute for Advanced Studies, Goethe University Frankfurt

ALICE

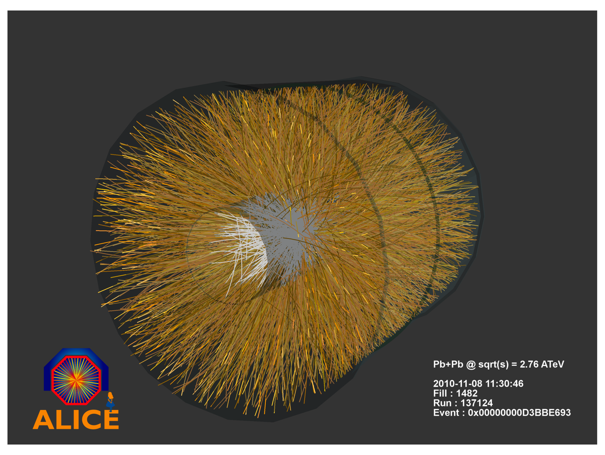
Common Online-Offline track fit aimed for the Run 3

The upcoming LHC Run 3 brings new challenges for the ALICE online reconstruction.
In particular, it will be also used for the offline data processing in the O2 (combined Online-Offline) framework.

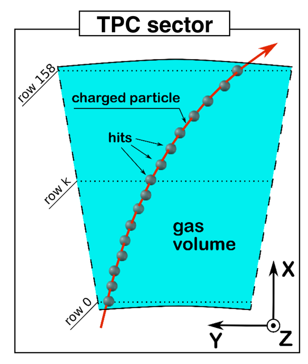
To improve the accuracy of the existing online algorithms they need to be enhanced with all the necessary offline features, while still satisfying speed requirements of the synchronous data processing.

Here we present our enhancements to the track fit algorithm which is currently used in the ALICE High Level Trigger (HLT) for the online reconstruction. The algorithm is based on the Kalman filter method. The developed fitting utilities are used both at the combinatorial track finding stage and at the final track fit stage of the track reconstruction.

TPC detector Geometry

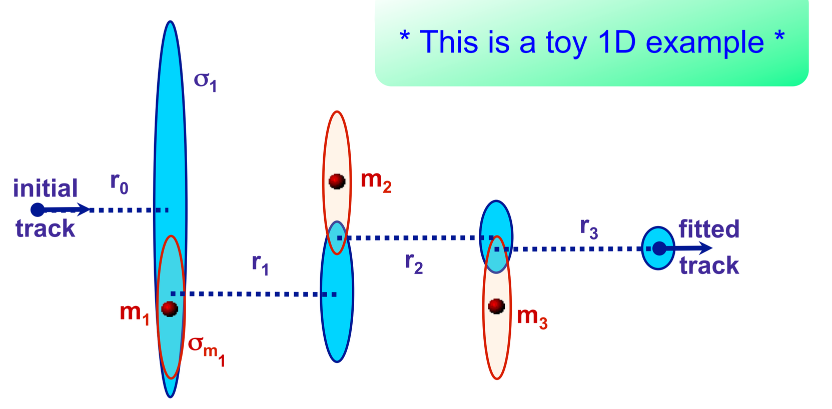


- Track fit is performed in local coordinates of correspondig TPC sector
- Helix track parameterisation: (x; y, z, $p_y/p_t=\sin(\varphi)$, $p_z/p_t=\tan(\theta)$, q/p_t) x is not fitted



Kalman Filter development: single precision for fast parallel calculations on GPU

* This is a toy 1D example *

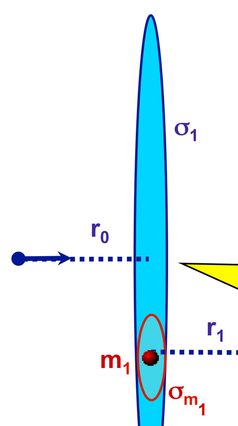


$$\sigma^2_{k+1} = \sigma^2_k - \frac{\sigma^2_k}{\sigma^2_k+\sigma^2_{m_k}}\cdot\sigma^2_k$$

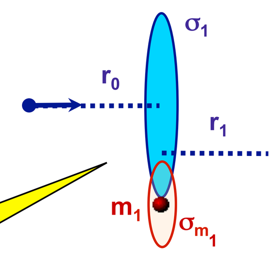
$$r_{k+1} = r_k - \frac{\sigma^2_k}{\sigma^2_k+\sigma^2_{m_k}}\cdot(m_k - r_k)$$

$\sigma^2_{m_k}$ vanished when $\sigma^2_{m_k} \ll \sigma^2_k$

Computational problems appear when errors of different orders of magnitude are combined.
I.e. big initial error with small measurement error.



Big initial error:
Mathematically correct, but produces roundoff problems.



Small initial error:
No roundoff problems but wrong results.

Operating the Kalman fiter in single precision

=> Calculate errors and parameters separately:
Start with small initial error, but treat it in a speciall way

$$\sigma^2_{k+1} = \sigma^2_k - \frac{\sigma^2_k}{\sigma^2_k+\sigma^2_{m_k}}\cdot\sigma^2_k$$

independent loop for the error evaluation

$$r_{k+1} = r_k - \frac{\sigma^2_k}{\sigma^2_k+w\cdot\sigma^2_{m_k}}\cdot(m_k - r_k)$$

$$w = (\sigma_k < 10\cdot\sigma_{m_k})$$

case ($\sigma_k \ge 10\cdot\sigma_{m_k}$) treated as ($\sigma_k = \infty$)

In the general case the Square-root Kalman filter can be used (way more slower and complicated)

Fast track propagation in 3D field

- **Accurate propagation for the linearisation trajectory t_0, simple propagation for the track t**

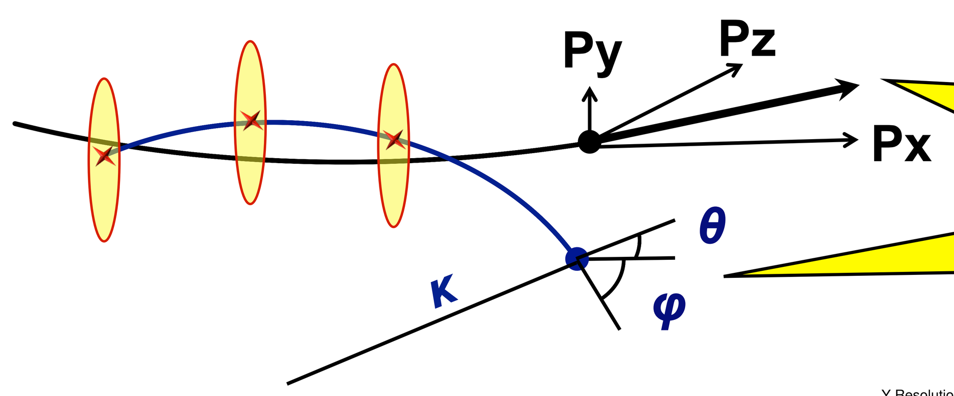
1) As Bx,By field components are minor and have only second-order effect and
2) the explicit linearisation at best-known t_0 trajectory is used,

one can use the full field propagation for t_0 trajectory and simple (Bz only) propagation for the track t and its covariance matrix.

- **Use different parameterisations for t0 and t**

As the t0 is only propagated and not fitted, one can use a parameterisation which is appropriate for the propagation, and not for the fit.
=> Let's use the physical parameterisation which is natural for the propagation in 3D magnetic field.

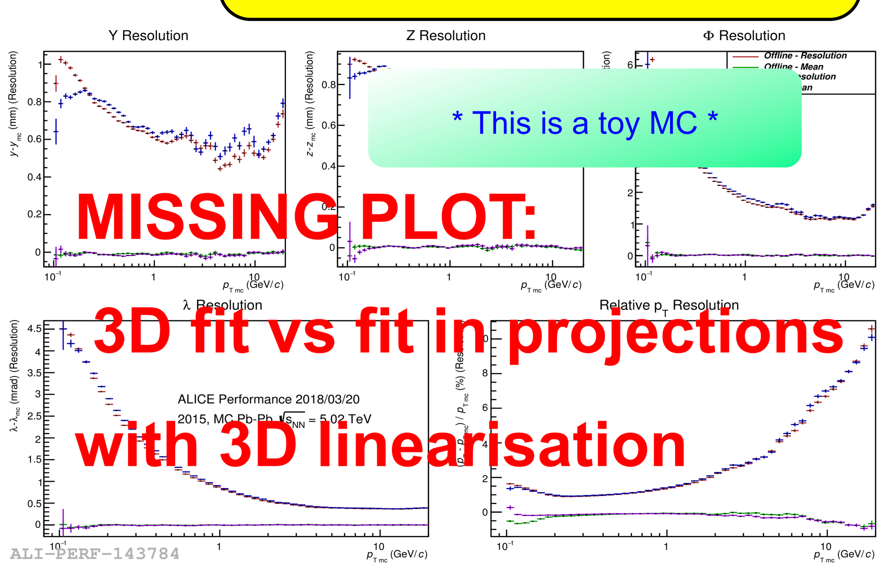
Propagation scheme



Linearisation t_0
- **Parametrised as (x, y, z, Px, Py, Pz)**
- **Propagated in the complete field (Bx,By,Bz)**

(partially) Fitted track t
- **Parametrised as (x; y, z, sin(φ), tan(θ), q/pt)**
- **Propagated in a simple 1-D field (0,0,Bz)**

- **Split XY and Z components of the covariance matrix: do fit in projections, but linearisation in space**
  - evaluate 9 elements instead of 15
  - calculate 13 derivatives instead of 25
  - no degradation of the fit quality when the linearisation is good (during final refit)



Kalman Filter : explicit linearisation

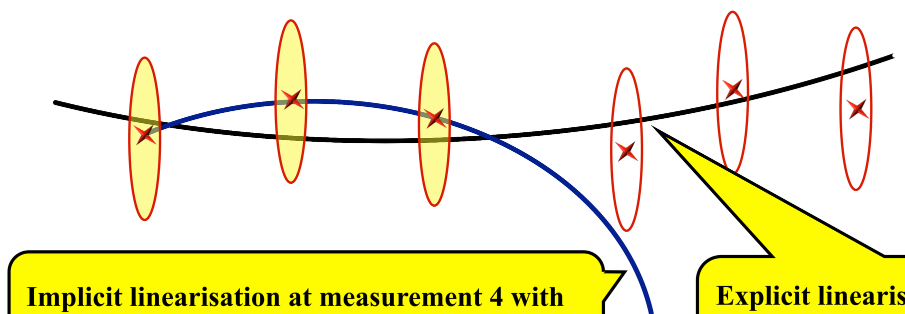
Kalman filter only operates on linear operators
⇒ there is always some linearisation performed
⇒ let's make it explicit and take a control on it

- **Common approach:** x_0=x (partially fitted track)
- **Explicit linearisation:** x_0=our best knowledge of x

$$F(x) \Rightarrow L(x) = F(x_0) + F'(x_0)(x-x_0)$$

Some nonlinear operator, i.e. track extrapolation

Corresponding linear operator actually used by the Kalman filter

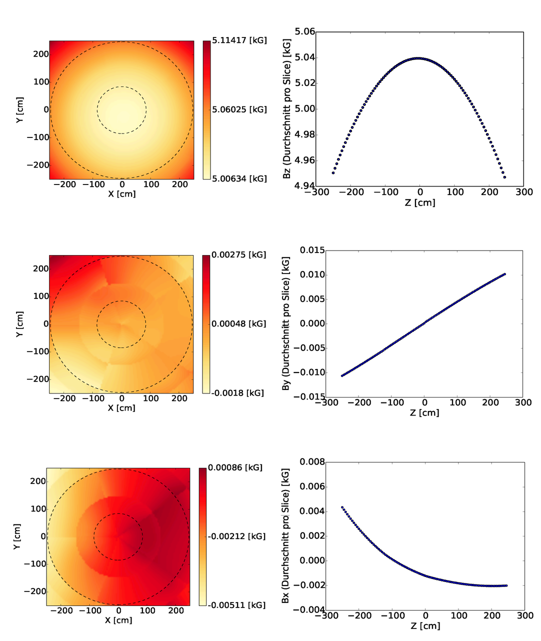


Implicit linearisation at measurement 4 with partially fitted trajectory

Explicit linearisation at measurement 4 with the best known trajectory

- **robust fit**
- **especially useful for the track finder**
  (one can choose which fitted parameters to use for the linearisation with respect to the current number of measurements on a track)
- **one can use different parameterisations for the non-linear (x_0) and for the linearised (x) trajectories**

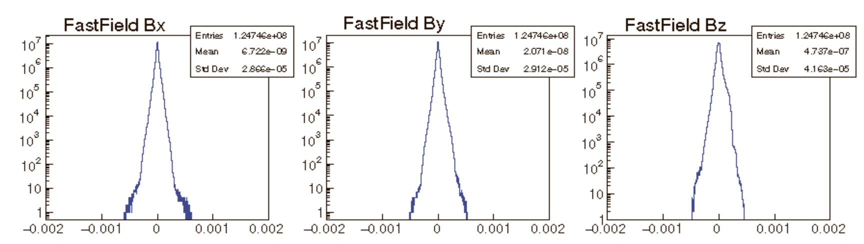
Approximation of magnetic field



- Polynomial approximation: no memory access
  => fast & GPU - friendly

$$B_{x/y/z} = (c_0 \dots c_{19})\cdot(1\ x\ y\ xx\ xy\ yy \dots zzz)^T$$

- Errors are negligible

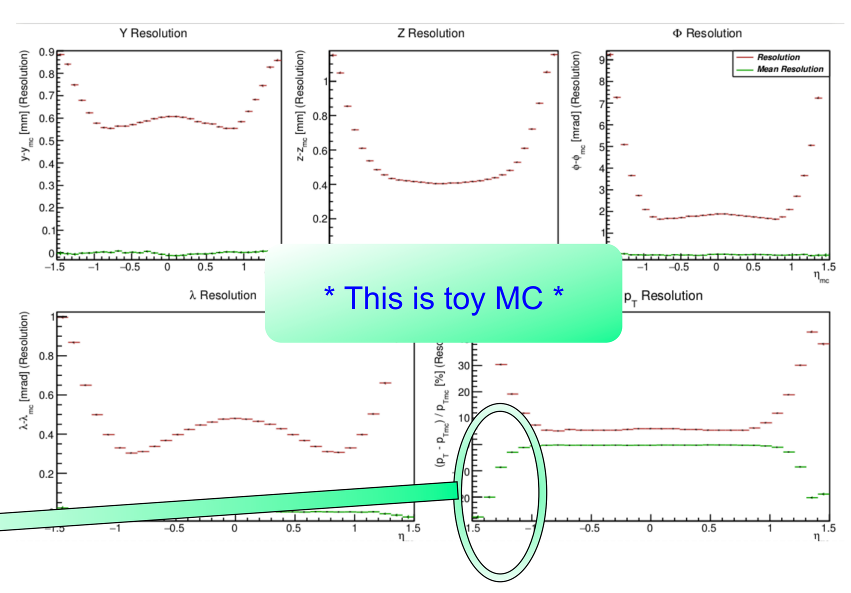


Field approximation errors relative to nominal Bz field value of 5kG

- No degradation of the fit quality

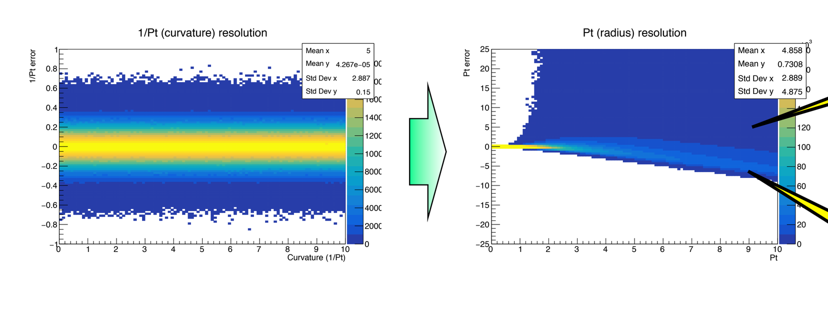
Investigation of fit features with toy MC

- **Suppress effects from detector responce, cluster finder, tracker finder, calibration**
- **Prononce effects from detector geometry, track model, magnetic field, fit mathematics**
  - ideal measurements with preferable errors
  - amplification of problems by extreme errors / angles/ track lengths



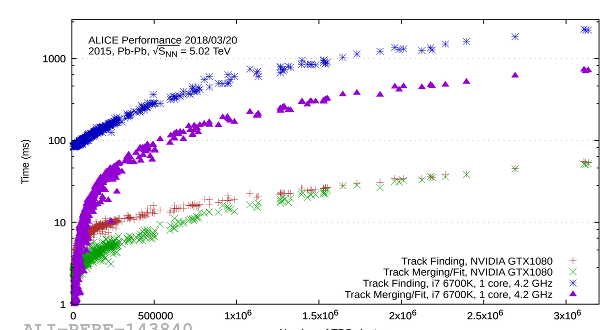
Non-linear effects in resolutions

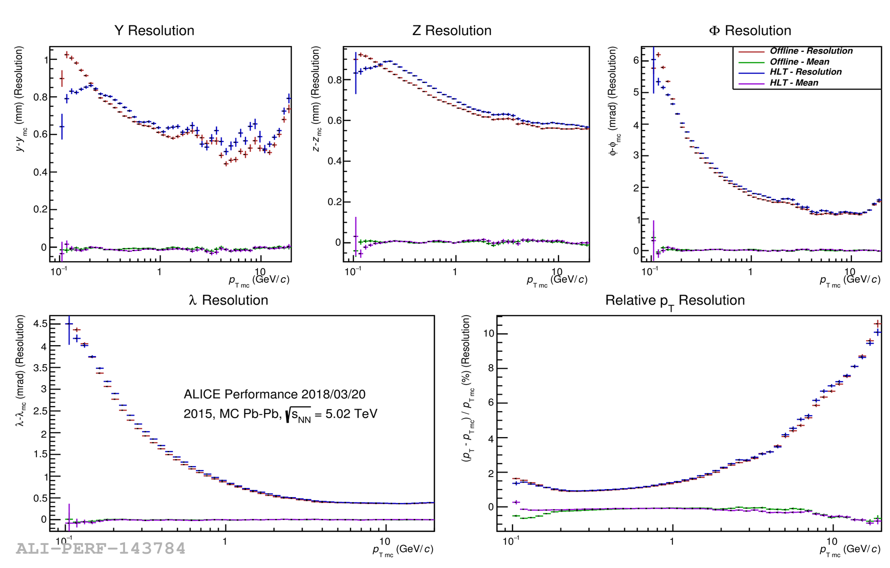
Example of how ideal gaussan errors in curvature (i.e. in measurements) are converted to biases in Pt




```
float mcpt = gRandom->Uniform(10);
float mccrv = 1/mcpt;
float crv = mccrv + gRandom->Gaus(0, 0.15);
float pt = 1./fabs(crv);
h->Fill(mcpt,pt-mcpt);
```

Performance: online speed with offline accuracy





- **Single precision mathematics and polynomial approximation of magnetic field allow effective parallel processing on GPU**
- **Difference in resolution at the very low-Pt is caused by the online track finder features**