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

Bundesministerium für Bildung und Forschung

BMBF Forschungsschwerpunkt ALICE Experiment 202

FIAS Frankfurt Institute for Advanced Studies

Sergey Gorbunov
for the ALICE collaboration
Frankfurt Institute for Advanced Studies, Goethe University Frankfurt

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

ALICE

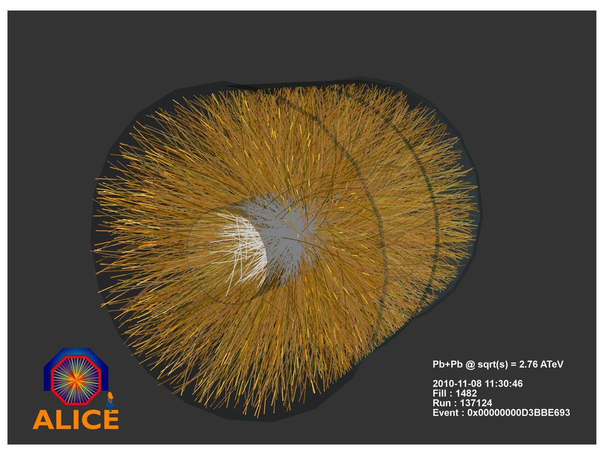
Common Online-Offline track fit aimed for the LHC Run 3

The upcoming LHC Run 3 brings new challenges for the ALICE online reconstruction which 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 online data processing.

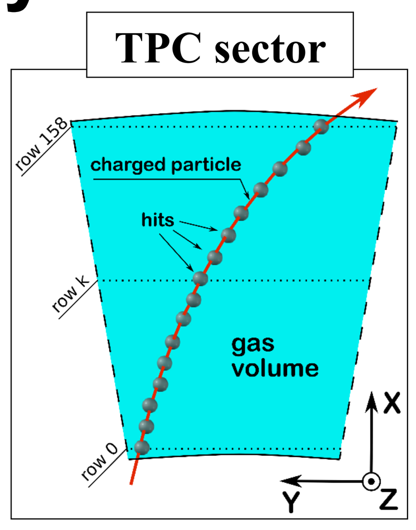
Here we present our enhancements to the track fit algorithm which is currently used in the ALICE High Level Trigger (HLT) for the online reconstruction. The algorithm is based on the Kalman filter method. The developed fitting utilities are used both at the combinatorial track finding stage and at the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

TPC detector Geometry

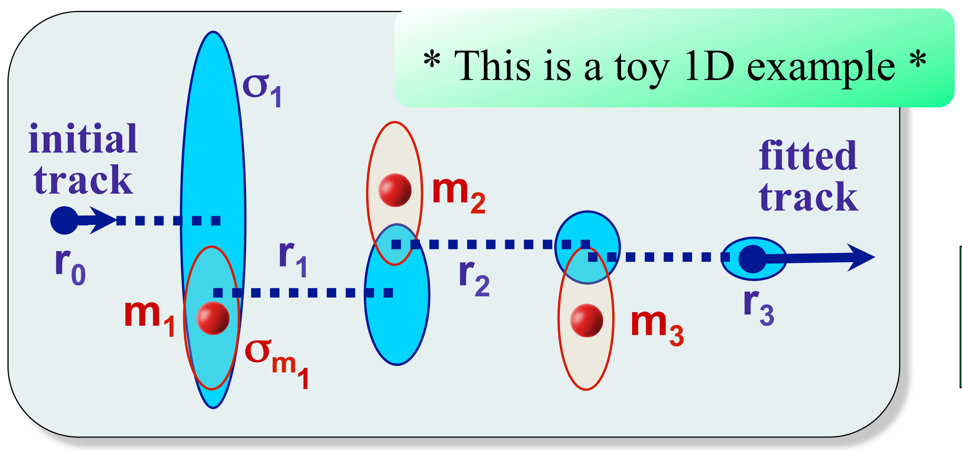


- Track fit is performed in local coordinates of the corresponding TPC sector
- Helix track parameterization:

$(x; y, z, p_y/p_t=\sin(\varphi), p_z/p_t=\tan(\theta), q/p_t)$



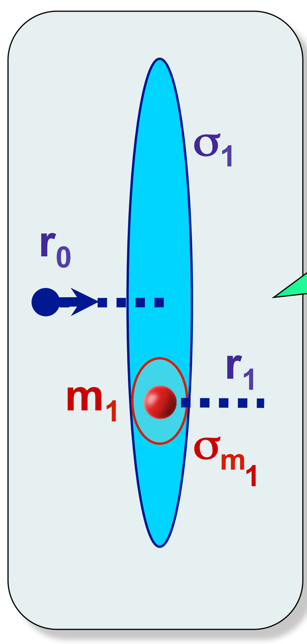
Kalman Filter: single precision for fast parallel calculations on GPU



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

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

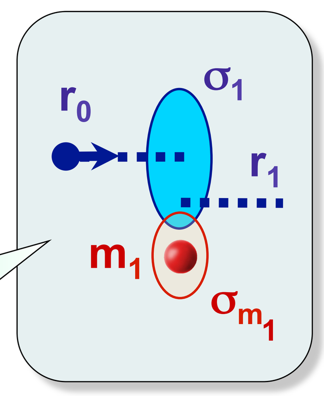
$\sigma^2_{m_k}$ vanishes when $\sigma^2_{m_k} << \sigma^2_k$



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

Large initial error: Mathematically correct, but produces rounding errors

Small initial error: No rounding errors but biased results



Separate error and parameters evaluation: Start with small initial error, but treat it in a special 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 $(\sigma_k \geq 10 \cdot \sigma_{m_k})$ treated as $(\sigma_k = \infty)$

Fast track propagation in 3D field

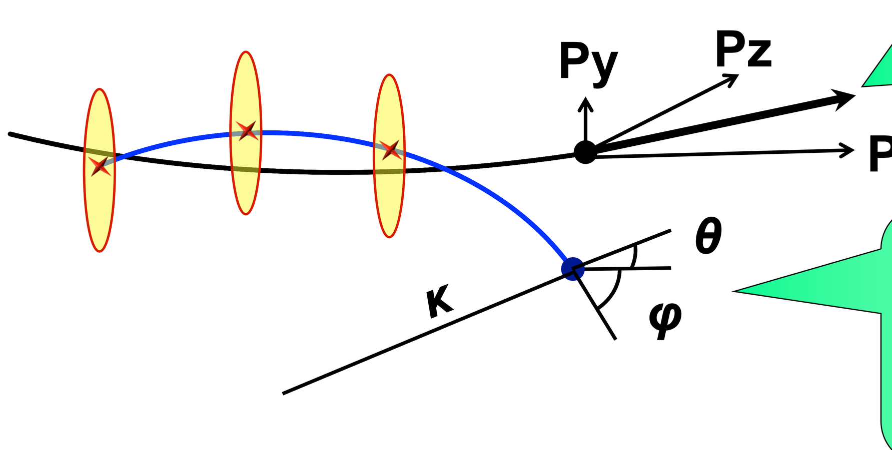
- **Accurate propagation for the linearization trajectory t_0, simple propagation for the track t**

Bx,By field components are minor and have only second-order effect => let's use the full-field propagation for t_0 and Bz-only propagation for the track t and for the covariance matrix.

- **Use different parameterisations for t_0 and t**

t_0 is only propagated and not fitted => let's use the physical parameterization which is better suited for the propagation in 3D magnetic field.

Propagation scheme



Linearization t_0
- Parameters (x, y, z, Px, Py, Pz)
- Propagated in the full field (Bx,By,Bz)

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

- **Fit in projections**

Split XY and Z components of the covariance matrix: do fit in projections, but linearization in space.

Kalman Filter : explicit linearization

As the Kalman filter operates on linear models only, there is always some linearization present.
⇒ Let's make it explicit and take a control on it.

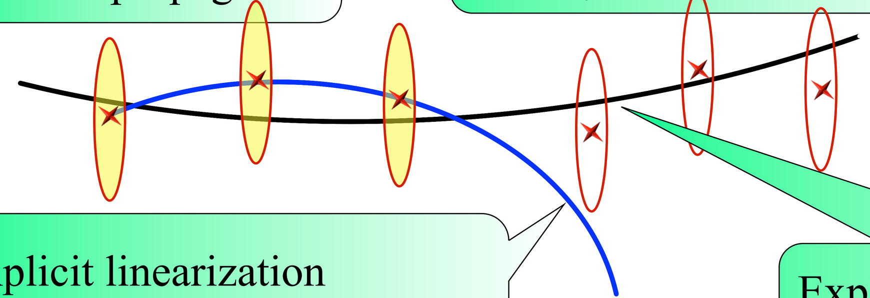
$F(x) \Rightarrow L(x) = F(x_0) + F'(x_0)(x-x_0)$

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

Nonlinear operator, i.e. track propagation

Corresponding linear operator used by the Kalman filter

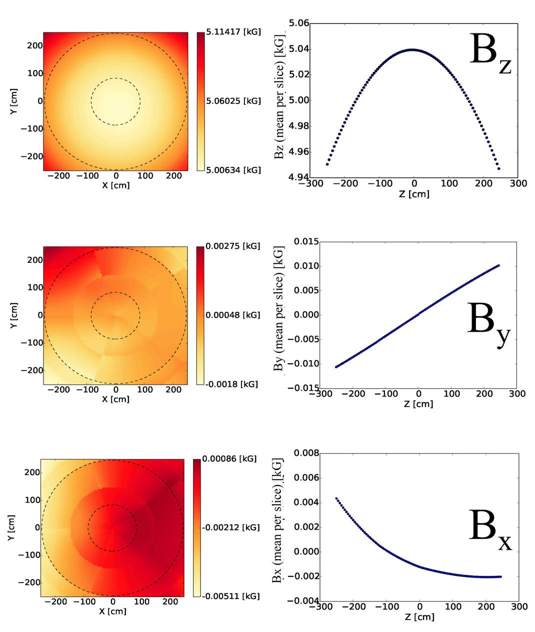
- Robust fit
- Precision tool for the track finder



Implicit linearization at partially fitted trajectory

Explicit linearization at the best known trajectory

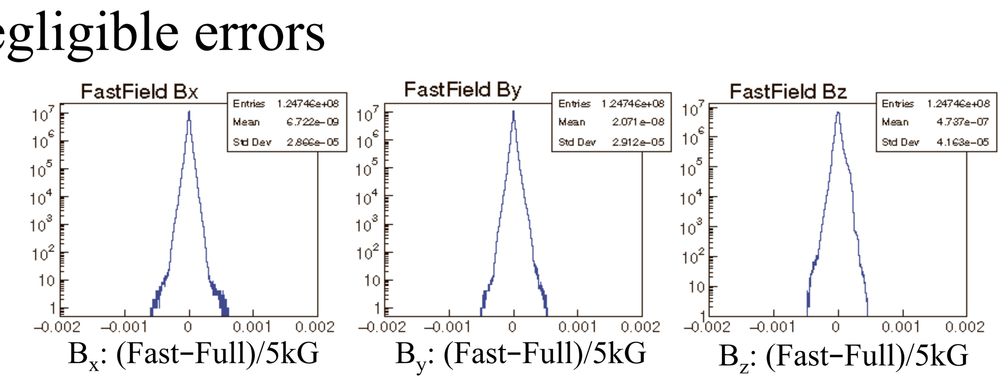
Approximation of magnetic field



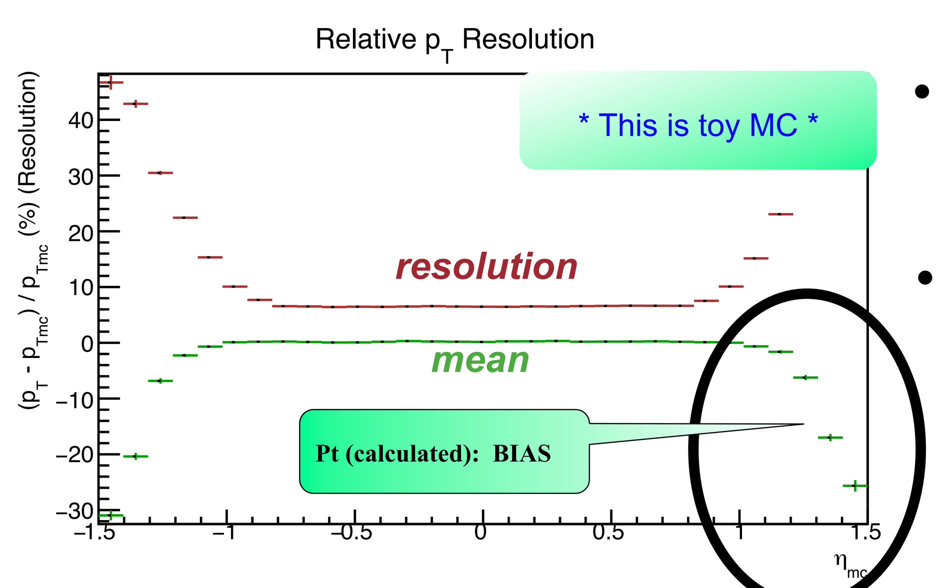
- Polynomial approximation: no main memory access => fast & GPU-friendly

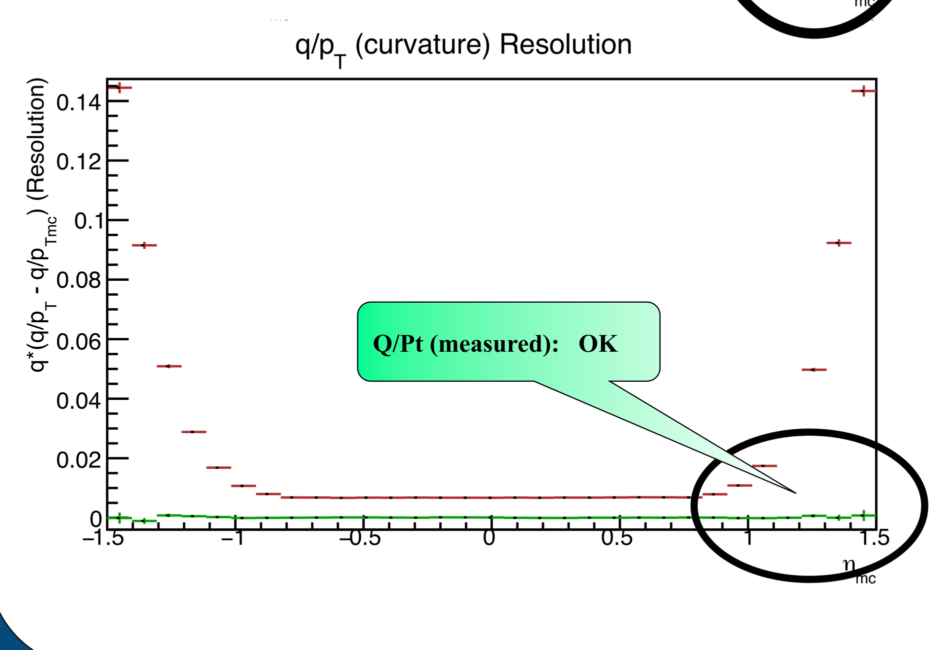
$B_{x/y/z} = (c_0 \dots c_{19}) \cdot (1\ x\ y\ xx\ xy\ yy \dots zzz)^T$

- Negligible errors



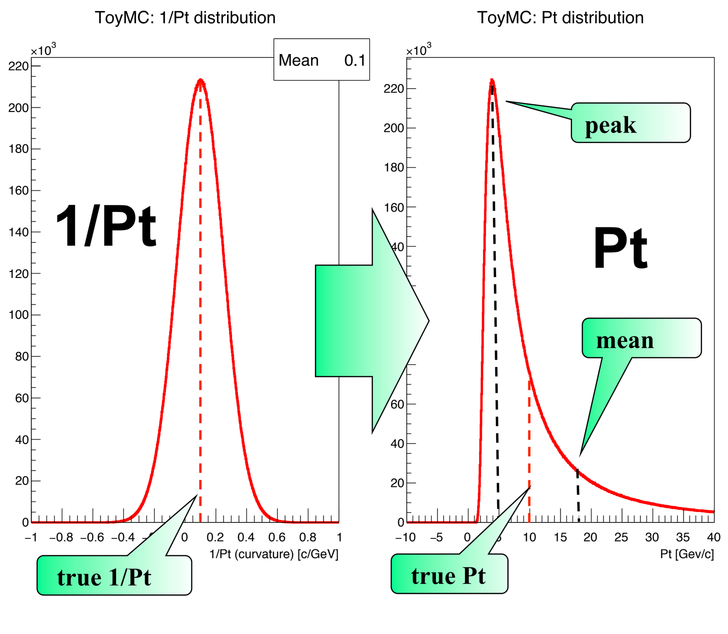
Investigation of features with toy MC



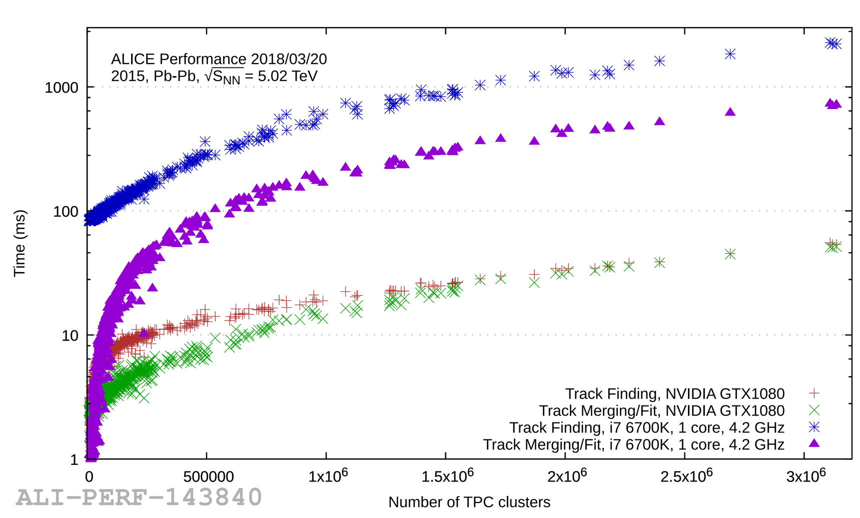


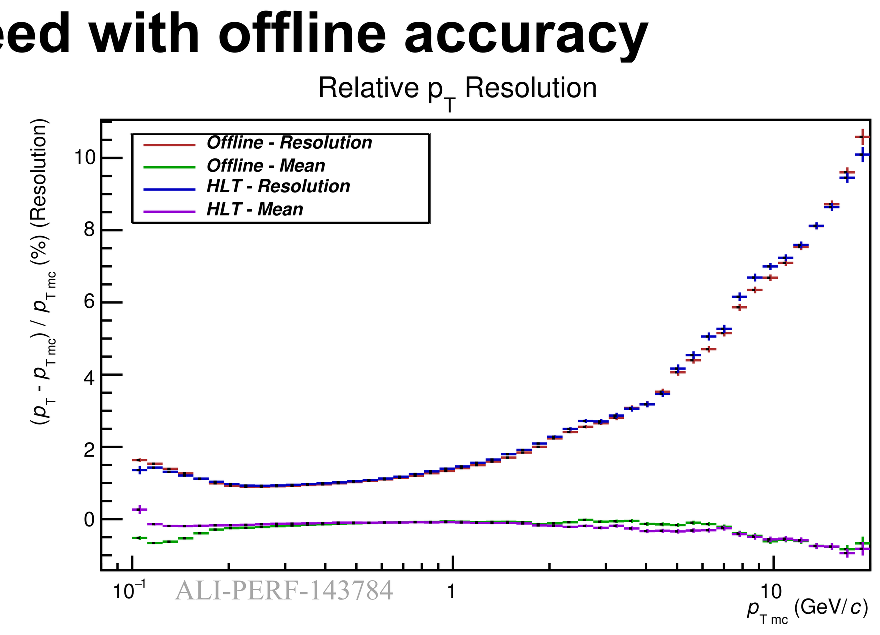
- Suppress effects from readout, cluster finder, tracker, calibration
- Amplify effects from geometry, track model, magnetic field, fit mathematics

Non-linear effects



Results: Online speed with offline accuracy





Summary

The developed extensions to the Kalman filter and approximation of magnetic field allow the ALICE online track fit to effectively run on GPU. The fit matches offline accuracy requirements keeping the online speed and designed for the future use in the LHC Run 3.

#423