T-TRACK EXTRAPOLATION MODEL AND PROSPECTS FOR RECONSTRUCTION AT HLT1

A. Oyanguren, B. Jashal, J. Zhuo, V. Kholoimov, V. Svintozelskyi,

Instituto de Física Corpuscular (IFIC), CSIC (ES), Univ. of Valencia
Taras Shevchenko National University of Kyiv

Overview

01 **Introduction**

LHCb High-level trigger system

02 **T-Track Extrapolation model**

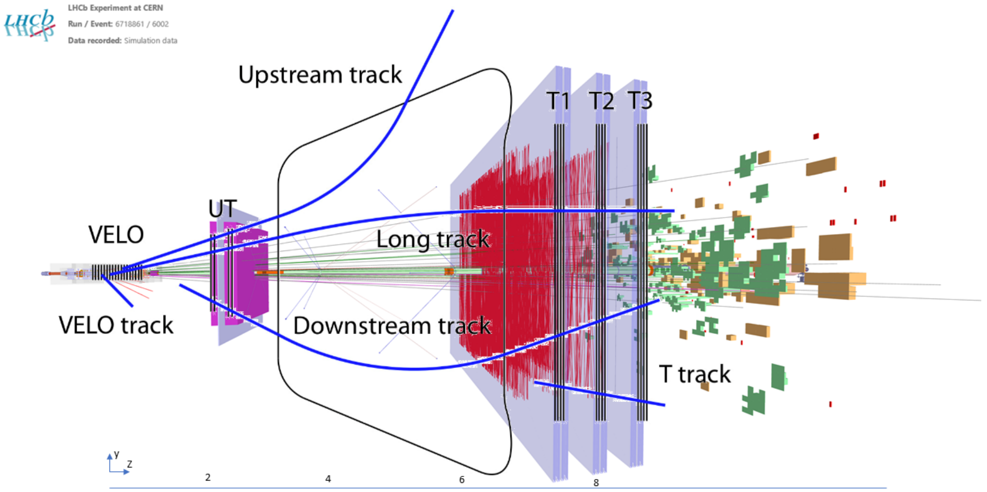
And why it's so tricky

03 **Model applications**

We have a model. What's now?

04 **Simpler fit**

Will it be accurate?



01

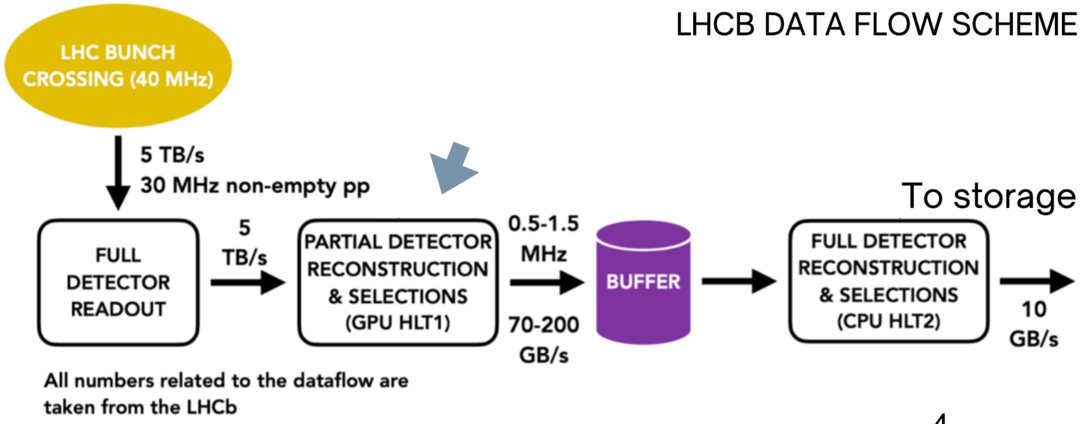
Introduction

LHCb High-level trigger system

HIGH-LEVEL TRIGGER SYSTEM

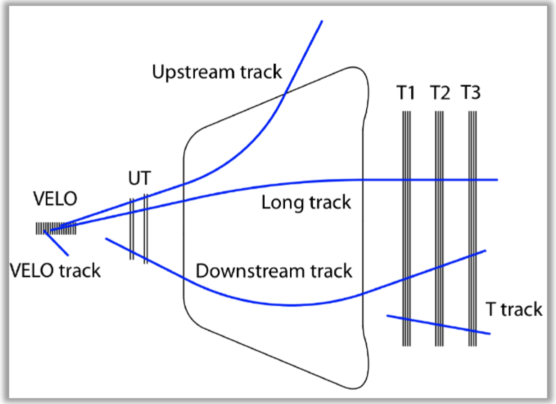
- LHC provides 40 million pp collisions per second (approx 5 TB/s of data)
- We're unable to save all the data from an economic point of view
- One needs to "select" only interesting events - done with high-level trigger system
- HLT1 & HLT2 - software triggers, able to reduce data flow down to 10GB/s

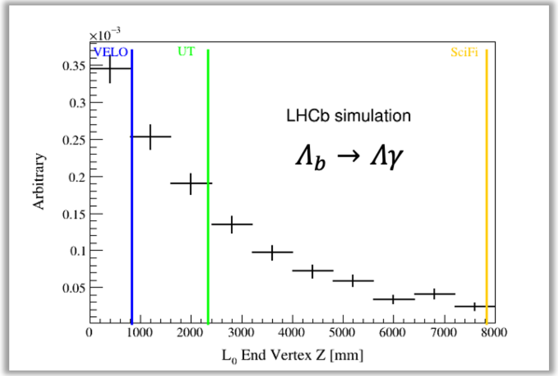
LHCB DATA FLOW SCHEME



Long-Lived particles

- Large fraction decays outside of VELO:
  For $\Lambda b \rightarrow \Lambda\gamma$ channel, the distribution of Λ decay vertices z leads to:
    - 51% - daughter hits UT & SciFi (downstream tracks)
    - 37% - daughter hits SciFi only (t-tracks)
- To reconstruct most of the Λ (as well as other LLPs), one needs downstream & t-track reconstruction and vertexing algorithms





Global Objective

Create an effective selection with TT at HLT1

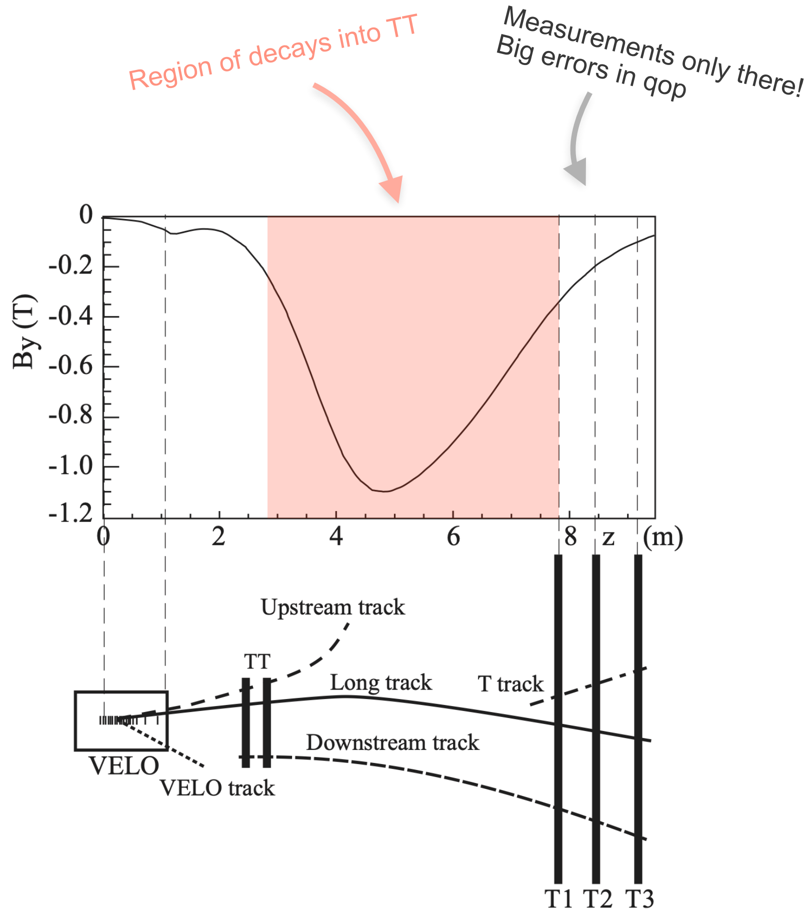
What's the problem?

A complex magnet field doesn't allow simple polynomial T-track extrapolation

Significant track momentum errors

Any existing solutions?

HLT2 uses RK for track extrapolation, which is completely unfeasible for HLT1 (timing constraint)



02

T-Track Extrapolation model

And why it's so tricky

T-Track Extrapolation model

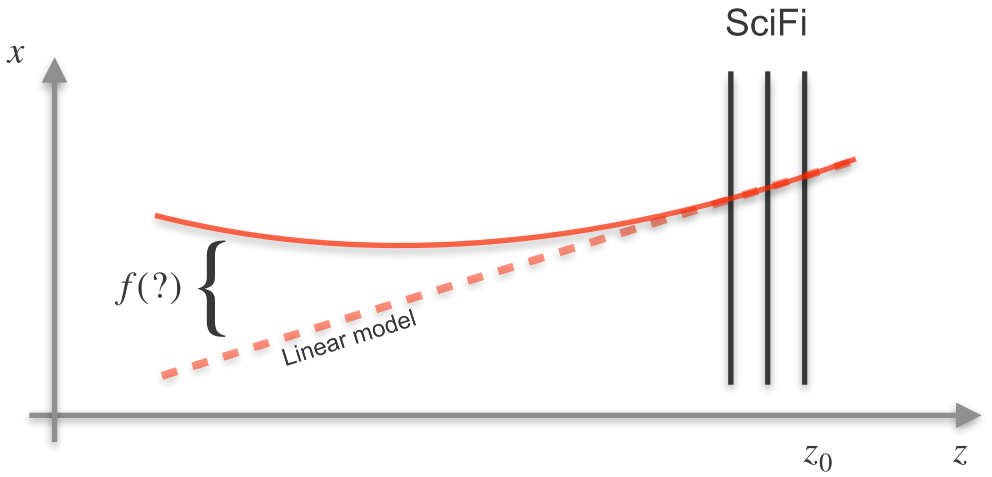
T-Track tracking model may be written as:

$$\begin{cases} x(z) = x_0 + tx(z - z_0) + f(?) \\ y(z) = y_0 + ty(z - z_0) \end{cases}$$

Obviously, f should satisfy:

$$f(\frac{q}{p} = 0) = 0$$

$$f(z = z_0) = 0$$



T-Track model with constraints

T-Track tracking model may be written as:

$$\begin{cases} x(z) = x_0 + tx(z - z_0) + f(?) \\ y(z) = y_0 + ty(z - z_0) \end{cases}$$

Obviously, f should satisfy:

$$f(\frac{q}{p} = 0) = 0$$

$$f(z = z_0) = 0$$

The unknown function f may be redefined as:

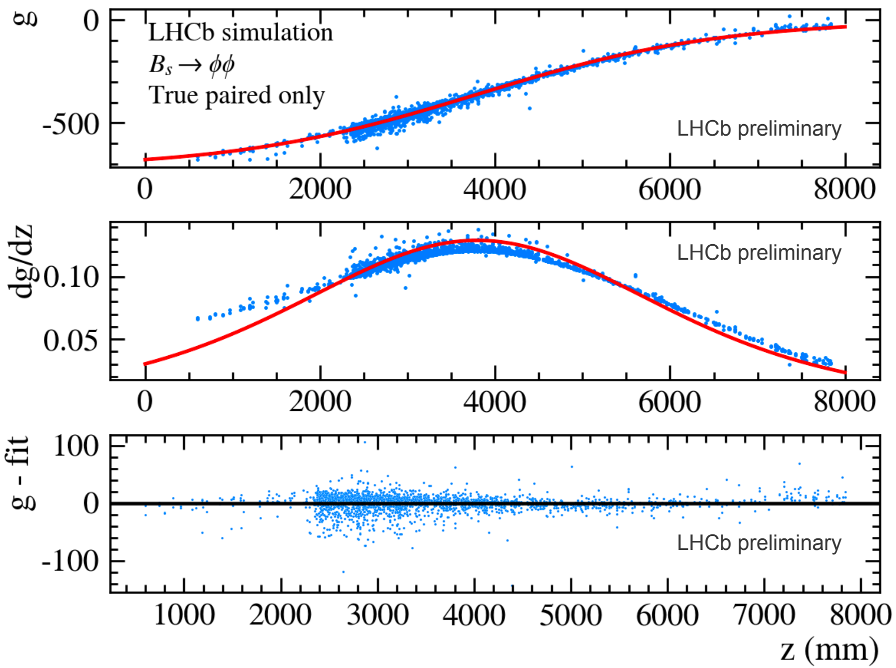
$$f(?) = \frac{q}{p}(z - z_0)g(?)$$

What is $g(z, \frac{q}{p}, \ldots)$?

g and its derivative may be found from MC data:

$$g = \frac{x_{ov} - x_{sf} - tx_{sf}(z_{ov} - z_{sf})}{\frac{q}{p}(z_{ov} - z_{sf})} \qquad \frac{dg}{dz} = \frac{tx_{ov} - tx_{sf} - \frac{q}{p}g}{\frac{q}{p}(z_{ov} - z_{sf})}$$

T-Track model for true paired only



Fitted with mirrored sigmoid:

$$g = \frac{a}{1 + e^{b(z-c)}}$$

$$\lim_{z\to+\infty} g(z) = 0 \qquad \lim_{z\to-\infty} g(z) = a$$

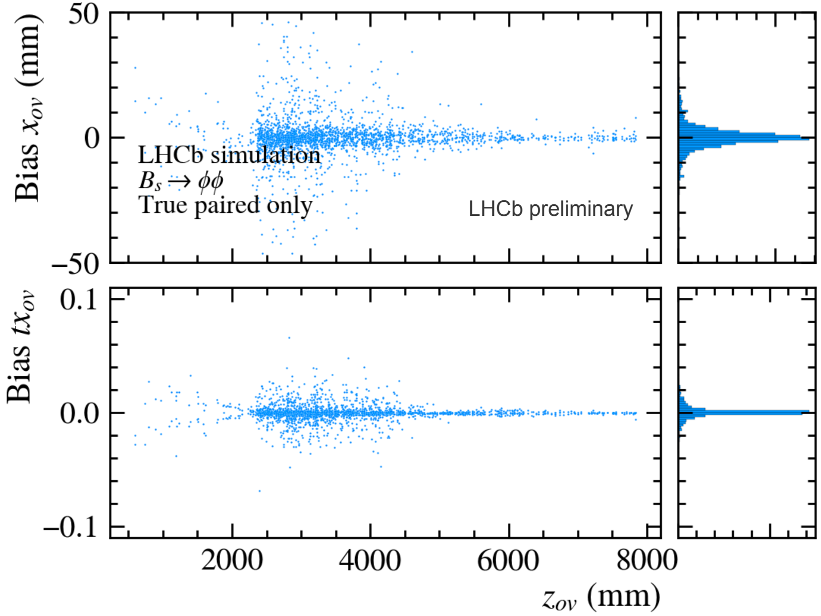
Constant slope before magnet

Still, some deviations exist, try correction

$$g = g_z(z) \cdot g_{qop}\left(\frac{q}{p}\right) \cdot g_{x_0}(x_0)$$

Should be near 1

T-Track model global fit



The fitting procedure was repeated, but this time over all variables at once:

$$g = \frac{a}{1 + e^{b(z-c)}}(e^{d(x_0-e)} + e^{-d(x_0+e)})(e^{k(qop-l)} + e^{-k(qop+l)})$$

Bias was estimated using true sf variables

Final track model:

$$x(z) = x_0 + tx(z - z_0) + \frac{q}{p}(z - z_0)g$$

$$y(z) = y_0 + ty(z - z_0)$$

03

Model applications

We have a model. What's now?

Better slopes with the model

T-Track tracking model allows better slope computation:

$$tx(z_{ov}) = tx_{sf} + \frac{q}{p}g(z_{ov}, x_{sf}, \frac{q}{p}) + \frac{q}{p}(z_{ov} - z_{sf})\frac{dg}{dz}(z_{ov}, x_{sf}, \frac{q}{p})$$

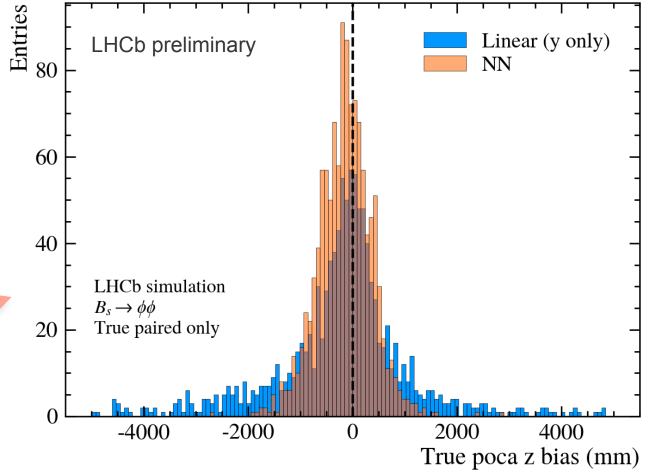
No z_{ov}

Numerical $DOCA(z_{ov})$ minimization fails because of bad SciFi qop accuracy

Poca z may be estimated from two-layer NN with inputs:

- $y^A_{sf} - y^B_{sf}$
- $ty^A_{sf} - ty^B_{sf}$
- $x^A_{sf} - x^B_{sf}$
- $tx^A_{sf} - tx^B_{sf}$
- qop^A
- qop^B

The size of hidden layers - 14; 5

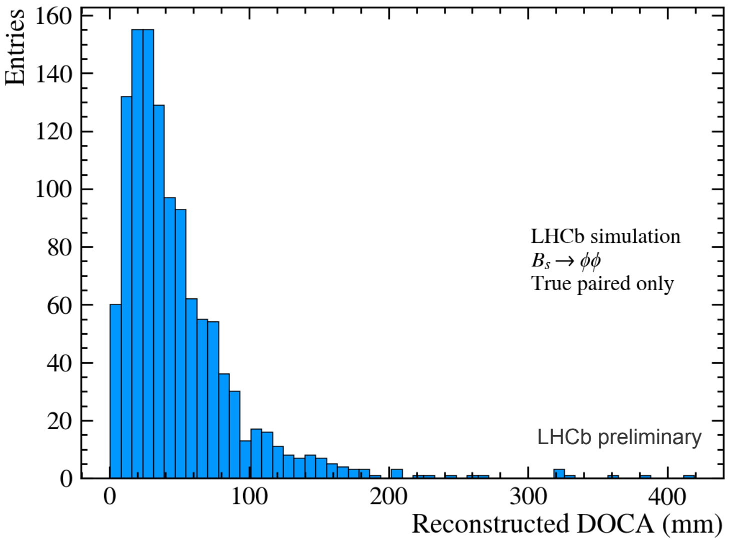


MAY WE USE IT FOR SELECTION?

Unlikely

- The track model allows better DOCA computation, but otherwise, it doesn't add any new variables
- With existing ones, there is no way to make an effective selection, at least with NN classification

However, the track model allows vertex fit - an effective tool in ghost reduction

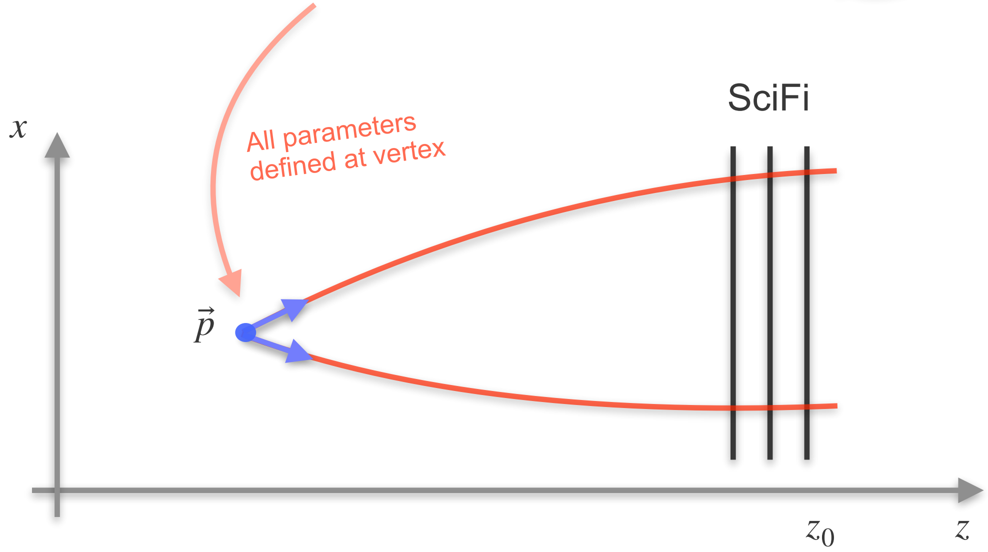


Mathematics of fitting – Part 1

Generally, there are 9 unknown parameters for a two-track vertex:

$$\vec{p} = (\underbrace{x_{ov}, y_{ov}, z_{ov}}_{\text{Vertex}}, \underbrace{tx^A_{ov}, ty^A_{ov}, qop^A}_{\text{Track A}}, \underbrace{tx^B_{ov}, ty^B_{ov}, qop^B}_{\text{Track B}})$$

Improve SciFi estimation of qop

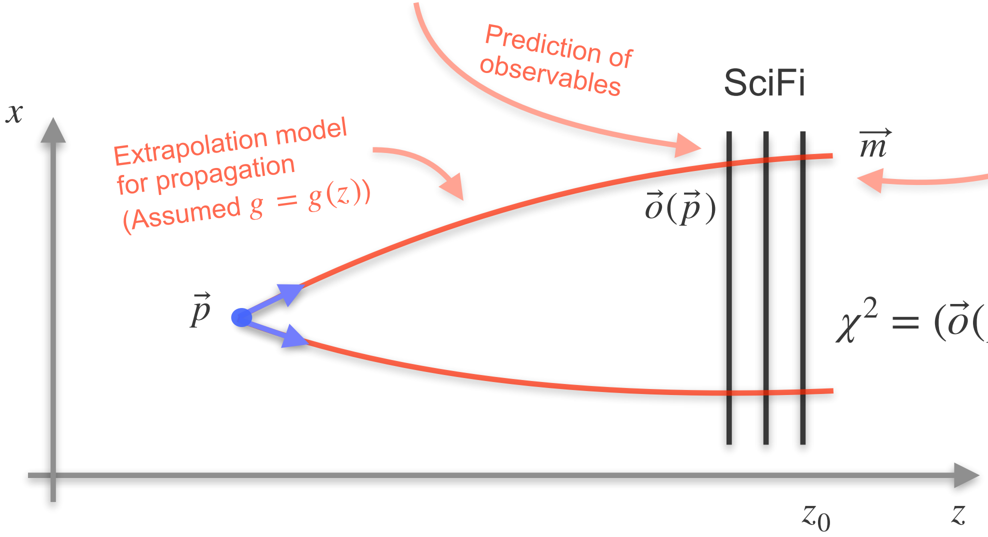


Mathematics of fitting - part 2

Using the parameter vector $\vec{p}$ one may construct the vector of 10 predicted observables:

$$\vec{o}(\vec{p}) = (\underbrace{x^A_{sf}, tx^A_{sf}, y^A_{sf}, ty^A_{sf}, qop^A_{sf}}_{\text{SciFi track A}}, \underbrace{x^B_{sf}, tx^B_{sf}, y^B_{sf}, ty^B_{sf}, qop^B_{sf}}_{\text{SciFi track B}})$$

Observables are measured ($\vec{m}$) with SciFi



The χ^2 may be constructed as:

$$\chi^2 = (\vec{o}(\vec{p}) - \vec{m}) \cdot cov^{-1} \cdot (\vec{o}(\vec{p}) - \vec{m})^T$$

Covariance matrix construction

Elements of the covariance matrix were estimated by definition:

$$cov_{ij} = (x_i - x_i^{true})(x_j - x_j^{true})$$

… and mapped versus some convenient reconstructible variable (e.g. $cov_{00} = cov_{00}(x_0)$)

Some elements were zeroed because of extremely small values ($10^{-10} - 10^{-12}$), and others because of independence in x and y axis, and assumption that qop is independent from y

The resulting covariance matrix:

$$covA = \begin{pmatrix} c00 & 0 & \boxed{0} & 0 & c40 \\ 0 & c11 & 0 & c31 & 0 \\ \boxed{0} & 0 & c22 & 0 & \boxed{0} \\ 0 & c31 & 0 & c33 & 0 \\ c40 & 0 & \boxed{0} & 0 & c44 \end{pmatrix} \begin{matrix} \text{X} \\ \text{Y} \\ \text{Tx} \\ \text{Ty} \\ \text{qop} \end{matrix}$$

$$cov = \begin{pmatrix} covA & 0 \\ 0 & covB \end{pmatrix}$$

Mathematics of fitting - part 3

The minimization with Newton–Raphson method:

$$\vec{p}^{\,i+1} = \vec{p}^{\,i} - H^{-1}\nabla\chi^2$$

where:

$$\nabla\chi_i^2 = 2(\nabla_i d_j \cdot cov_{jk}^{-1} \cdot d_k)$$

$$H_{ij} = 2(\nabla_i d_k \cdot cov_{kl}^{-1} \cdot \nabla_j d_l) + 2(\nabla_j \nabla_i d_k \cdot cov_{kl}^{-1} \cdot d_l)$$

$$\nabla_i = \frac{\partial}{\partial p_i}$$

Here the summation over the same indexes is assumed

To simplify equations, g in the extrapolation model is assumed to be $g = g(z_{ov})$

Turns out, that fit is numerically unstable and slow…

04

Simpler fit

Will it be accurate?

How to make fit faster & simpler

The simplest way to improve stability is to reduce the number of parameters:

Do we need y-z fit?

$$\vec{p} = (\underbrace{x_{ov}, y_{ov}, z_{ov}}_{\text{Vertex}}, \underbrace{tx^A_{ov}, ty^A_{ov}, qop^A}_{\text{Track A}}, \underbrace{tx^B_{ov}, ty^B_{ov}, qop^B}_{\text{Track B}})$$

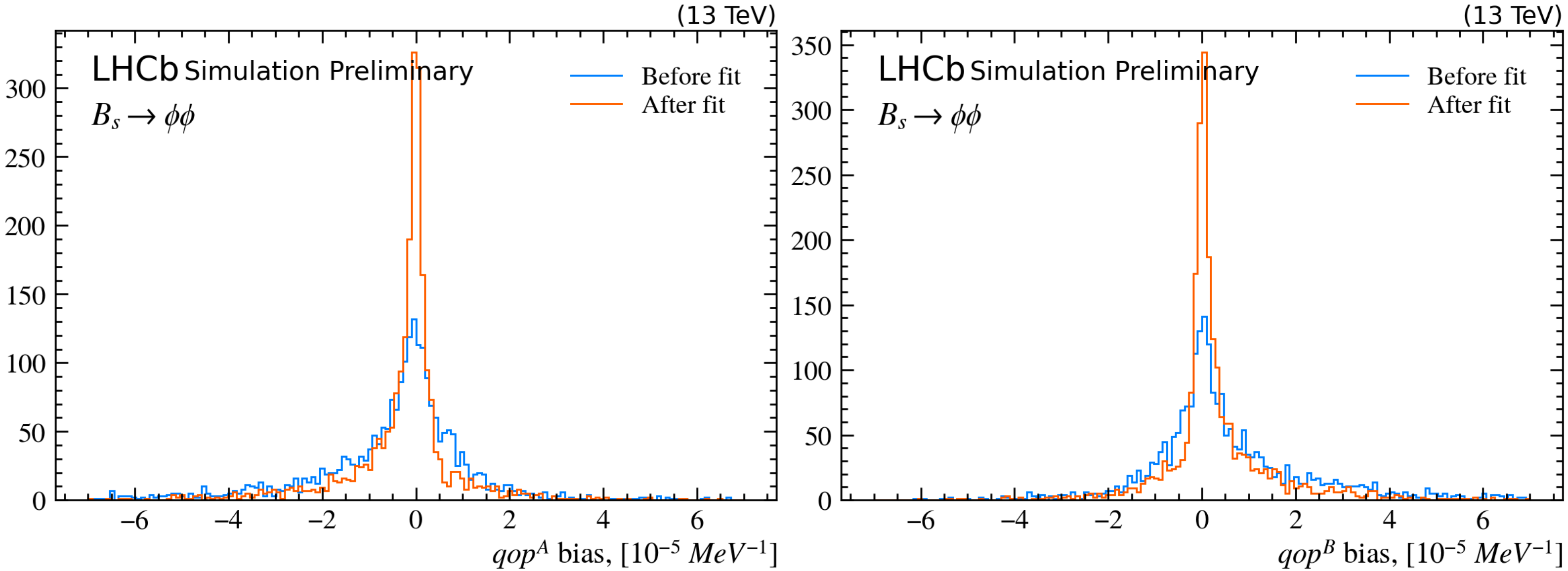
By removing z_{ov} we remove non-linearity from equations

Only 5 iterations enough in Python tests…

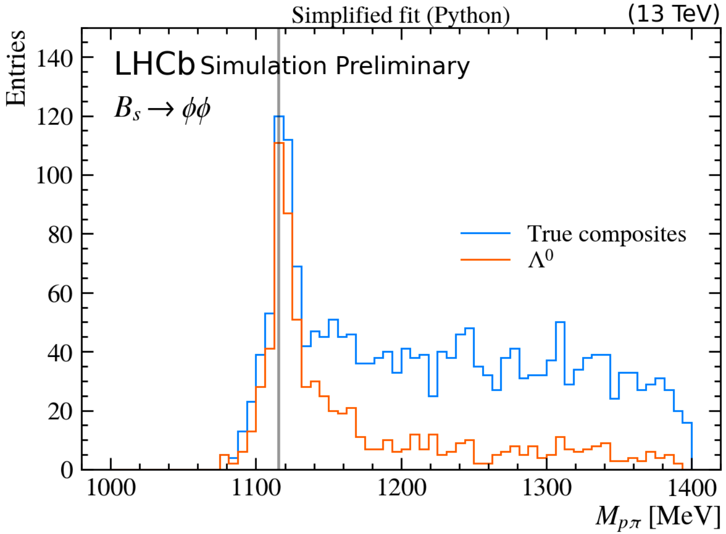
This leads to the following parameter vector:

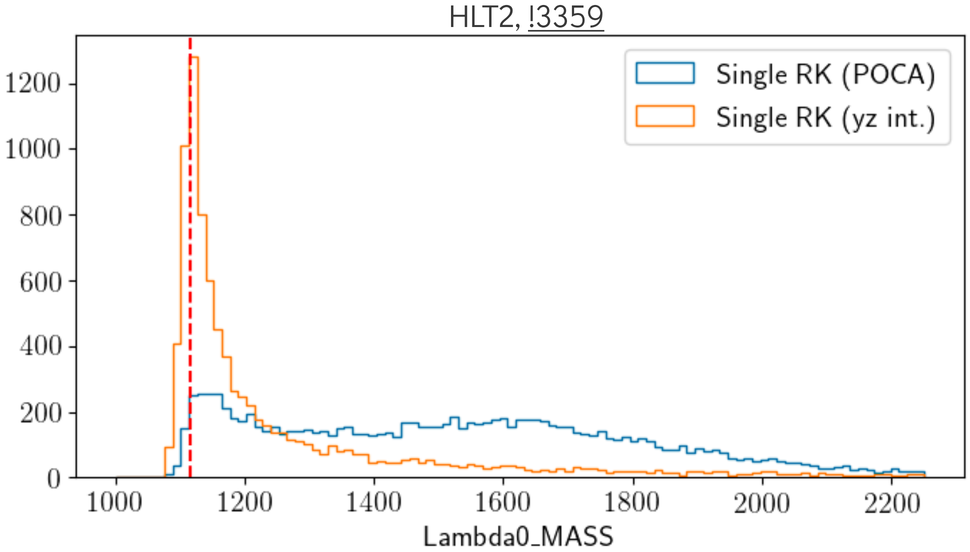
$$\vec{p} = (x_{ov}, \underbrace{tx^A_{ov}, qop^A}_{\text{Track A}}, \underbrace{tx^B_{ov}, qop^B}_{\text{Track B}})$$

Faster Fitting results



Final mass resolution





Thanks for your attention!

What's next?

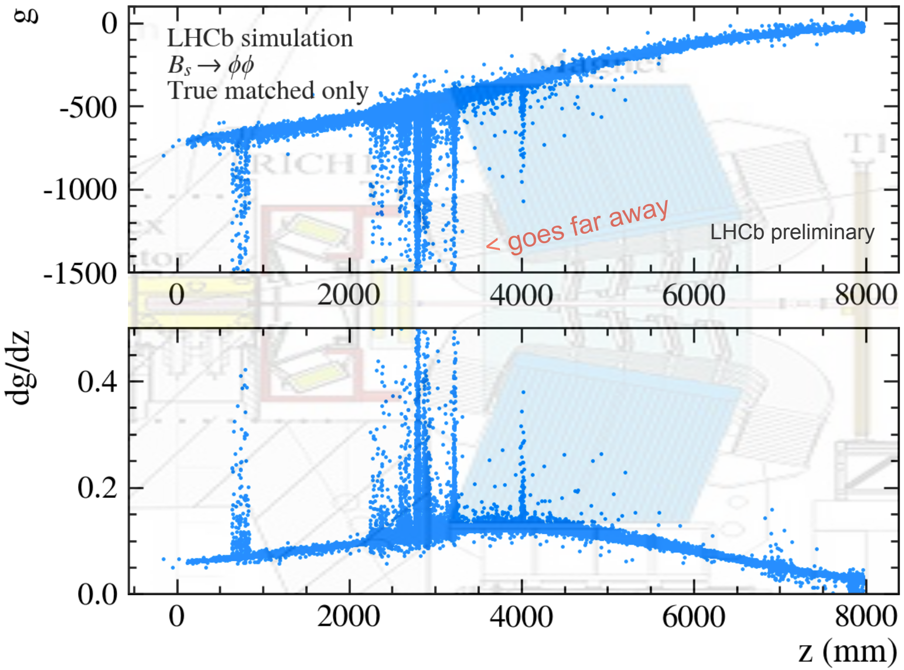
- Implementation & benchmarking in Allen
- Additional studies in order to simplify and/or improve fitting
- Development of selection lines

-01

Backup

Who's there?

T-Track Extrapolation model

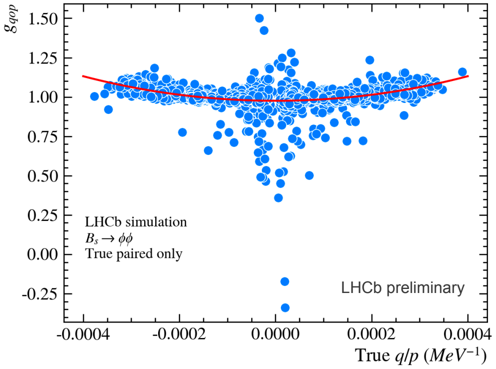


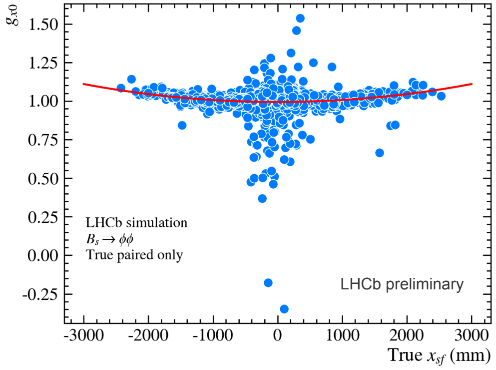
Spikes at specific z, that correspond to detector parts

Material interaction effect?

What if use only true T-Tracks, paired with any other T-Track by common mother id condition?

Corrections for T-Track model





$$g = g_z(z) \cdot g_{qop}(\frac{q}{p}) \cdot g_{x_{sf}}(x_0)$$

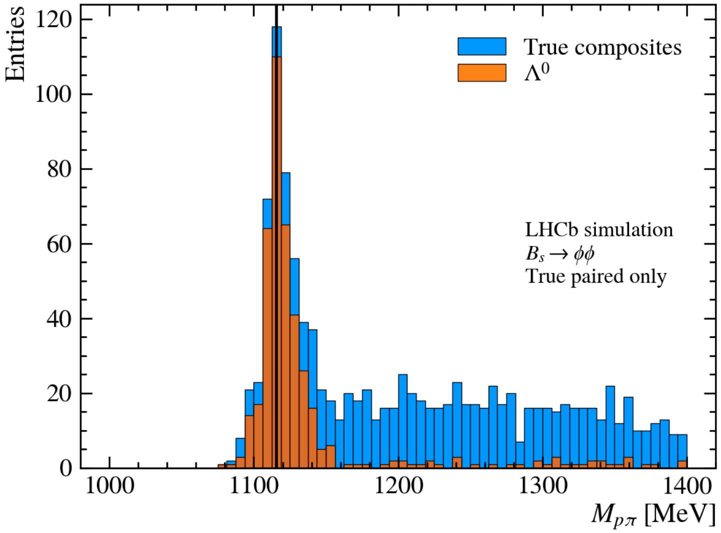
Corrections may be found in a sequential way:

$$g_{qop} = \frac{g}{g_z(z_{ov})} \qquad g_{x_0} = \frac{g}{g_z(z_{ov})g_{qop}(\frac{q}{p})}$$

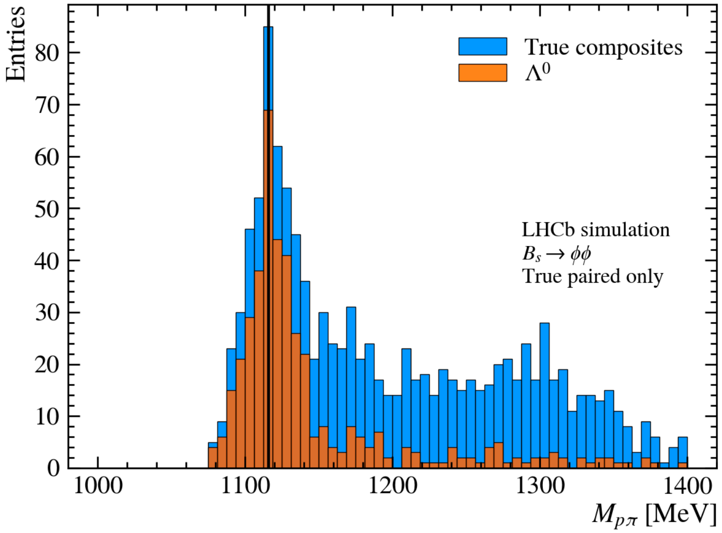
Both functions fitted with:

$$g_{qop/x_0} = a(e^{b(x-c)} + e^{-b(x+c)})$$

Mass resolution with better slope

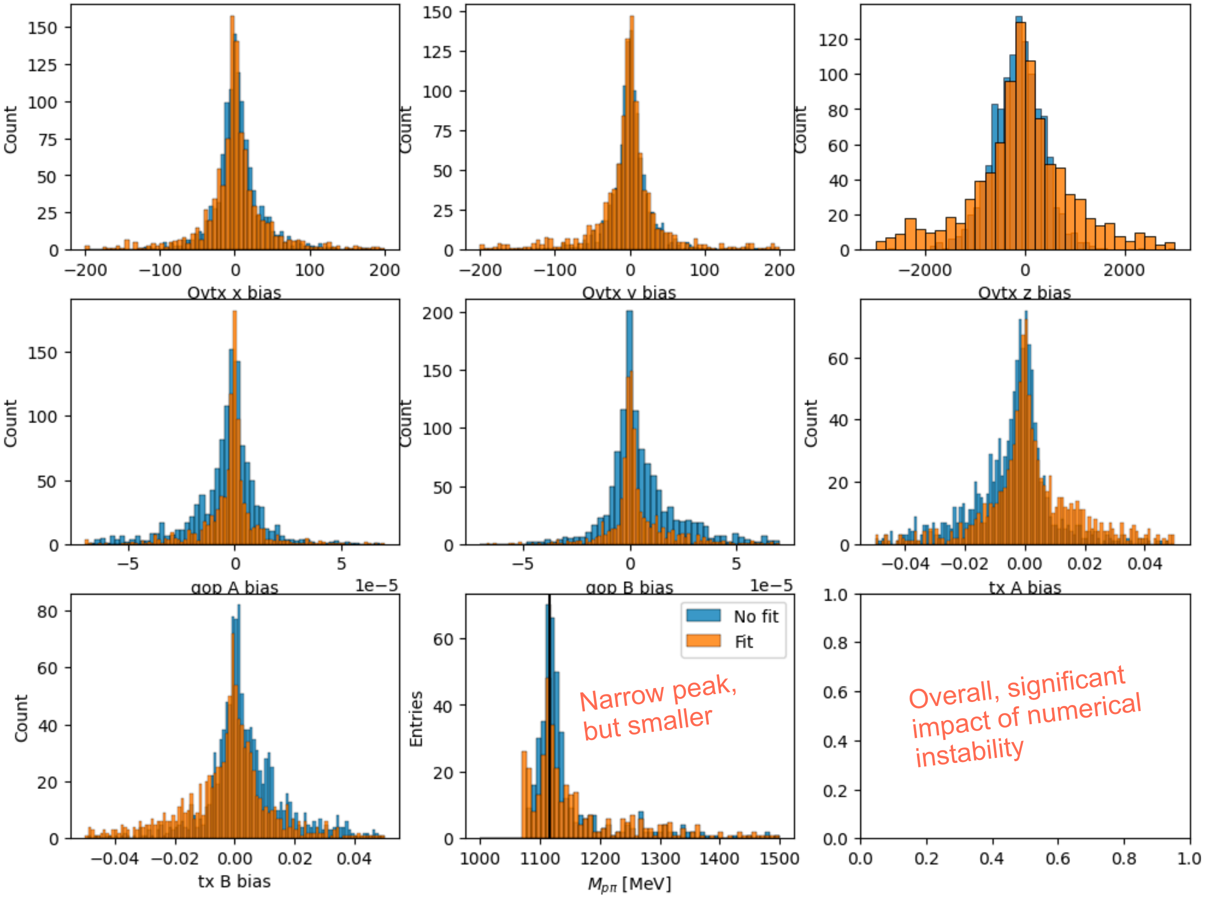


True variables



Reconstructed variables

Fitting results



Little to no improvement in vertex pos.

Significant improvement of qop

Little improvement tx