INTERNATIONAL MAX PLANCK RESEARCH SCHOOL FOR PRECISION TESTS OF FUNDAMENTAL SYMMETRIES

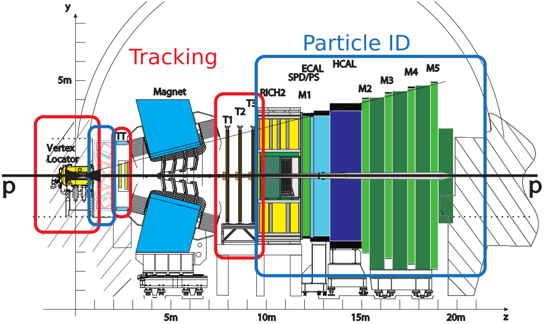
Fast, Parallel and Parametrized Kalman Filters for LHCb upgrade

Simon Stemmle

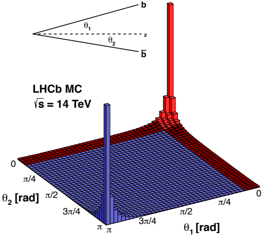
On behalf of the LHCb collaboration
Physikalisches Institut Heidelberg

CTD/WIT 06.-09.03.2017

The LHCb detector



[JINST 3 (2008) S08005]



[LHCb, LHCb webpage]

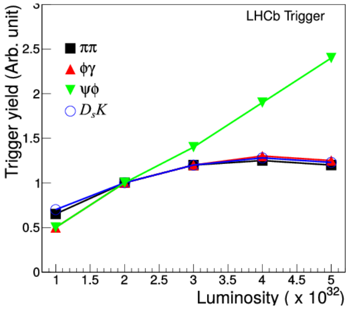
Key features

- Single arm, forward region detector at the LHC
- Main focus is the study of b- and c-quark systems: e.g. new physics in *CP*-violation and rare decays
- Excellent vertex/decay time ($\sim$ 45fs) and momentum ($0.5 - 1\%$) resolution
  [Int. J. Mod. Phys. A 30, 1530022 (2015)]

The LHCb upgrade

Motivation

- Many physics results will still be statistically limited at the end of Run II (2018)
- Need to run at a much higher Luminosity:
  $4 \times 10^{32} \mathrm{cm}^{-2}\mathrm{s}^{-1}$ (Run I,II) $\rightarrow$ $2 \times 10^{33} \mathrm{cm}^{-2}\mathrm{s}^{-1}$ (Run III)



[JINST 9 (2014) C10026]

Currently:

- L0 hardware trigger: 30 MHz $\rightarrow$ 1 MHz

Upgrade:

- Detector readout @ 30 MHz
- Removed hardware trigger. Software trigger with full reconstruction

Upgrade (tracking): Higher granularity detectors with fast read out

Upstream Tracker

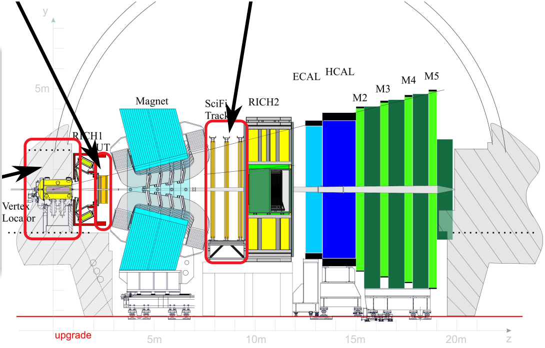
- Silicon strip technology ($95 - 180\,\mu$m pitch)
- 4 tracking layers ($0^\circ / +5^\circ / -5^\circ / 0^\circ$)

Scintillating Fiber tracker

- Scintillating fibers ($250\,\mu$m diameter)
- 12 tracking layers ($\sim 5 \times 6\,\mathrm{m}^2$) $3\times(0^\circ / +5^\circ / -5^\circ / 0^\circ)$

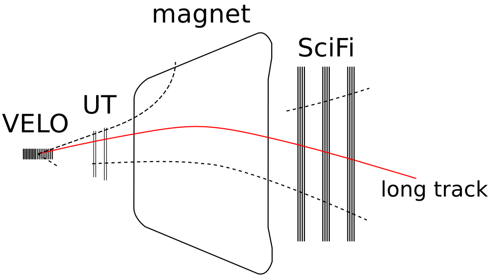
VErtex LOcator

- Silicon pixel sensors ($55 \times 55\,\mu\mathrm{m}^2$)
- 26 tracking layers
- 5.1 mm away from the beam



[CERN-LHCC-2013-021]
[CERN-LHCC-2014-001]

Track reconstruction in the fast (first) trigger stage



- Reconstruct all VELO tracks
- Expand with UT hits
- Find matching hits in SciFi (require $p_T > 400\,\text{MeV}$)
- Only these tracks are further reconstructed

[LHCb-PUB-2017-005]

Challenges

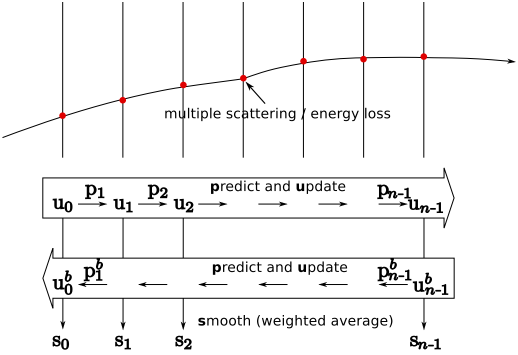
- The 30 MHz readout and the increased luminosity make a significant speed-up in the online reconstruction necessary (5 ms per event, factor 2-3 missing)
- Currently around 60% of the track reconstruction time is spent on the track fitting (Kalman filter)

Kalman filter in the upgrade

Principles of a Kalman filter

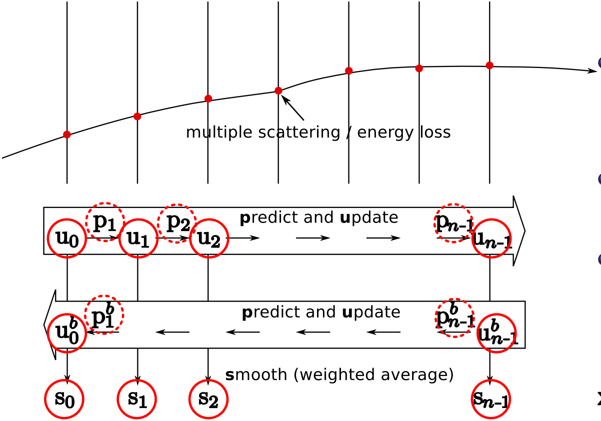
- Sequential adding of new information (hits) to obtain an optimal track estimate
- The strategy at LHCb is:

  Filtering Forward + Filtering Backward + Smoothing



What are the time critical parts?

1. Costly matrix algebra

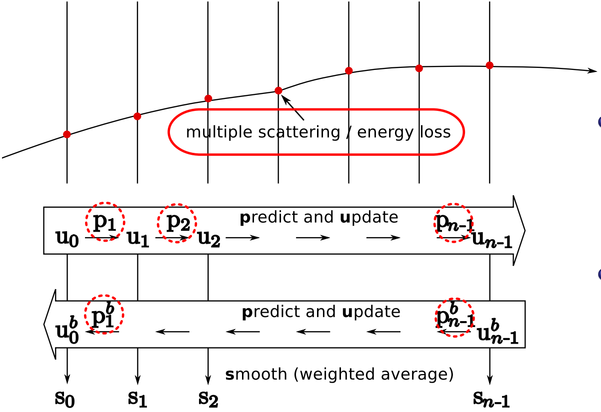


- mainly 5×5 matrix multiplications and inversions
- all Kalman filter steps are affected
- necessary and irreducible calculations

$\mathbf{x} = (x, y, t_x, t_y, \frac{q}{p})$

→ Parallelize these calculations

2. Costly detector modelling



- Magnetic field: Extrapolation using Runge-Kutta differential equation solving method
- Material look-up for multiple scattering and energy loss

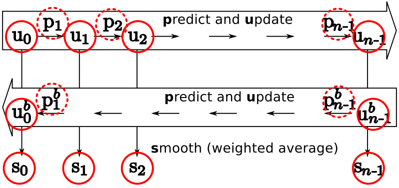
→ Find simple parametrizations for both

Parallelized Kalman Filter

- A Kalman Filter is fundamentally non-parallel
  - Sequential predicting and updating for every hit (forward and backward)
  - Only the smoother step is independent for all hits
- **But:** Between different tracks the process is:

1. Independent → highly **parallelizable**
2. Always the same (matrix operations) → even **vectorizable (SIMD)**

For M tracks:



M at a time

M at a time

$\sum_{i=1}^{M} n_i$ at a time

The scheduler - Flexible vectorisation

- What we want:
  - Perform filter steps for several tracks at the same time on a single core
  - Avoid empty vector units although tracks have different number of hits

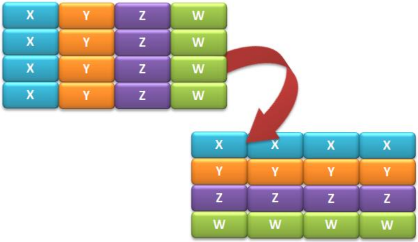
- The scheduler
  - Static scheduler for available cores and vector units
  - Same schedule for forward/backward and smoothing
    $\rightarrow$ Preserves data structure ordering for posterior smoothing
- Cross architecture
  - Configurable vector width at compile time
  - Precision can also be switched

```
it    in   out  act  vector (#particle-#hit)
#540: 0000 0001 1111 { 112-9 80-11 81-11 113-10 }
#541: 0001 1110 1111 { 112-10 80-12 81-12 79-3 }
#542: 1110 0000 1111 { 107-2 109-1 108-2 79-4 }
#543: 0000 0000 1111 { 107-3 109-2 108-3 79-5 }
#544: 0000 0000 1111 { 107-4 109-3 108-4 79-6 }
#545: 0000 0000 1111 { 107-5 109-4 108-5 79-7 }
#546: 0000 0000 1111 { 107-6 109-5 108-6 79-8 }
#547: 0000 0000 1111 { 107-7 109-6 108-7 79-9 }
#548: 0000 0000 1111 { 107-8 109-7 108-8 79-10 }
#549: 0000 0000 1111 { 107-9 109-8 108-9 79-11 }
```

Data structures

Array Of Structures Of Arrays

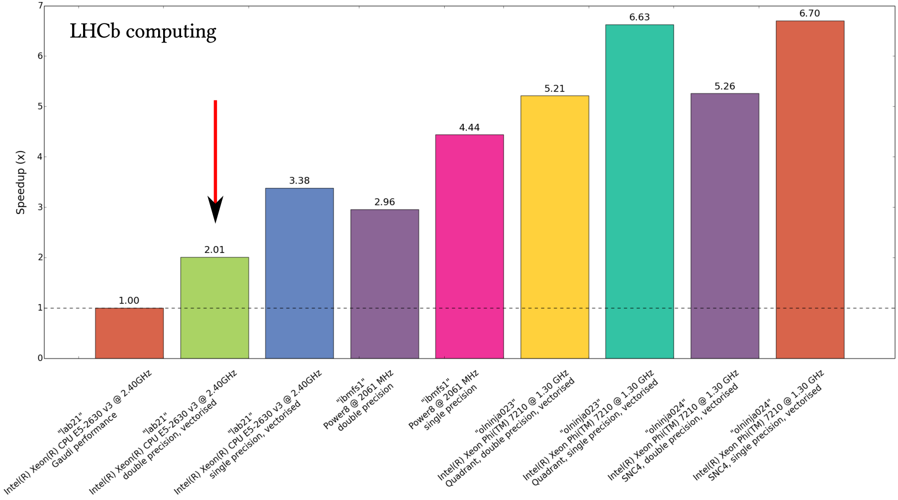
- Data is correctly aligned for the use of SIMD processors
- Vector width is represented at compile time
- E.g.: State information (x, covariance, χ^2) aligned in groups of 4



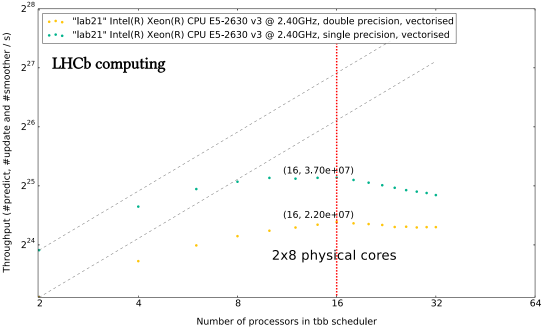
$$\begin{pmatrix} x_0 & x_1 & x_2 & x_3 \\ y_0 & y_1 & y_2 & y_3 \\ tx_0 & tx_1 & tx_2 & tx_3 \\ ty_0 & ty_1 & ty_2 & ty_3 \\ \frac{q}{p}_0 & \frac{q}{p}_1 & \frac{q}{p}_2 & \frac{q}{p}_3 \\ c_{0,0} & c_{1,0} & c_{2,0} & c_{3,0} \\ c_{0,1} & c_{1,1} & c_{2,1} & c_{3,1} \\ \vdots & \vdots & \vdots & \vdots \\ c_{0,14} & c_{1,14} & c_{2,14} & c_{1,14} \\ {\chi^2}_0 & {\chi^2}_1 & {\chi^2}_2 & {\chi^2}_3 \end{pmatrix}$$

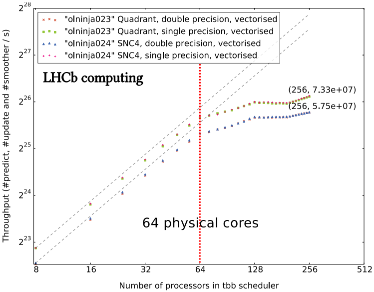
Performance

Speed gain of "matrix algebra" (predicting, updating, smoothing)



Scalability of the Kalman Filter





Conclusion - Parallelized Kalman Filter

- Vectorised the Kalman Filter for several fits in parallel
- Flexible across different architectures
- Around 2$\times$ speed up for the "matrix algebra" part
- Integrated in the full Kalman framework:
  - Speed up of around $10 - 20\%$
  - Track extrapolation and material correction might also be parallelized in the future

More information: LHCb-TALK-2016-372

Parametrized Kalman Filter

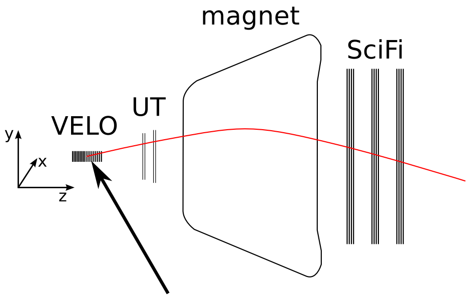
Basic idea: use simple parametrizations..

- ..for the extrapolation from one detector layer to the next
- ..for the noise added due to multiple scattering

General remarks

- Energy loss is not explicitly taken into account, but:
  - Each extrapolation is tuned to match initial momentum at the vertex
  - The extrapolation is tuned to MC truth (with energy loss)
    $\rightarrow$ Energy loss is implicitly contained in the parametrization of the extrapolation
- The Jacobian matrix can be easily obtained by simple derivatives
- Ongoing work, still very preliminary

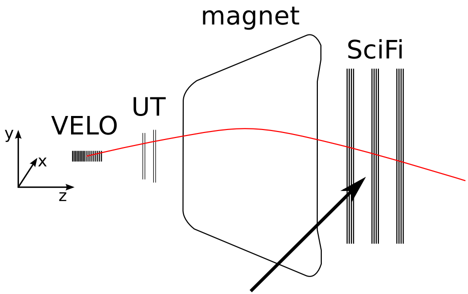
Simplified parametrizations



Inside the VELO (weak mag. field)

- Straight line prediction for y
- First order correction in q/p for x (effect is small)

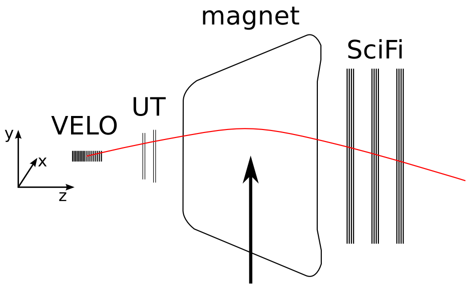
Simplified parametrizations



Inside the SciFi stations (medium mag. field)

- Empirical parametrization depending on q/p and y for prediction
- e.g.: $t_x' = t_x + par_1 \frac{q}{p} + par_2 \left(\frac{q}{p}\right)^3 + par_3 y^2 \frac{q}{p} \left|\frac{q}{p}\right|$

Simplified parametrizations



Between VELO and SciFi stations (strong mag. field)

- More sophisticated parametrization based on:

$$\Delta p_x = p\left(\frac{t_{x,T}}{\sqrt{1+t_{x,T}^2+t_{y,T}^2}} - \frac{t_{x,V}}{\sqrt{1+t_{x,V}^2+t_{y,V}^2}}\right) = q\int |d\mathbf{l}\times\mathbf{B}|$$

$$x_T = x_V + (z_{mag}-z_V)t_{x,V} + (z_T - z_{mag})t_{x,T}$$

- $\int |d\mathbf{l}\times\mathbf{B}|$ and z_{mag} are parametrized as functions of the state parameters at the last VELO hit

Simplified parametrizations

Noise matrix - Material

- Mostly linear parametrization in q/p; no other dependency
- Correlations between $x - t_x$ and $y - t_y$ are also tuned for each extrapolation
- No explicit energy loss

Parameter tuning

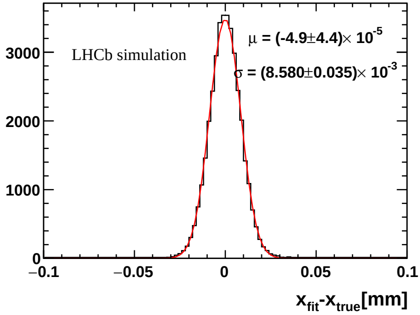
- For each extrapolation step we need (∼O(4-20)) parameters
- Parameters are tuned by likelihood fits of a Gaussian to e.g. $(f_{t_x}(x) - t_x^{true})/\sigma_{t_x}$ using MC truth

Comparison with the default full Kalman Filter

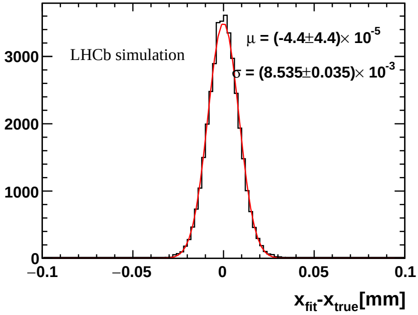
- First trigger stage: Only tracks with $p_T > 400\,\mathrm{MeV}$ and hits in the SciFi

  $\rightarrow$ Mainly relatively high momentum tracks starting near the primary vertex
- Compare states and covariances at the first VELO hit

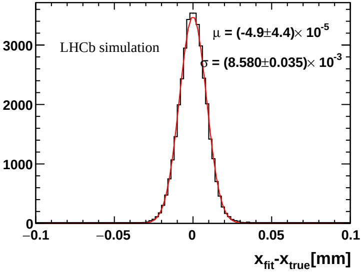
Parametrized

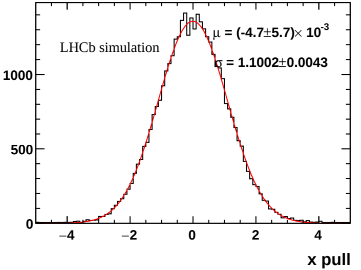


Default

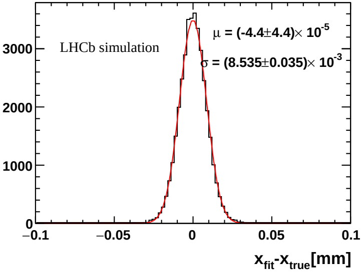


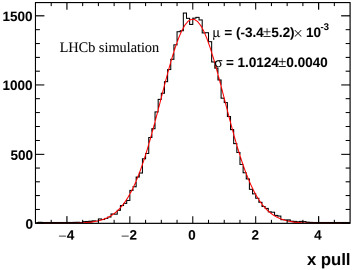
Parametrized



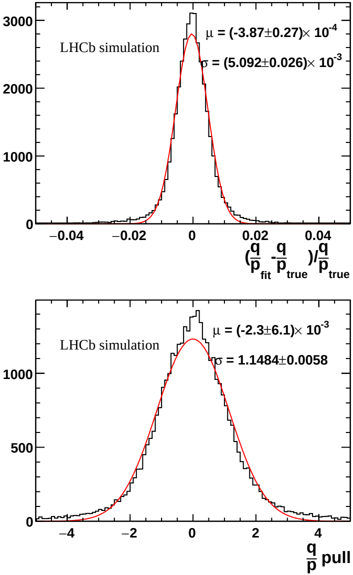


Default

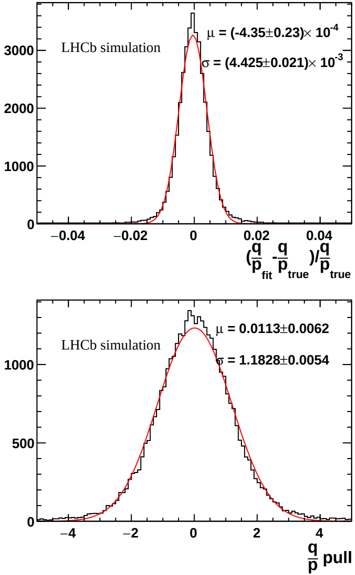




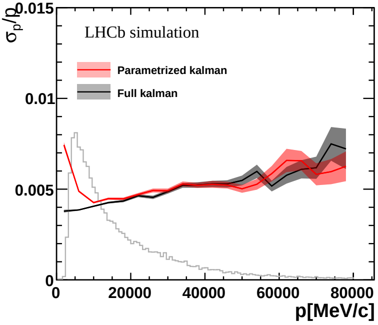
Parametrized

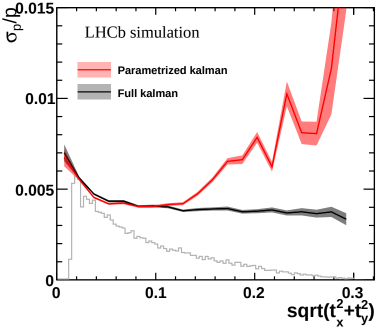


Default



Momentum resolution



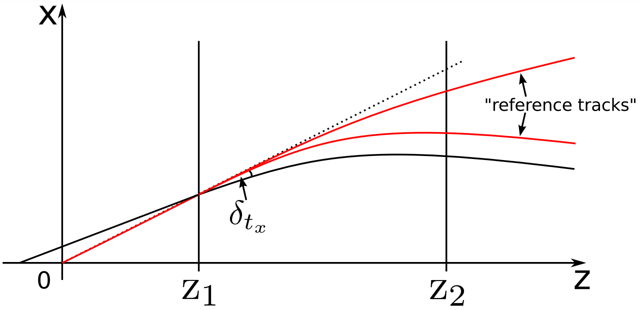


- At low momentum: wrongly modelled multiple scattering
- Higher order corrections in t_x, t_y seem necessary. Two solutions:
  - Use alternative parametrization for the step VELO->SciFi stations (next slide)
  - Use the full default Kalman filter for these tracks

Alternative VELO↔SciFi stations prediction

See talk by Pierre Billoir for details. Basic Idea:

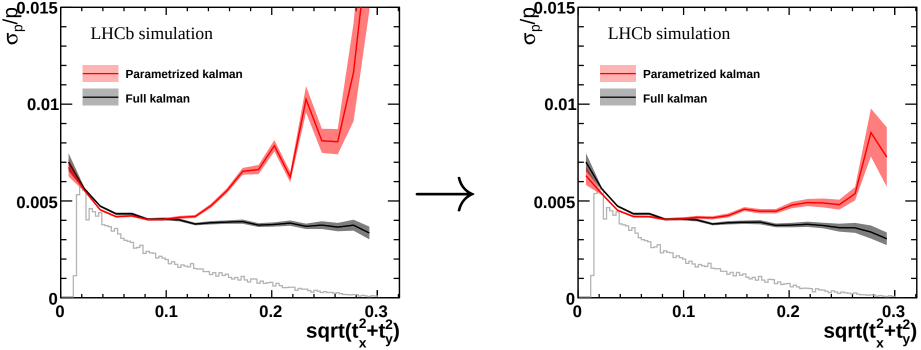
- Use primary (x,y=0 at z=0) tracks as "reference":
  - For them the extrapolation is a expansion in $\frac{q}{p}$ (4th order)
  - Using coefficients that are tabulated as a function of x, y
- Perform a expansion in the deviation from these tracks (δ_{t_x} and δ_{t_y}) for the correction of the coefficients of the $\frac{q}{p}$ expansion



Alternative VELO↔SciFi stations prediction

For this purpose

- Additional overall $\frac{q}{p}$ correction necessary to account for energy loss
- Tuned for $p > 3\,\mathrm{GeV}$ → apply this cut (loose 1-2%)



Speed up and conclusion - Parametrized Kalman Filter

Timing performance

- In the current framework, $30 - 50\%$ is spent on extrapolation/material effects
- This can be sped up by a factor $\sim$ 5-10 using the simplified parametrizations
- Not yet implemented in the current framework

Conclusion: Study is still in development but seems promising

- For a large fraction of tracks the results are comparable to the full Kalman filter
- Even better: We can predict for which tracks it works and for which not

Summary

Two strategies for speeding up the LHCb Kalman filter were presented:

- The parallelization of the "matrix algebra" part using SIMD
  - Yields a factor 2 in timing for these methods
  - Integrated in the full framework: speed gain of 10-20%
- Using simple parametrizations for magnetic field and material effects
  - Promising for most of the tracks in the first trigger stage
  - Speed gain of factor 5-10 for the respective part

Thank you for your attention