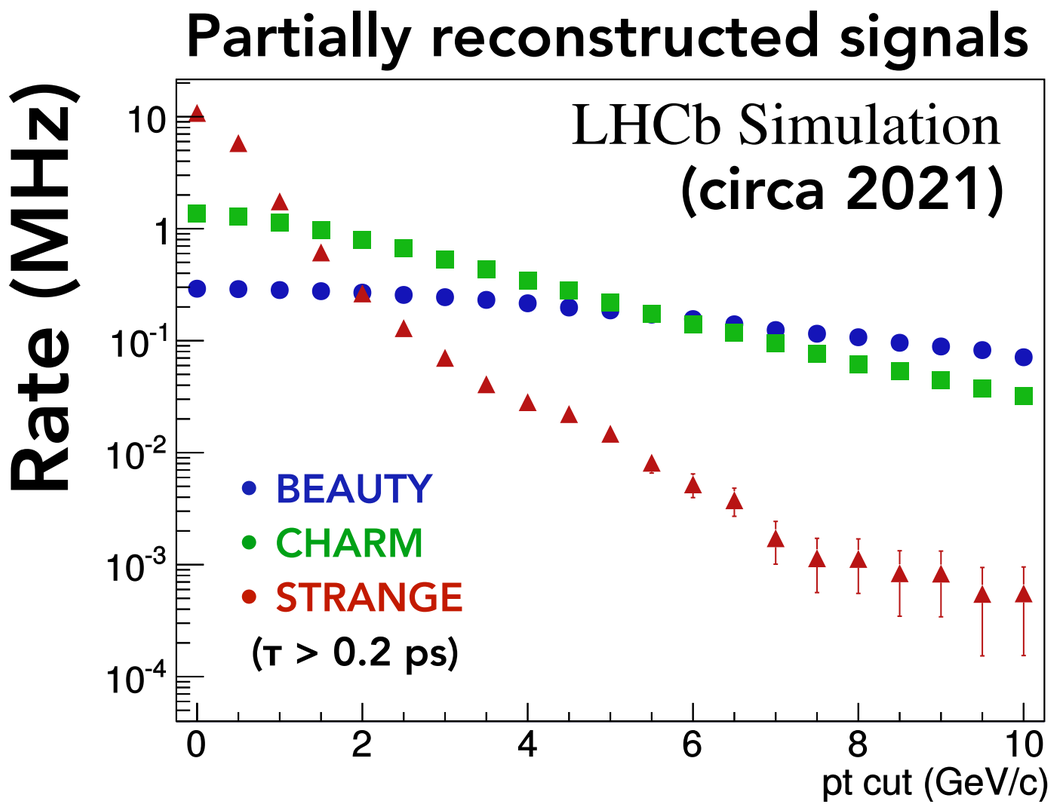
Too much of a good thing
(how to drink New Physics from a 40 Tb/s firehose)

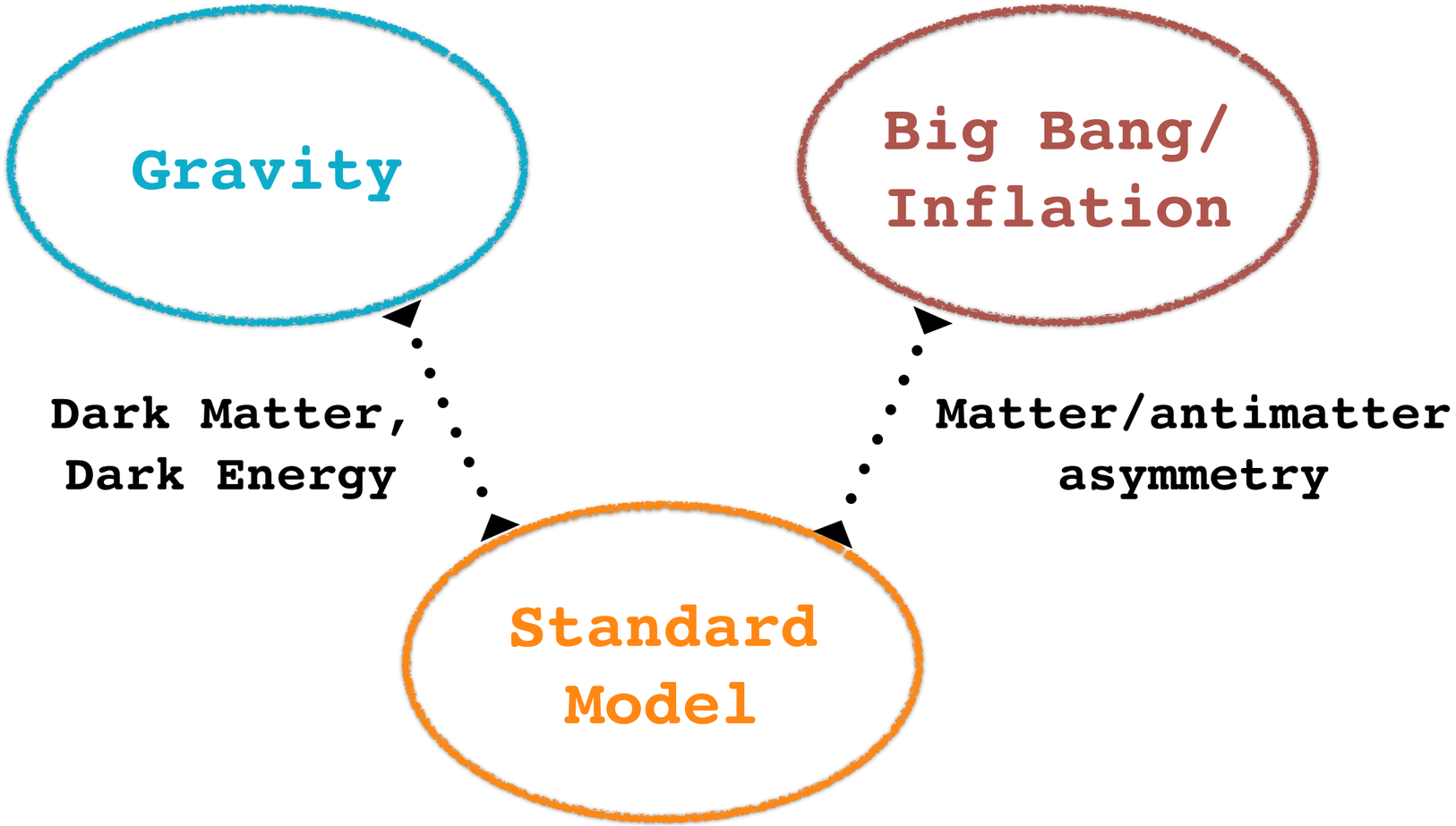


Vladimir V. Gligorov, CNRS/LPNHE

On behalf of the LHCb collaboration

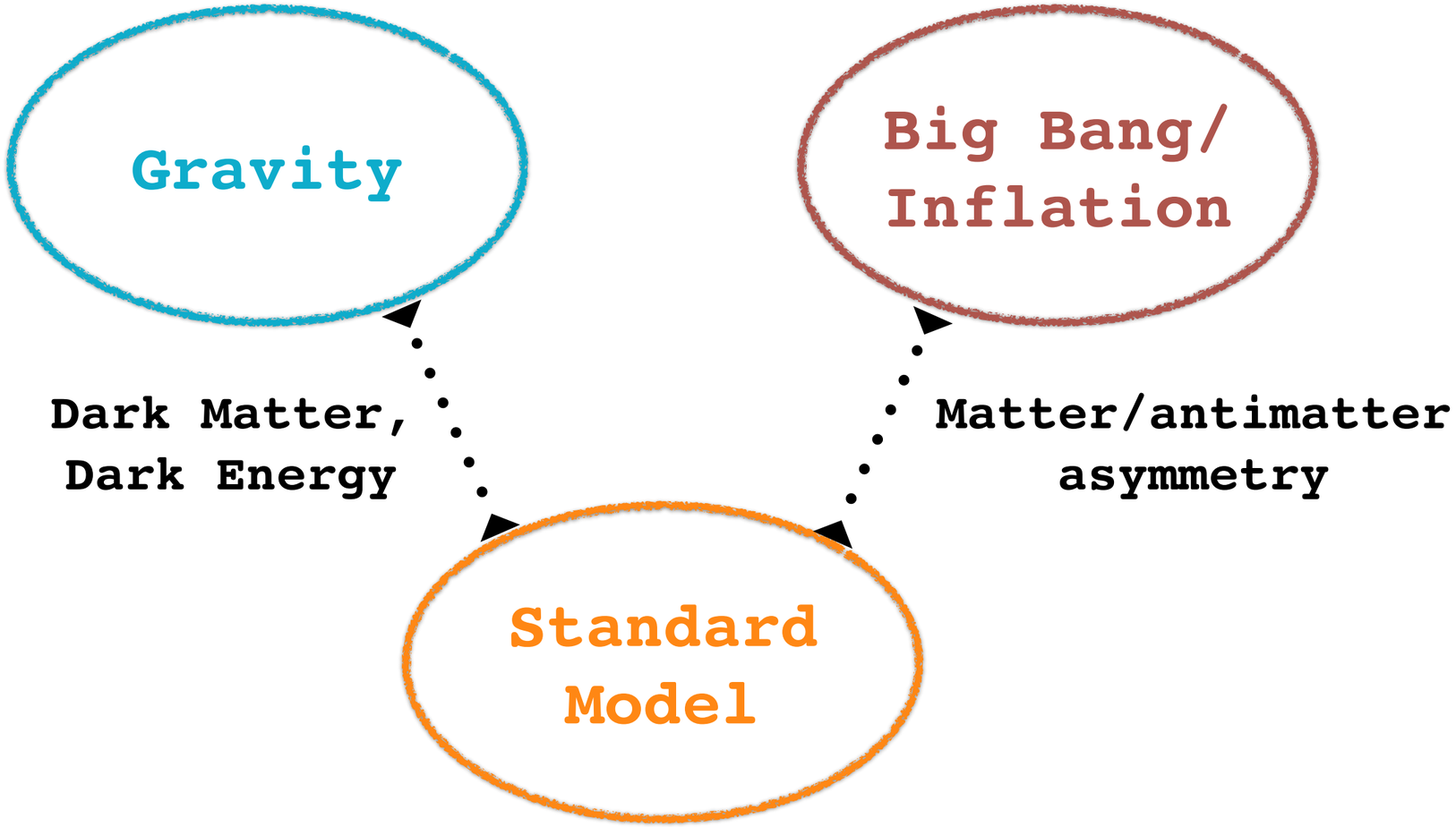
AI @ CERN and SKA, Alan Turing Institute, London 17.09.2018

Why are we here?



Why are we here?

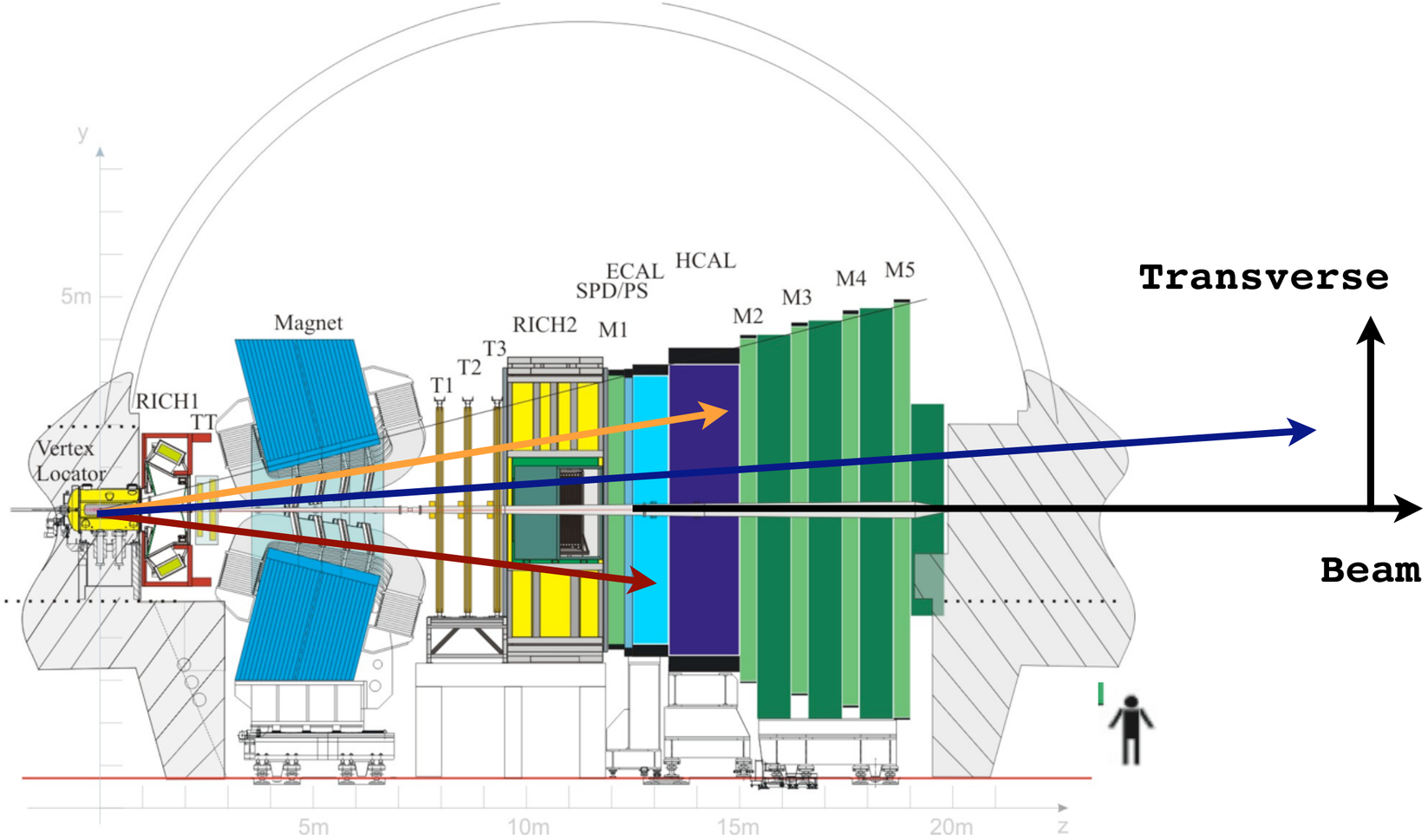
And the really big bad ghoul… nonlocality. But let's not go there.



Our theories of nature are inconsistent with each other => new physics!

Dramatis personae

- ELECTRONS
- PHOTONS
- HADRONS
- MUONS



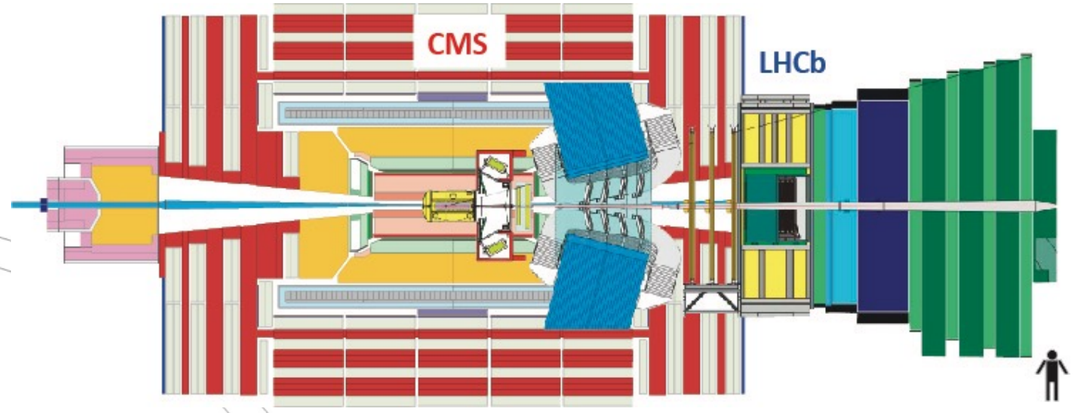
p_T = Transverse momentum

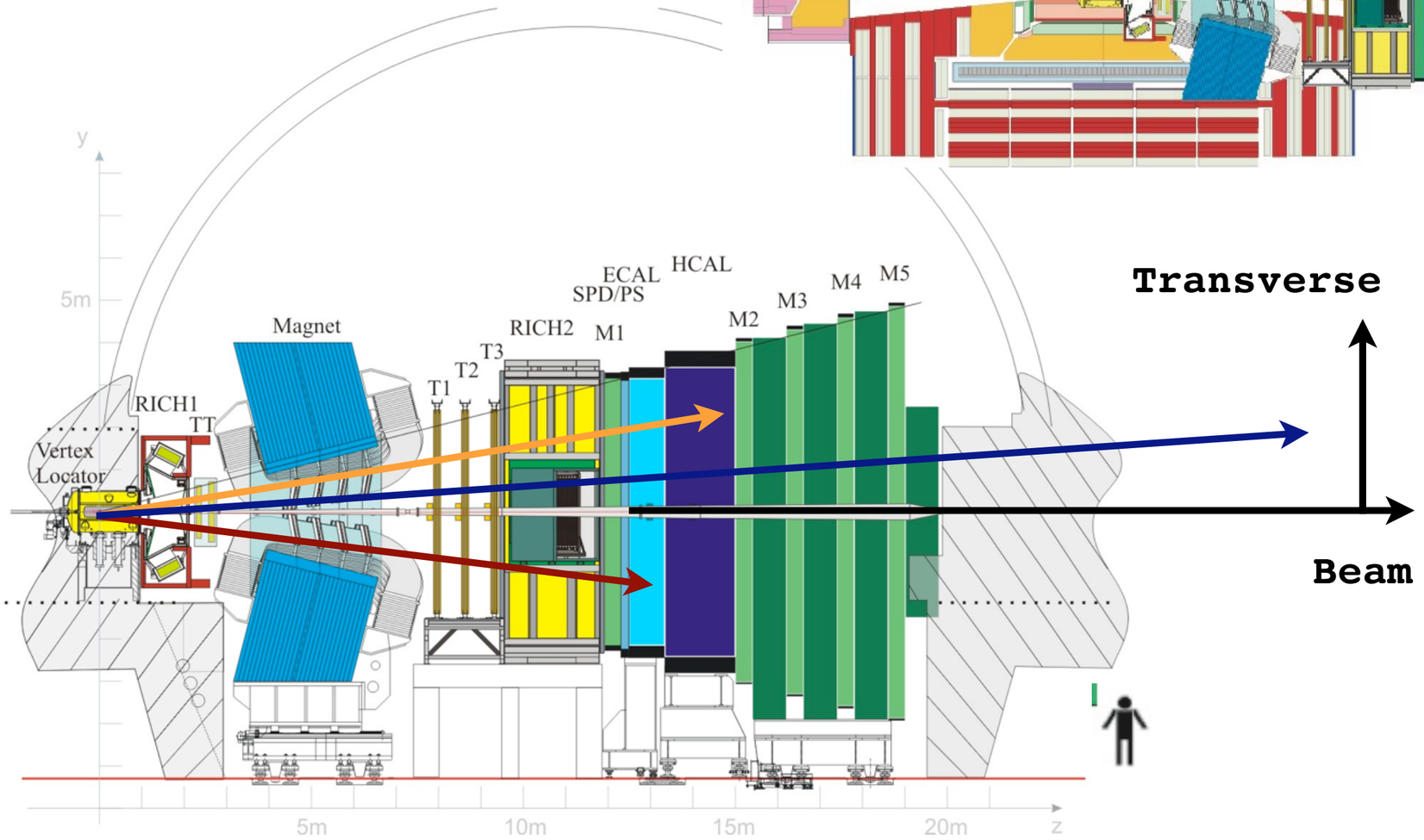
E_T = Transverse energy

LHCb : general purpose detector in the forward direction at the LHC

Dramatis personae

- ELECTRONS
- PHOTONS
- HADRONS
- MUONS





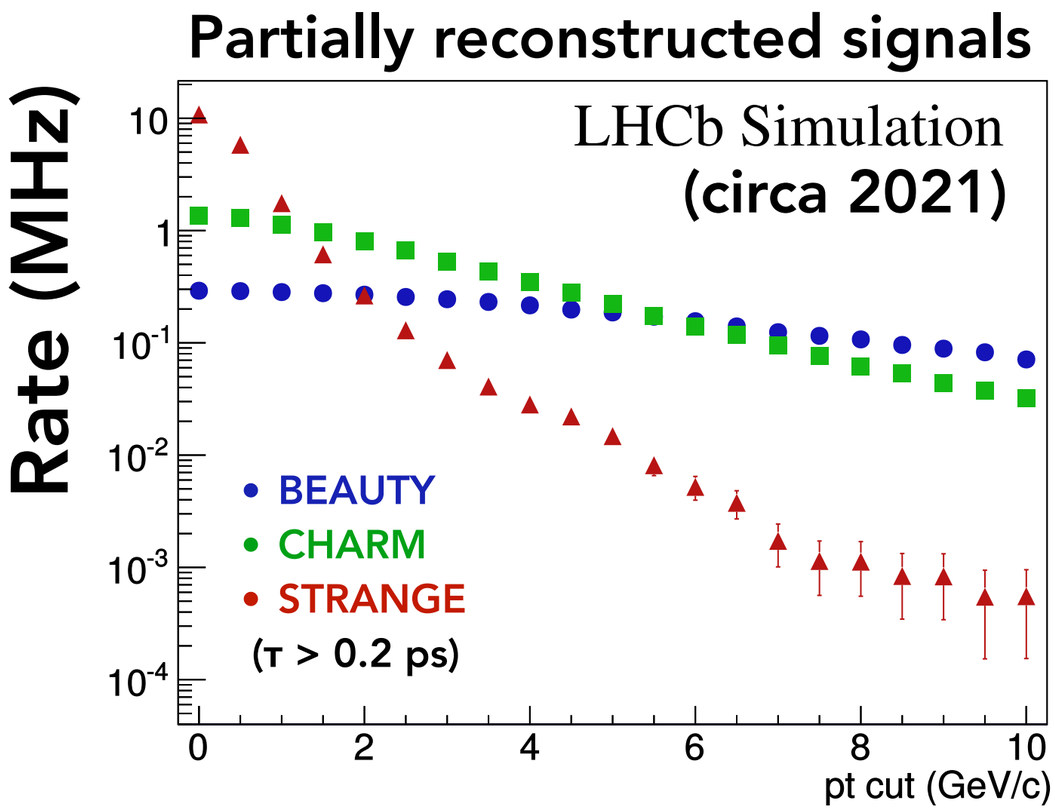
p_T = Transverse momentum

E_T = Transverse energy

LHCb : general purpose detector in the forward direction at the LHC

LHCb path to new physics

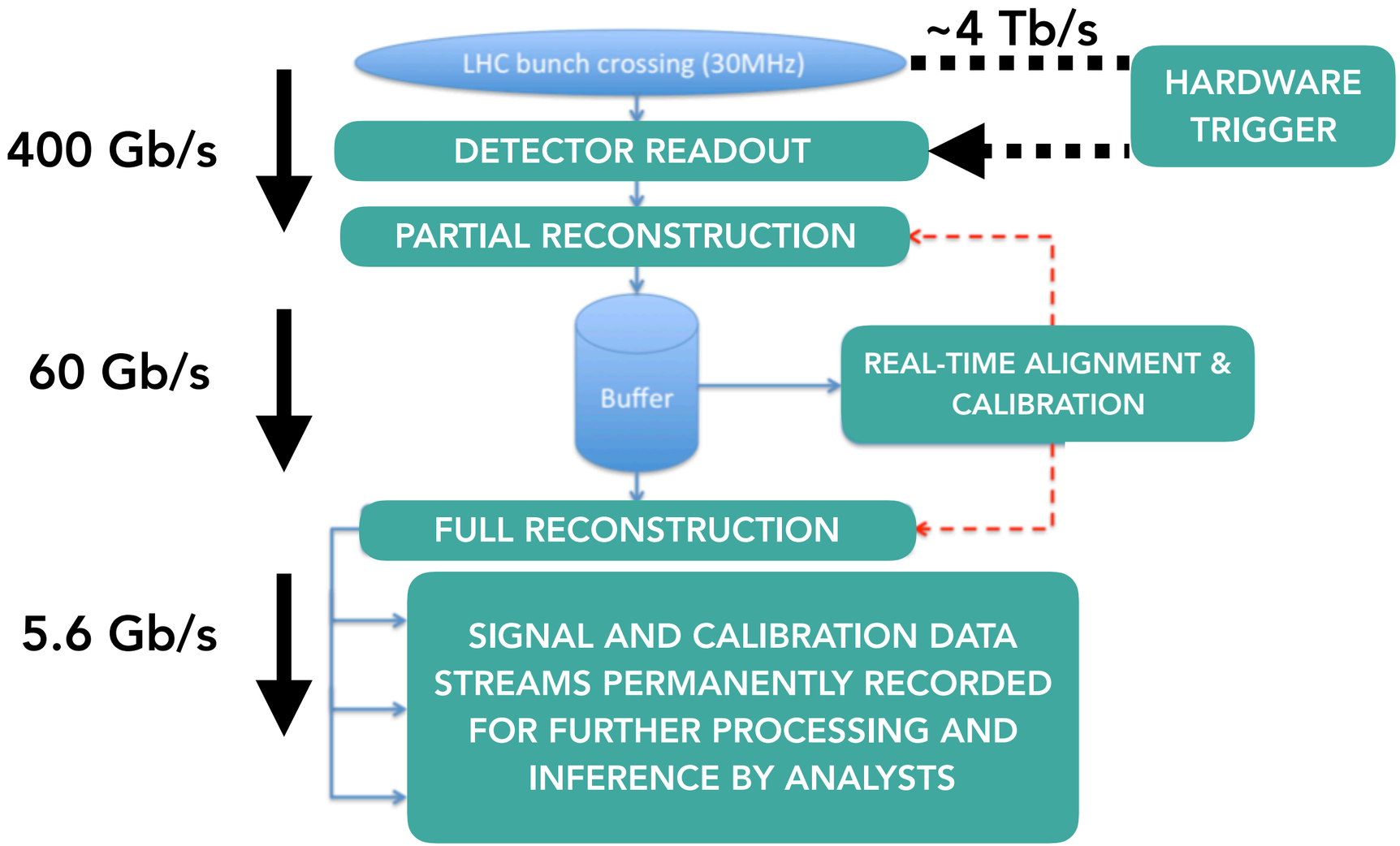
Fitzpatrick&Gligorov Note



Measure properties of known particles as precisely as possible

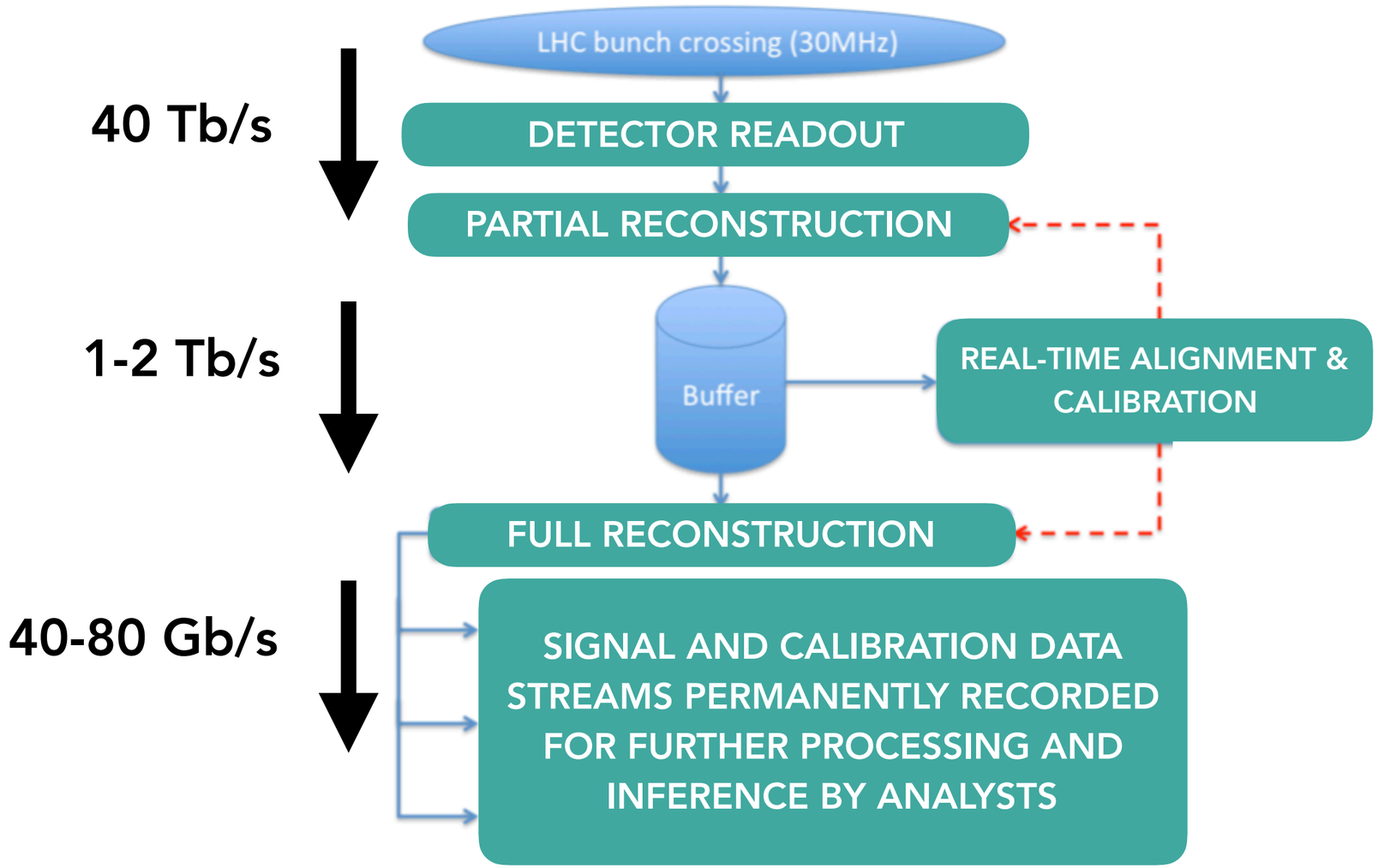
Look for deviations from Standard Model predictions

A lot of signal ⇒ a lot of data to process!



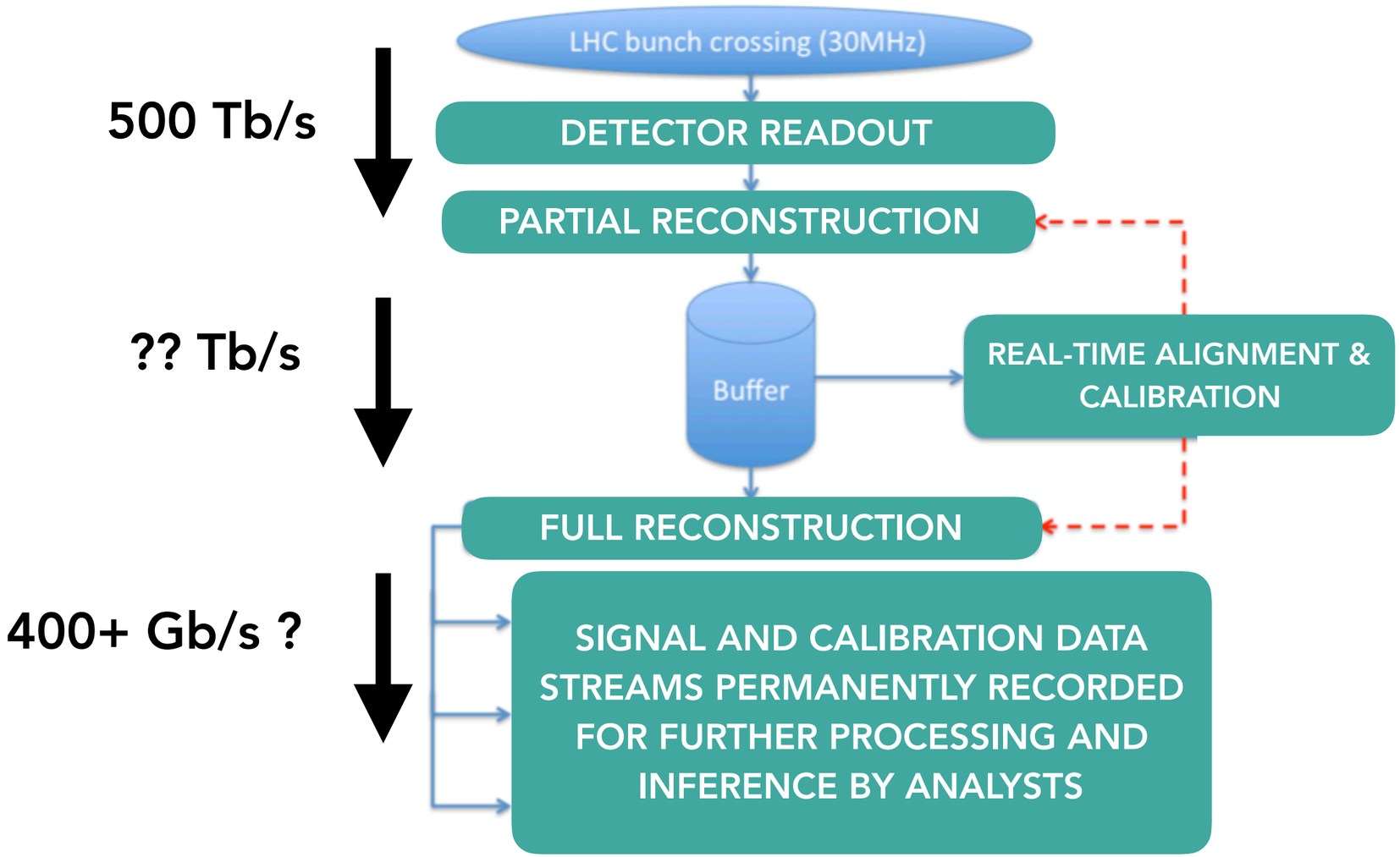
LHCb real-time data processing today

And this data volume will only rise...



LHCb real-time data processing circa 2021

...and rise again



LHCb real-time data processing circa 2032

Data processing is *the* challenge

Where can AI/ML help us make efficient use of our resources?

Where can AI/ML make new research paths possible?

Data processing is *the* challenge

Where can AI/ML help us make efficient use of our resources?

Where can AI/ML make new research paths possible?

I will conflate AI and ML a lot, this is a conscious shortcut but by all means quiz me about it!

Brief history of AI/ML in LHCb

2008 : Wise people run around saying “We have a new detector can I even trust the uncertainties on the particle positions? Please make everything simple!”

2009 : Whispers in the back of the class that actually there are these things called BDTs and actually they are less biasing than “simple” selections for our physics.

2010 : Turn on LHCb, get flooded with data. Deploy BDT in hottest area, reduce data volume by a factor 3 for same physics. It’s going to be OK.

2011—2013 : Rapid and permanent shift towards ML in physics analyses, almost all of which deployed at least a few BDTs or NNs by the end of this period.

2015—today : Gradual expansion of ML methods into pattern recognition. Increasingly energetic R&D into using AI to “learn” to transform the raw detector data directly into physics objects.

First step towards real-time use

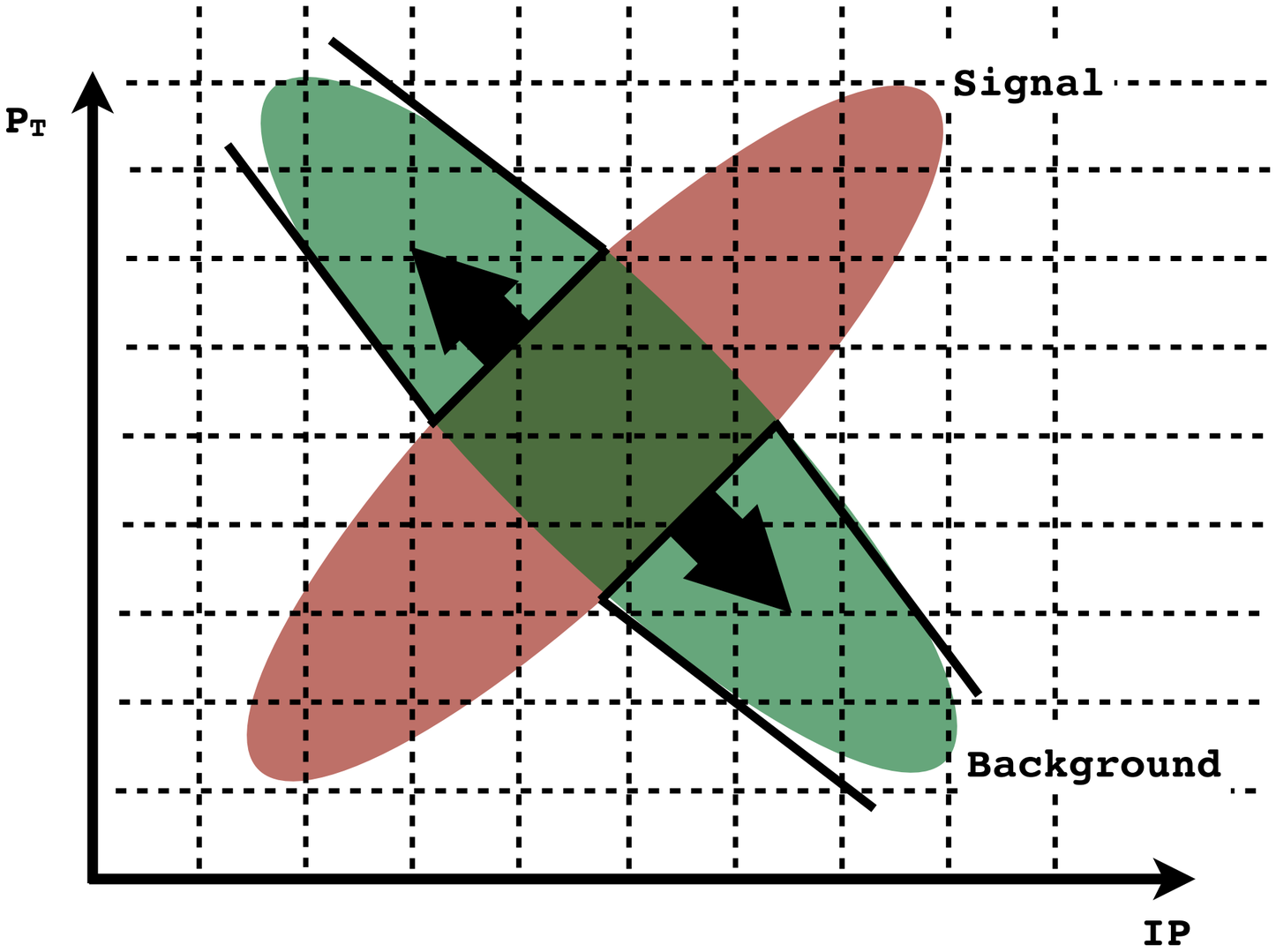
LHCb's performance changes over time

In 2010 we did not have the tools to follow these changes in real-time

How do we make our BDTs insensitive to these changes?

A bonsai boosted decision tree

Consider a two-feature boosted decision tree : this is like a binned selection where the BDT algorithm picks the optimal bin sizes and boundaries



Gligorov&Williams Paper

A bonsai boosted decision tree

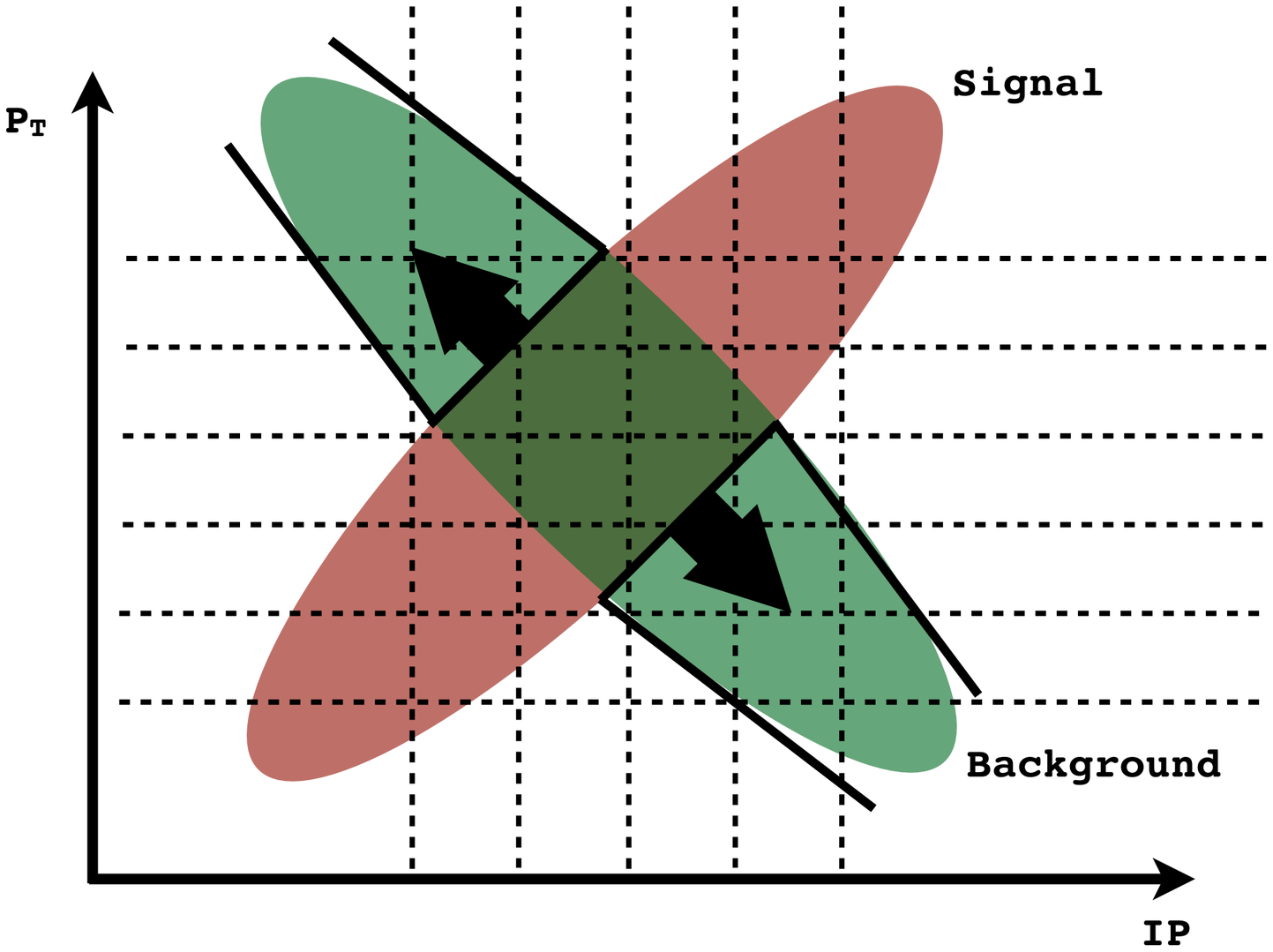
Consider a two-feature boosted decision tree : this is like a binned selection where the BDT algorithm picks the optimal bin sizes and boundaries

Discretize the features yourself!

Pick a bin size based on the detector resolution for that feature.

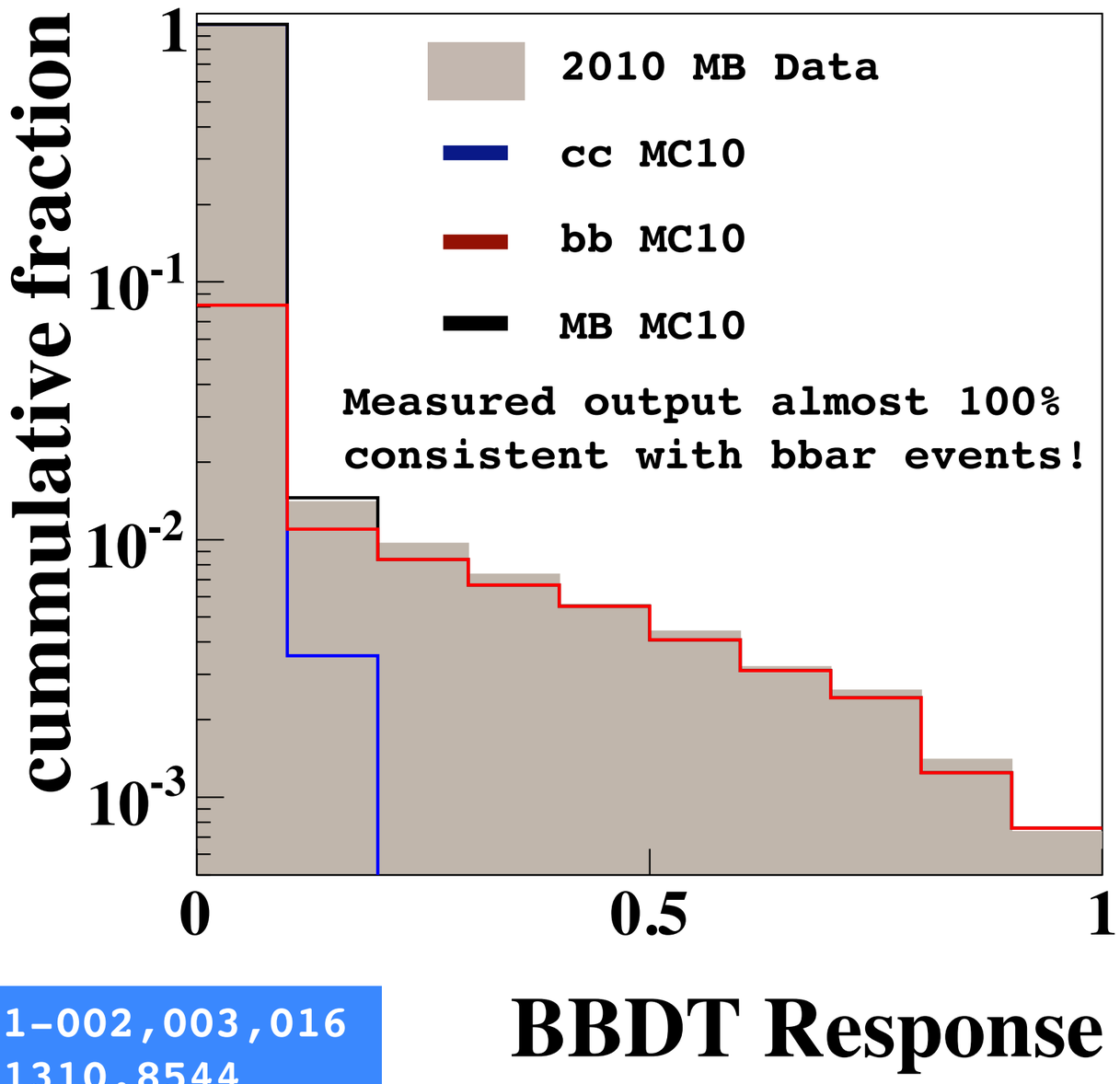
BDT becomes insensitive to bin migration due to changing detector performance.

Transforms BDT into a 1D lookup table making it essentially infinitely fast.



Gligorov&Williams Paper

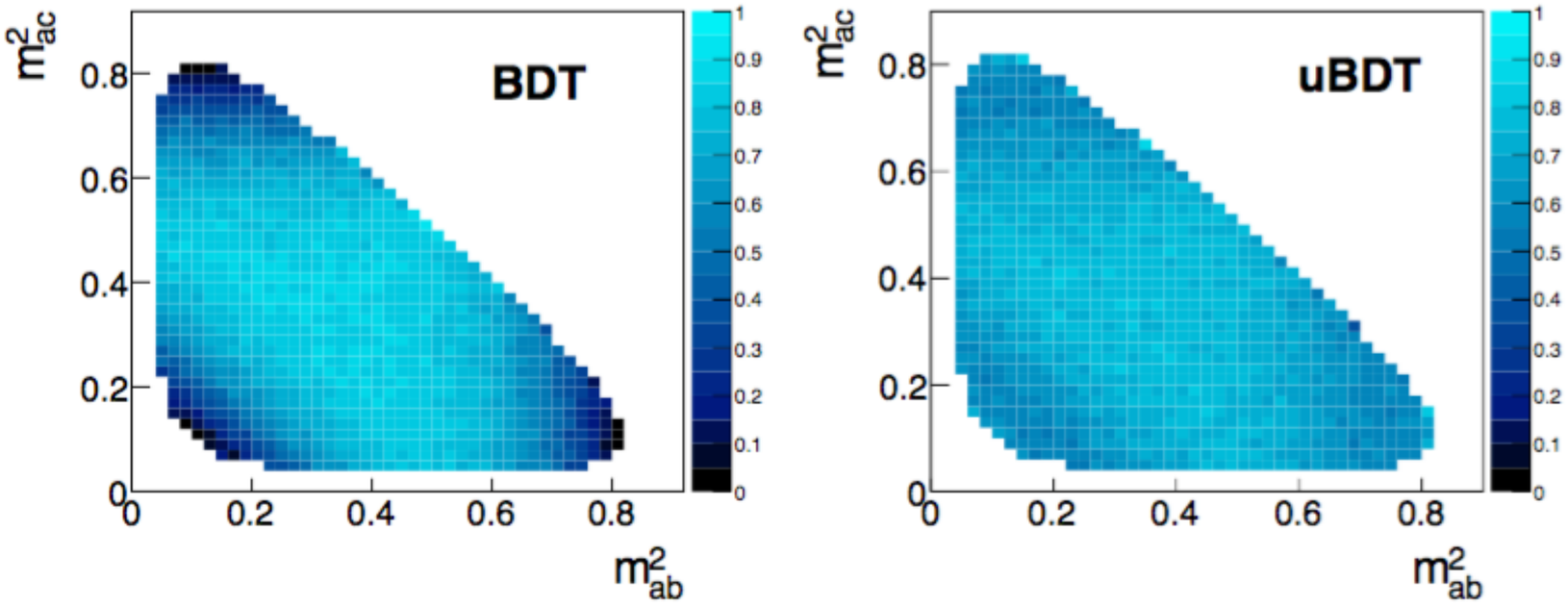
Has our BDT learned something? Yes!



See also LHCb-PUB-2011-002,003,016
http://arxiv.org/abs/1310.8544
http://arxiv.org/abs/1211.3055

Gligorov&Williams Paper

Making ML easier to calibrate



We have to know our classifier efficiencies to infer the physics from the data

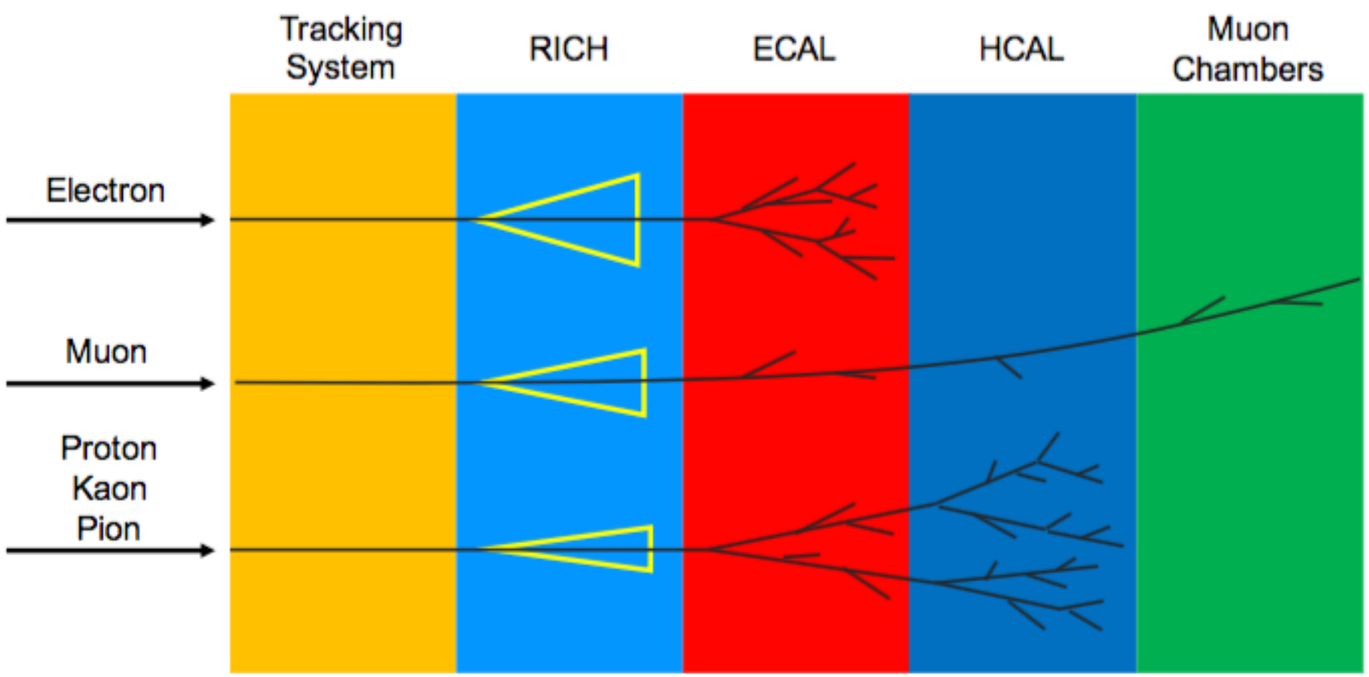
This inference often means measuring the gradients of the data density across certain particularly interesting features, for example particle masses

Easier to do this reliably if the efficiency gradient is as small as possible. LHCb members developed boosting techniques to ensure this, now widely used in our physics programme.

Stevens&Williams Paper

Identifying particles

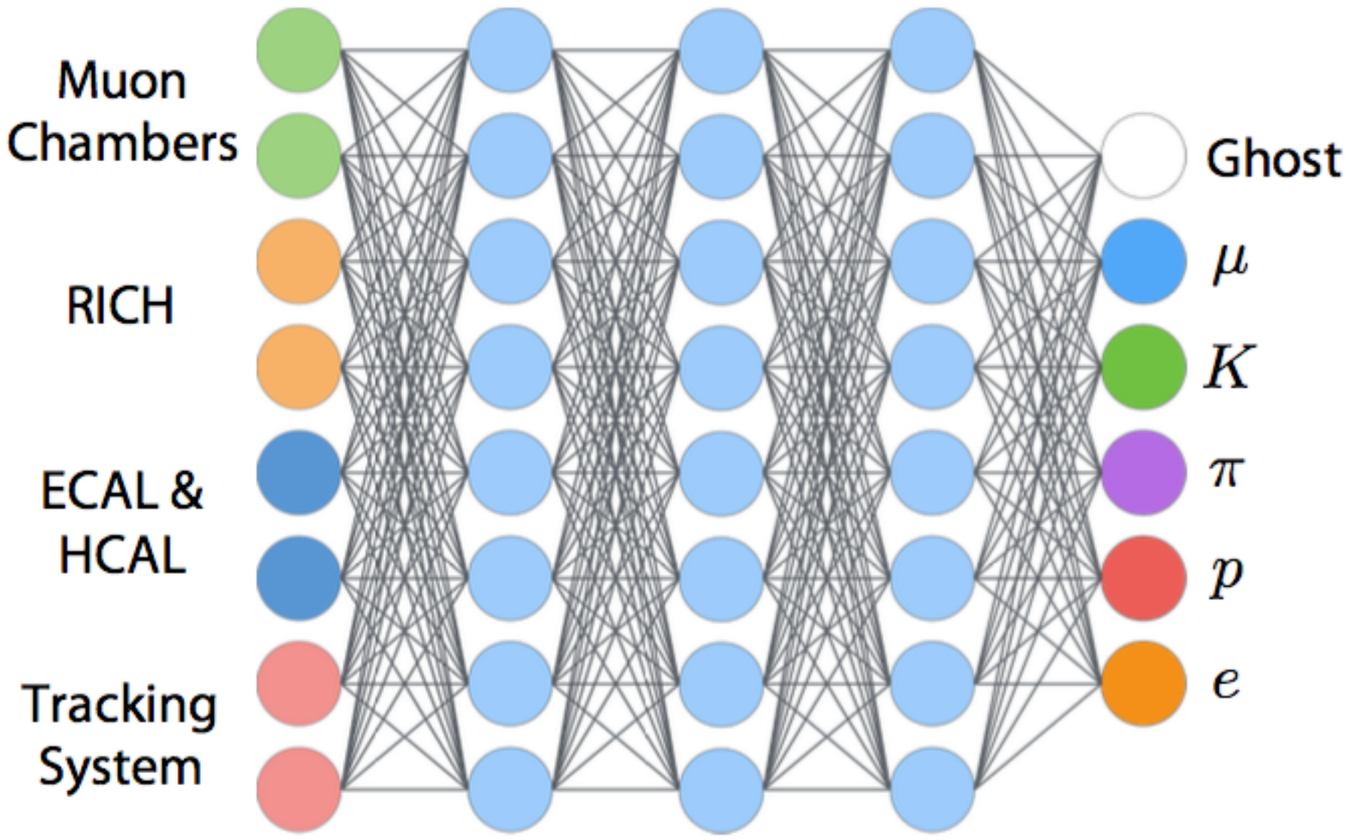
Derkach et al. Poster



LHCb has many detectors which can identify particles. Gain by using all of them!

Classifiers for global PID

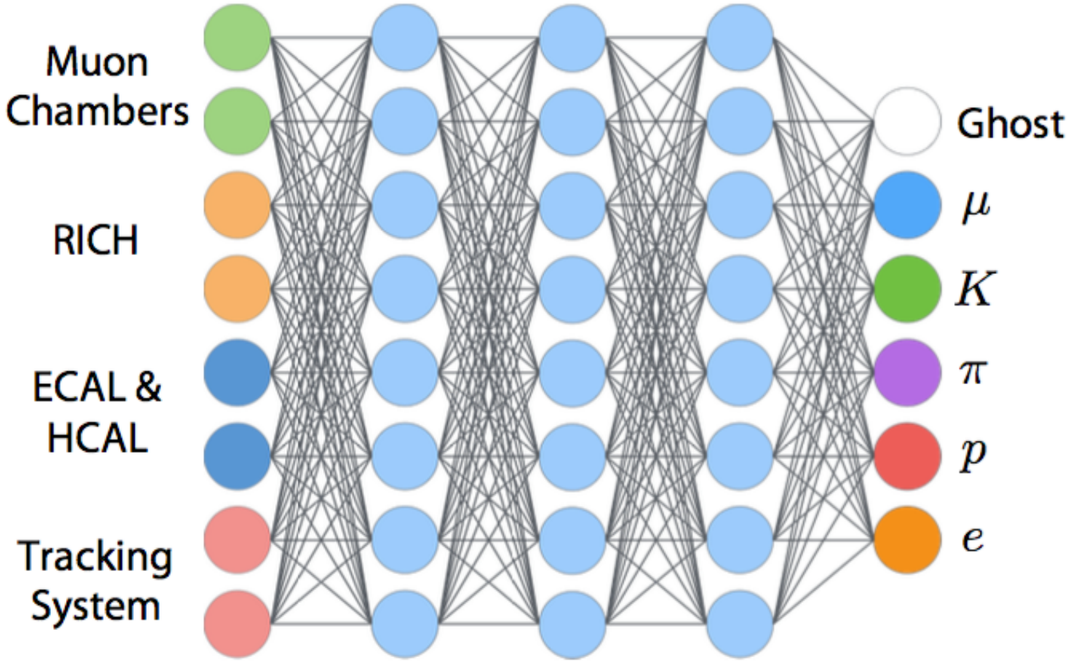
Derkach et al. Poster

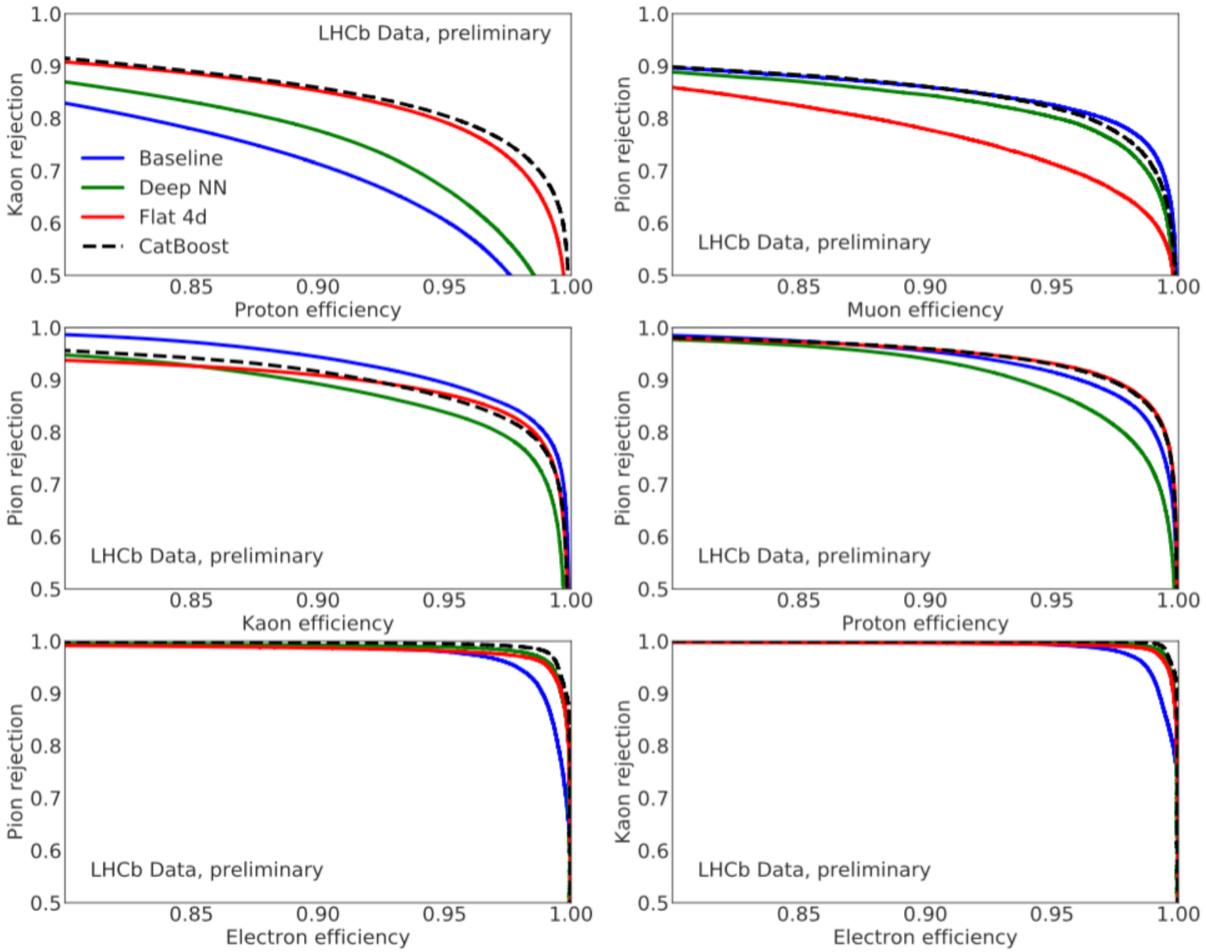


Train many different types of classifiers, BDTs and NNs, deep and shallow…

Classifiers for global PID

Derkach et al. Poster

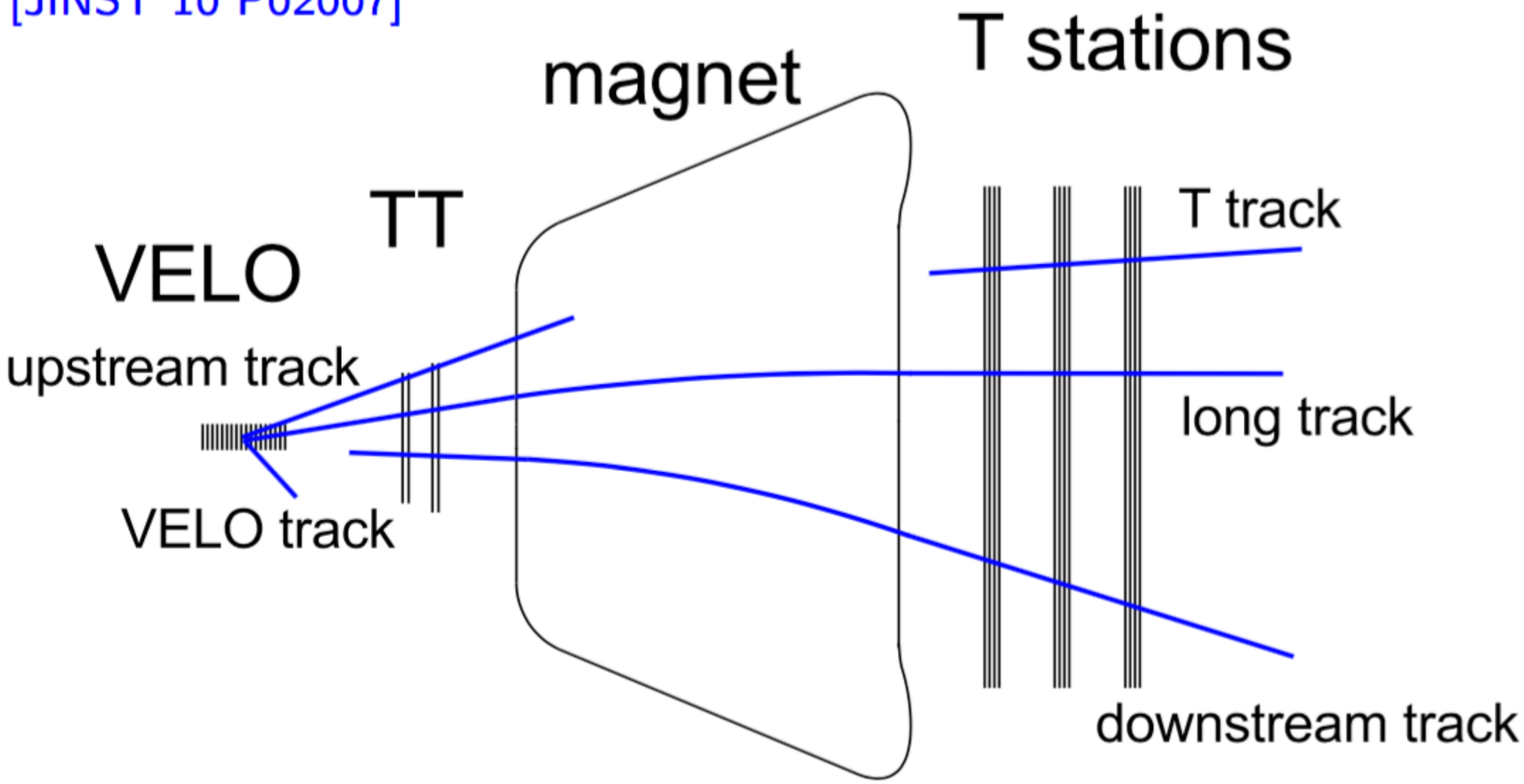




Excellent improvements over baseline in many cases! Final tunings in preparation.

Selecting particle trajectories...

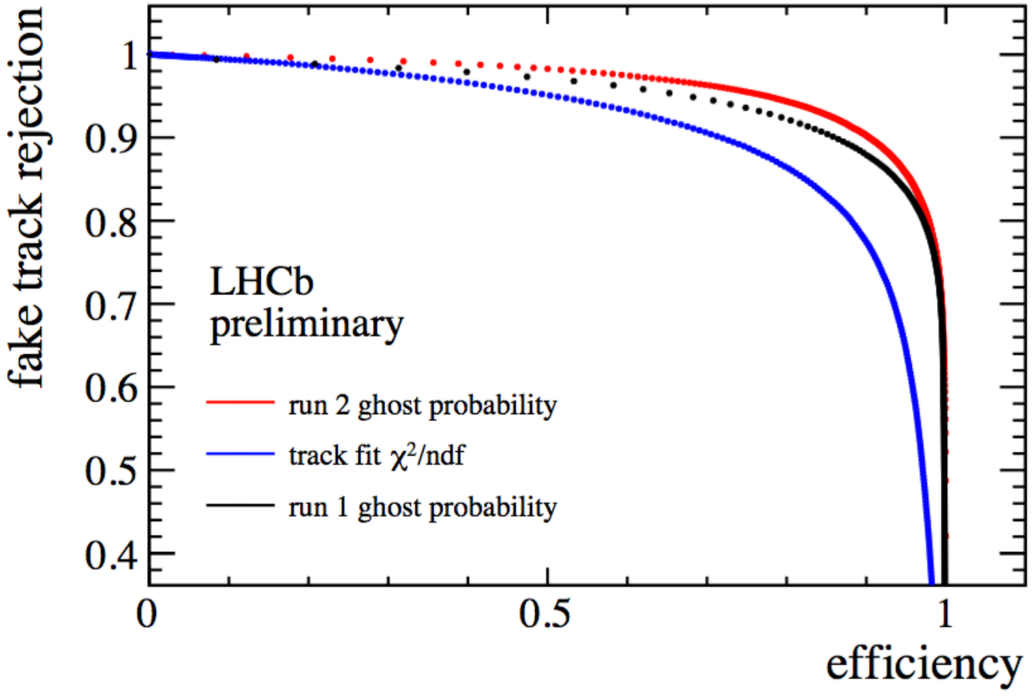
[JINST 10 P02007]

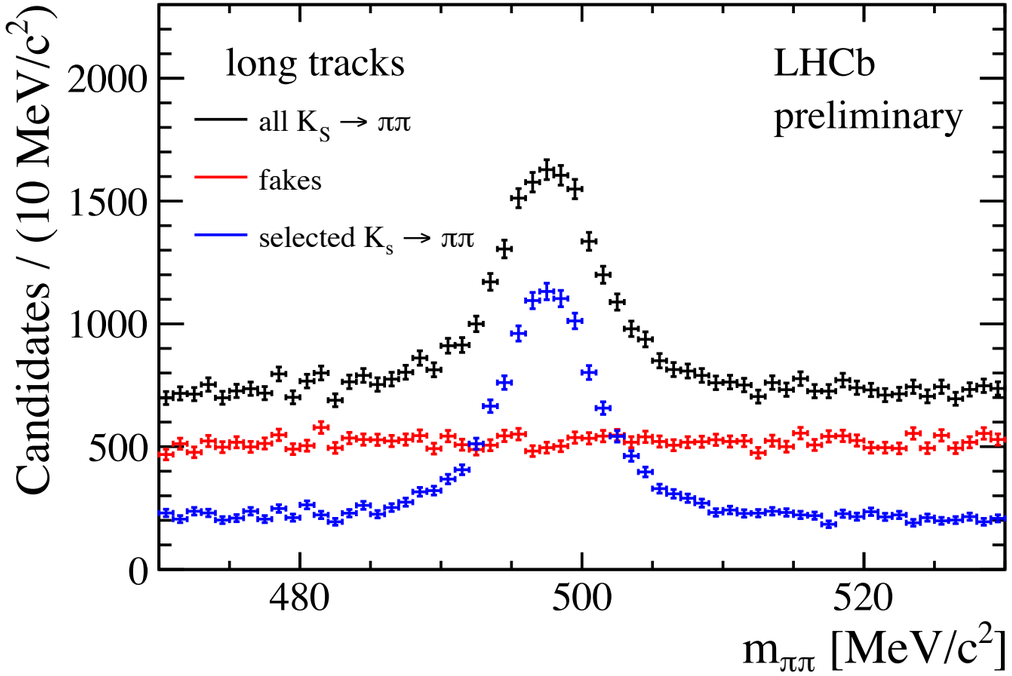


Different particles leave signals in different parts of LHCb. Crucial to reject fake trajectories early! Train NN to put the different detectors in a global selection.

Selecting particle trajectories...

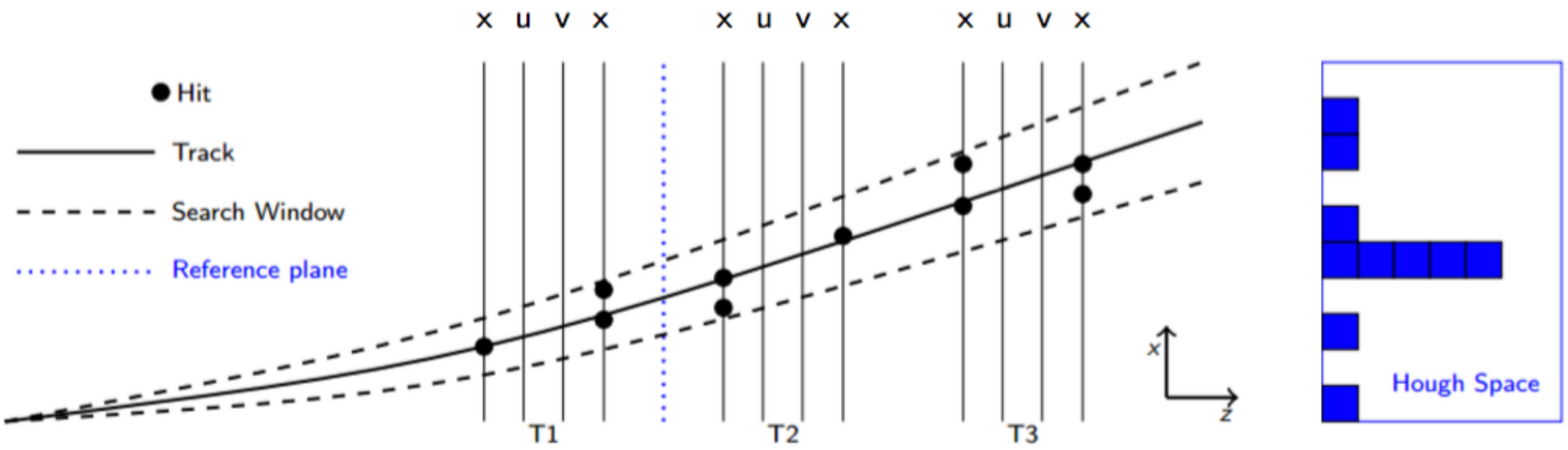
De Cian et al. Note





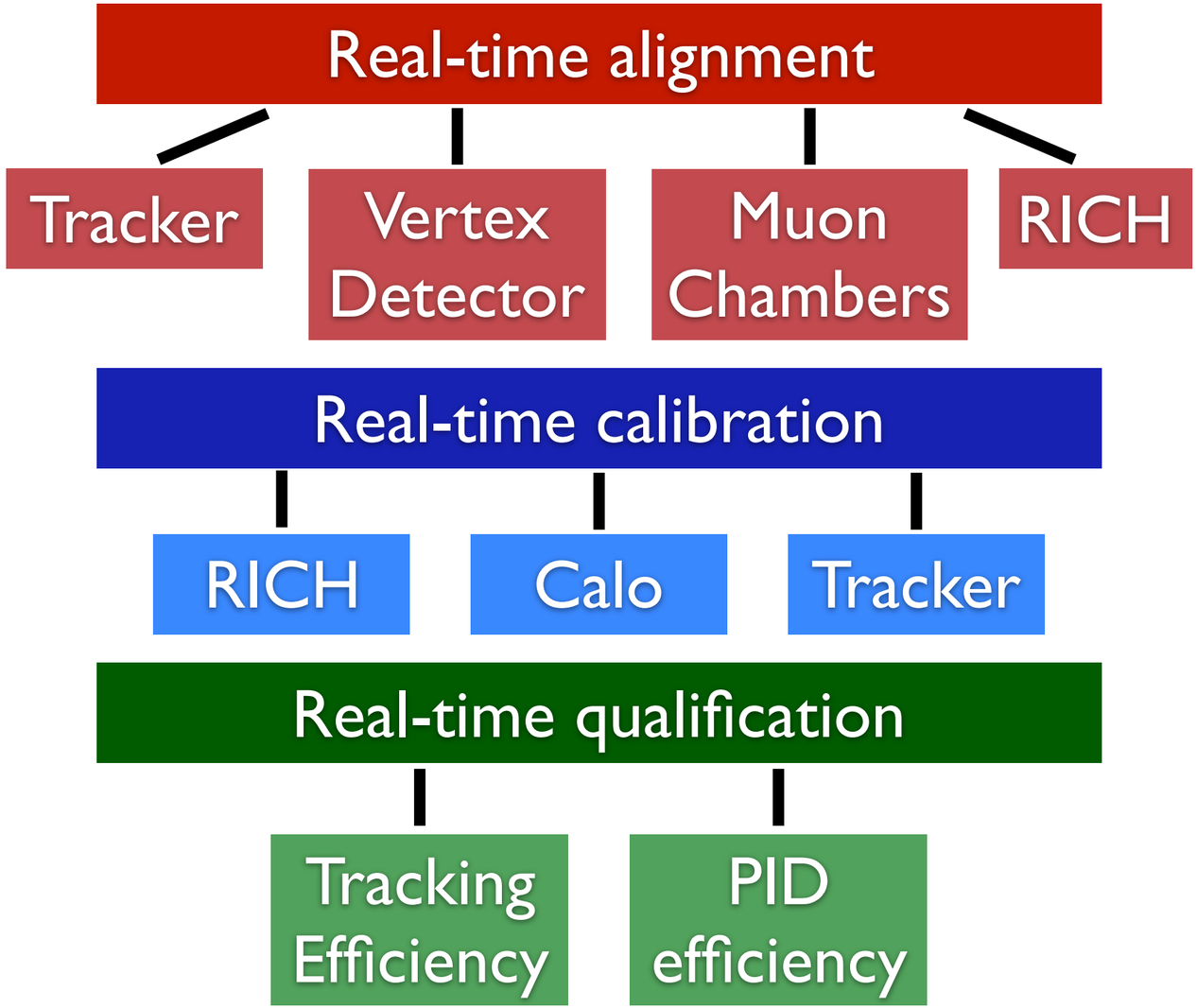
Different particles leave signals in different parts of LHCb. Crucial to reject fake trajectories early! Train NN to put the different detectors in a global selection.

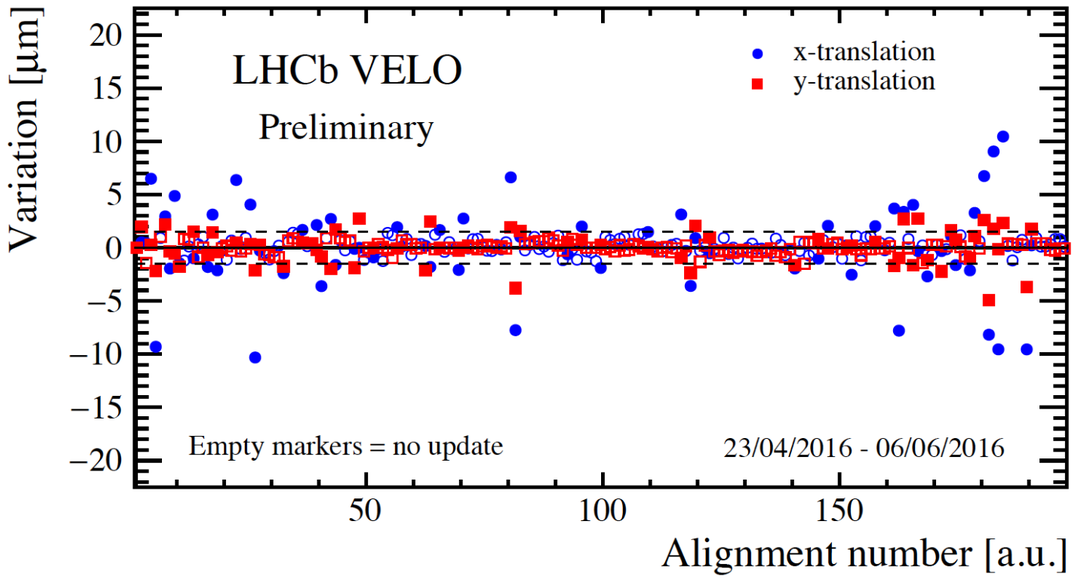
...as a step towards pattern recognition

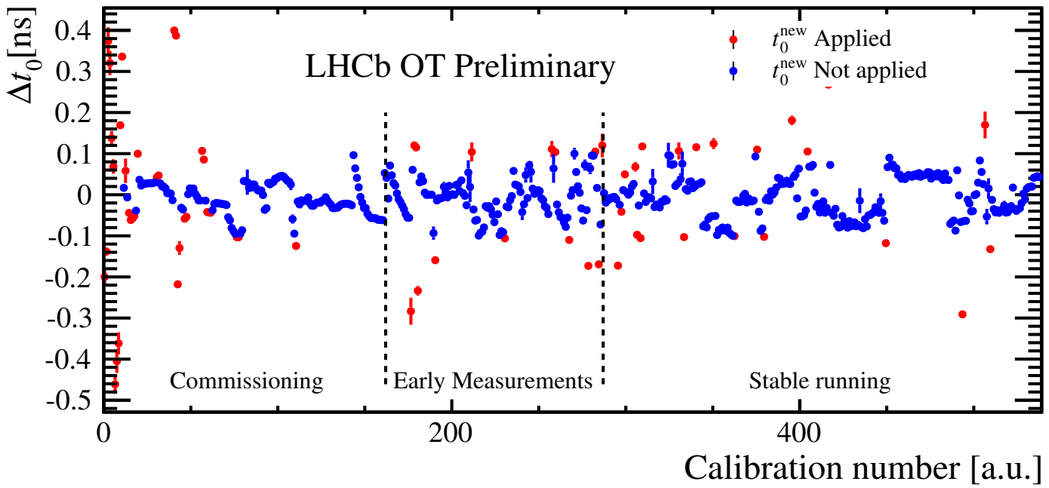


Also add two NN classifiers inside the pattern recognition itself for an early rejection of bad hit combinations => significant speedup of the pattern recognition code. But the NN is still not making trajectories from hits, but rather classifying trajectories.

Real-time alignment and calibration



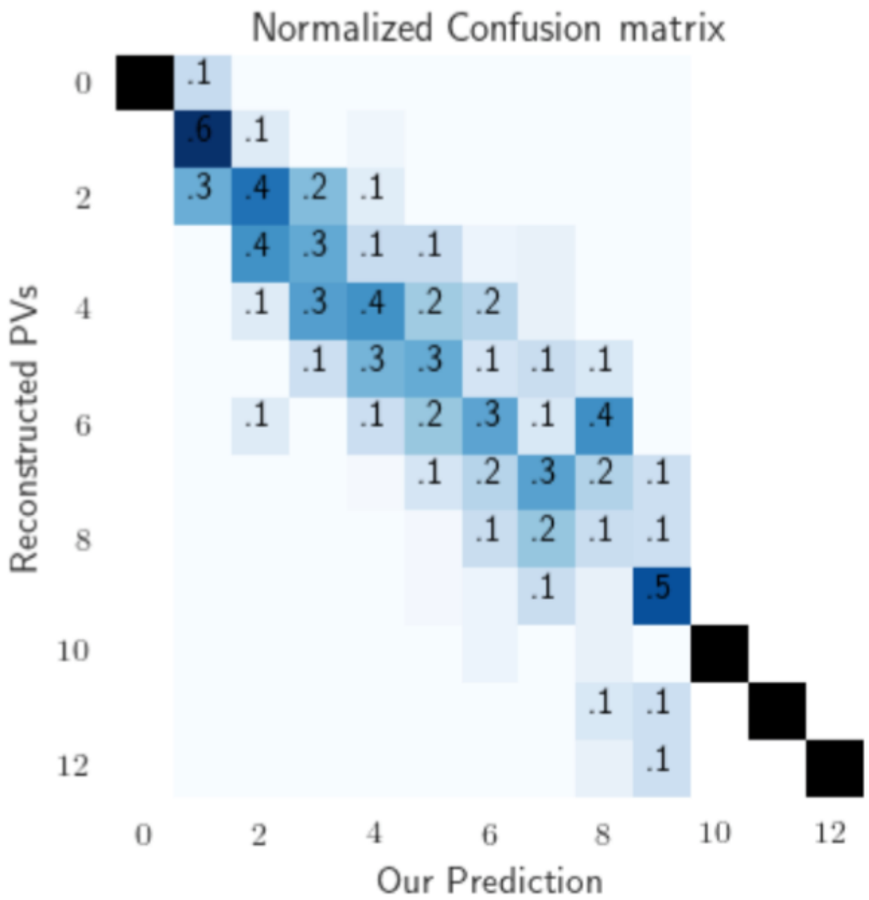




AI/ML can only be as good as the input data!
Big advance from 2015 onwards : automatically align and calibrate the detector in real time. Same performance as traditional "by hand" calibration previously run by experts!

Globally interpreting collisions

Bourgeois et al. Note

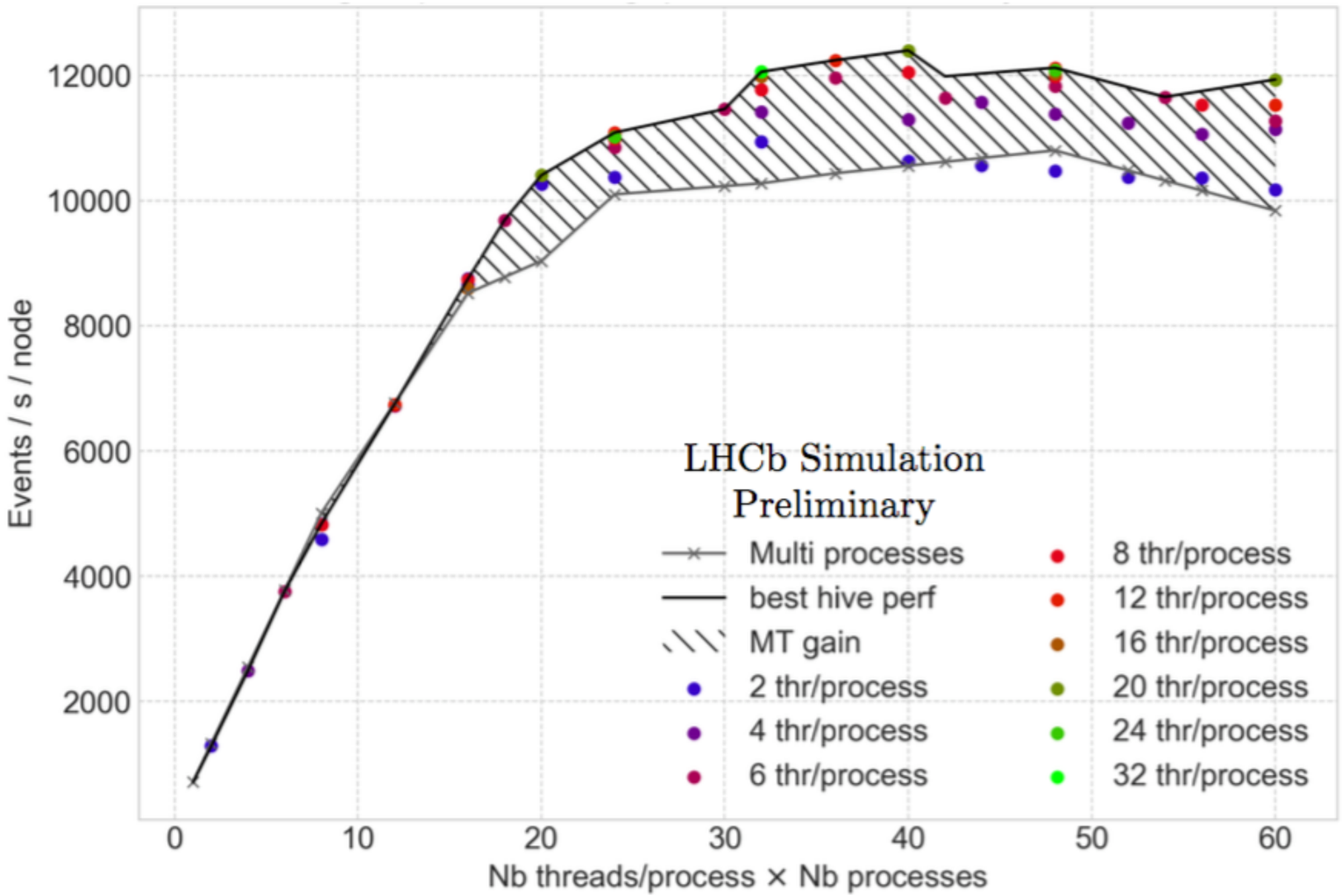


Predict number of proton collisions from the raw hit information in the individual detectors, to guide later reconstruction. Another step on the road from ML to AI…

So LHCb already uses AI/ML throughout its data processing and analysis chains, including in real-time applications

What avenues are we exploring for the future?

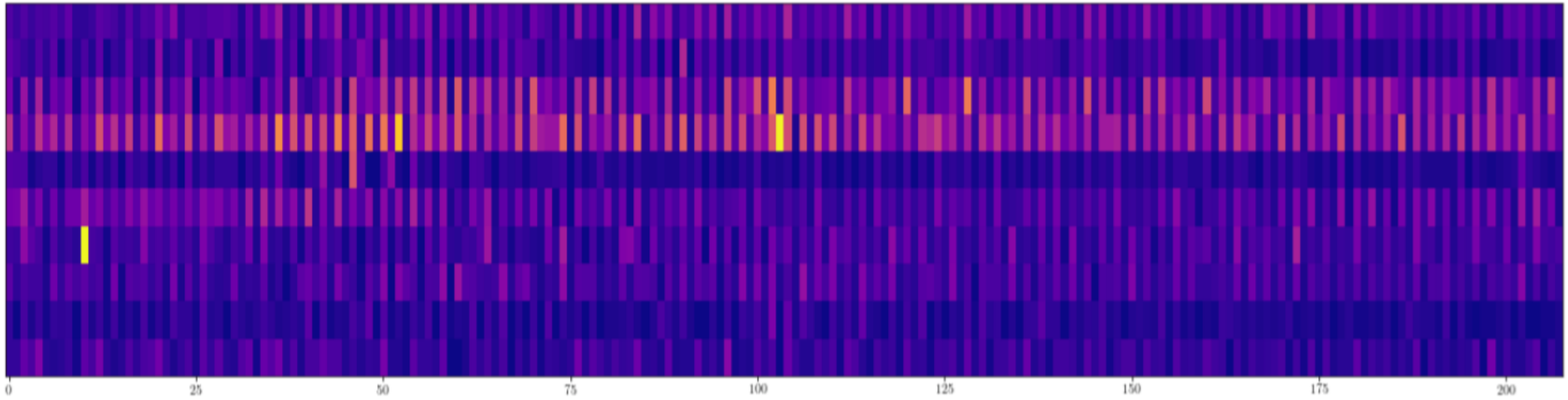
Efficiently using parallel architectures...



Huge effort over the last years to improve performance by enabling multithreading, vectorizing algorithms, etc. Significant gains achieved but more needed!

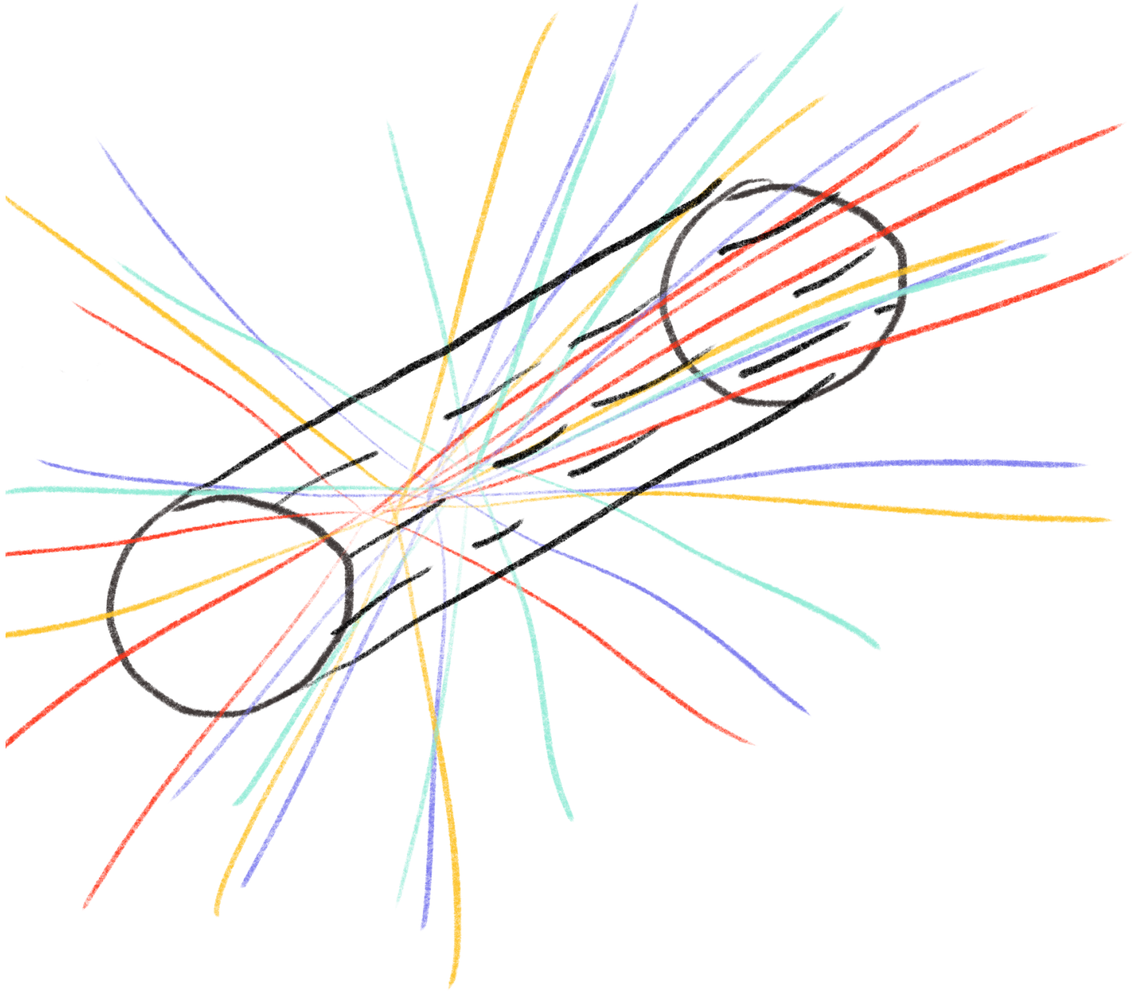
...with super sparse data

Bourgeois et al. Note



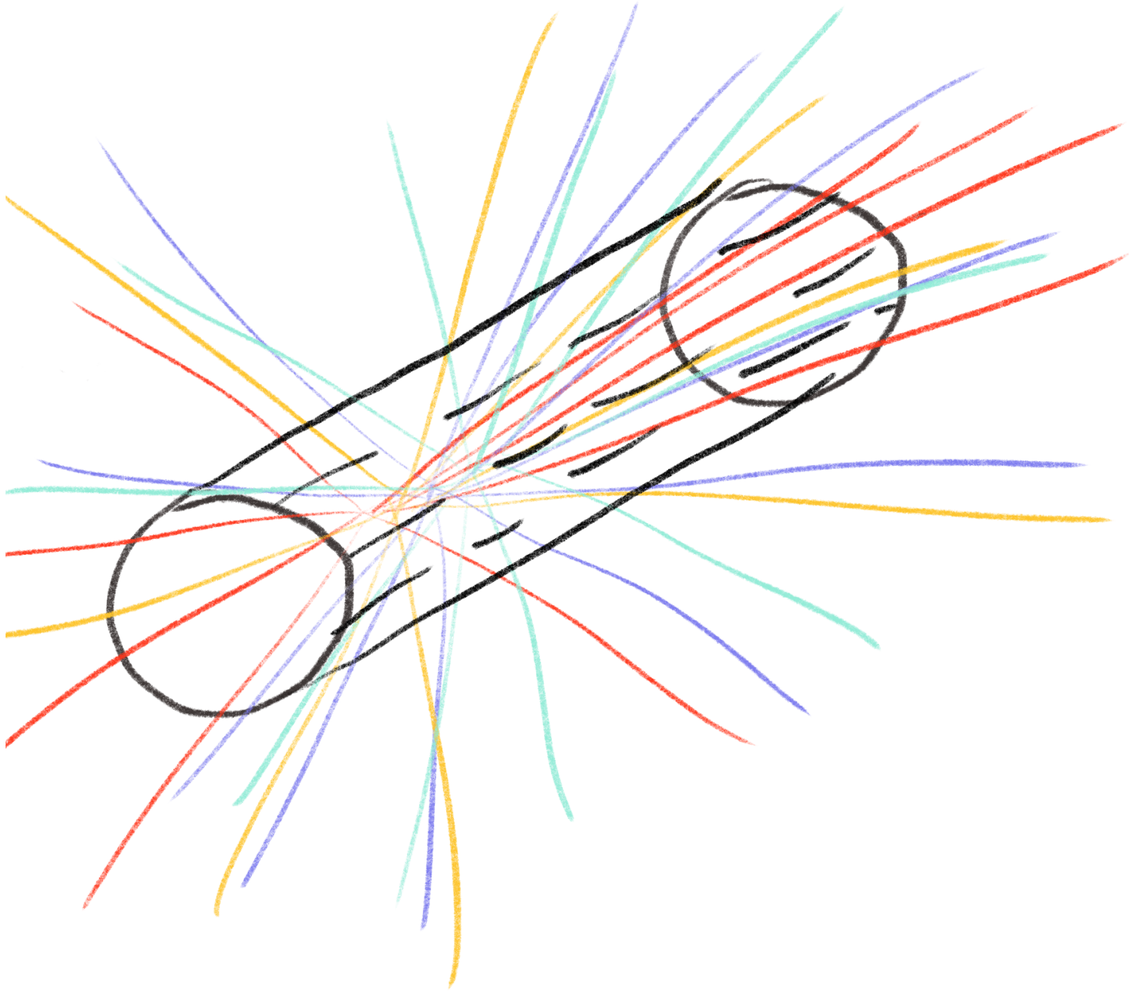
In principle parallelism, e.g. running on GPUs is where AI methods and deep NNs and so on should excel! But our data is exceptionally sparse (as low as 1% density in 3D space for some detectors), making many methods such as CNNs hard to apply.

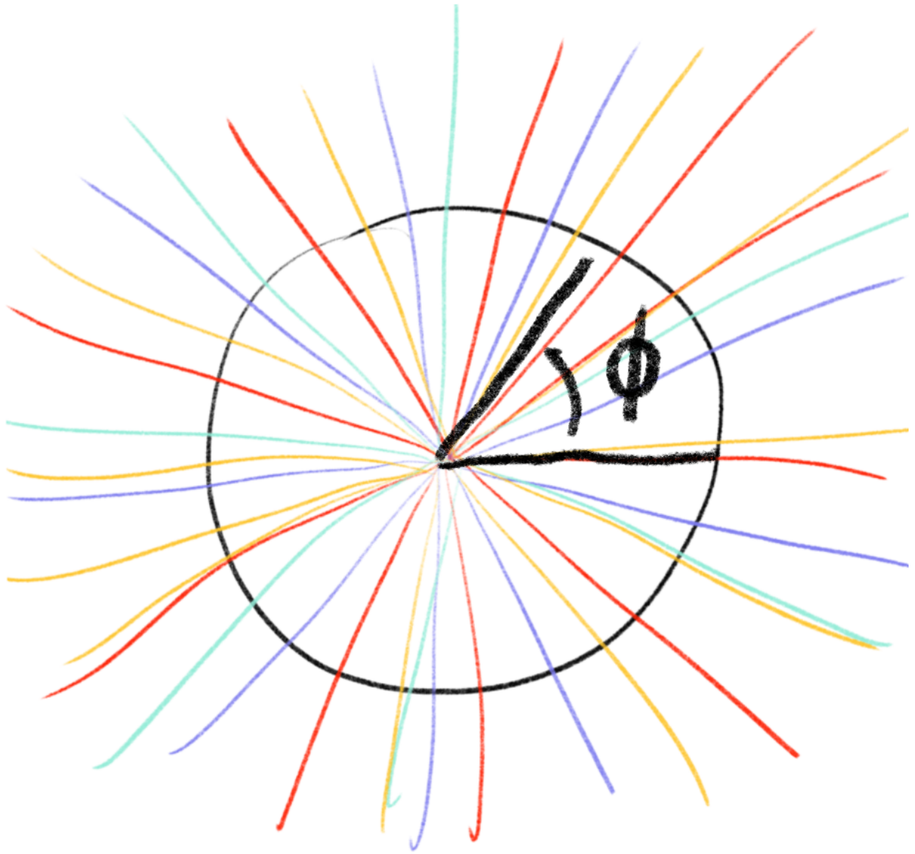
Exploiting symmetries to guide learning



Schematic view of our vertex detector which is used to find the proton collision points

Exploiting symmetries to guide learning





Schematic view of our vertex detector which is used to find the proton collision points
Particles produced in these collisions follow lines of equal φ – exploit this symmetry!

Learning to group hits into trajectories...

Truth: Prediction

0: 0.000
0: 0.000
1: 0.998
0: 0.000
1: 1.000
1: 1.000
1: 0.998
1: 0.998
1: 0.999
1: 0.999
1: 1.000
1: 0.997
1: 0.996
1: 0.994
1: 0.992
0: 0.000

Sort the hits in φ, use LSTMs to find patterns corresponding to tracks. Performance not on a par with traditional algorithms for now, but gives an idea of thinking.

...is still a work in progress

Truth: Prediction
0: 0.000
0: 0.000
1: 0.998
0: 0.000
1: 1.000
1: 1.000
1: 0.998
1: 0.998
1: 0.999
1: 0.999
1: 1.000
1: 0.997
1: 0.996
1: 0.994
1: 0.992
0: 0.000

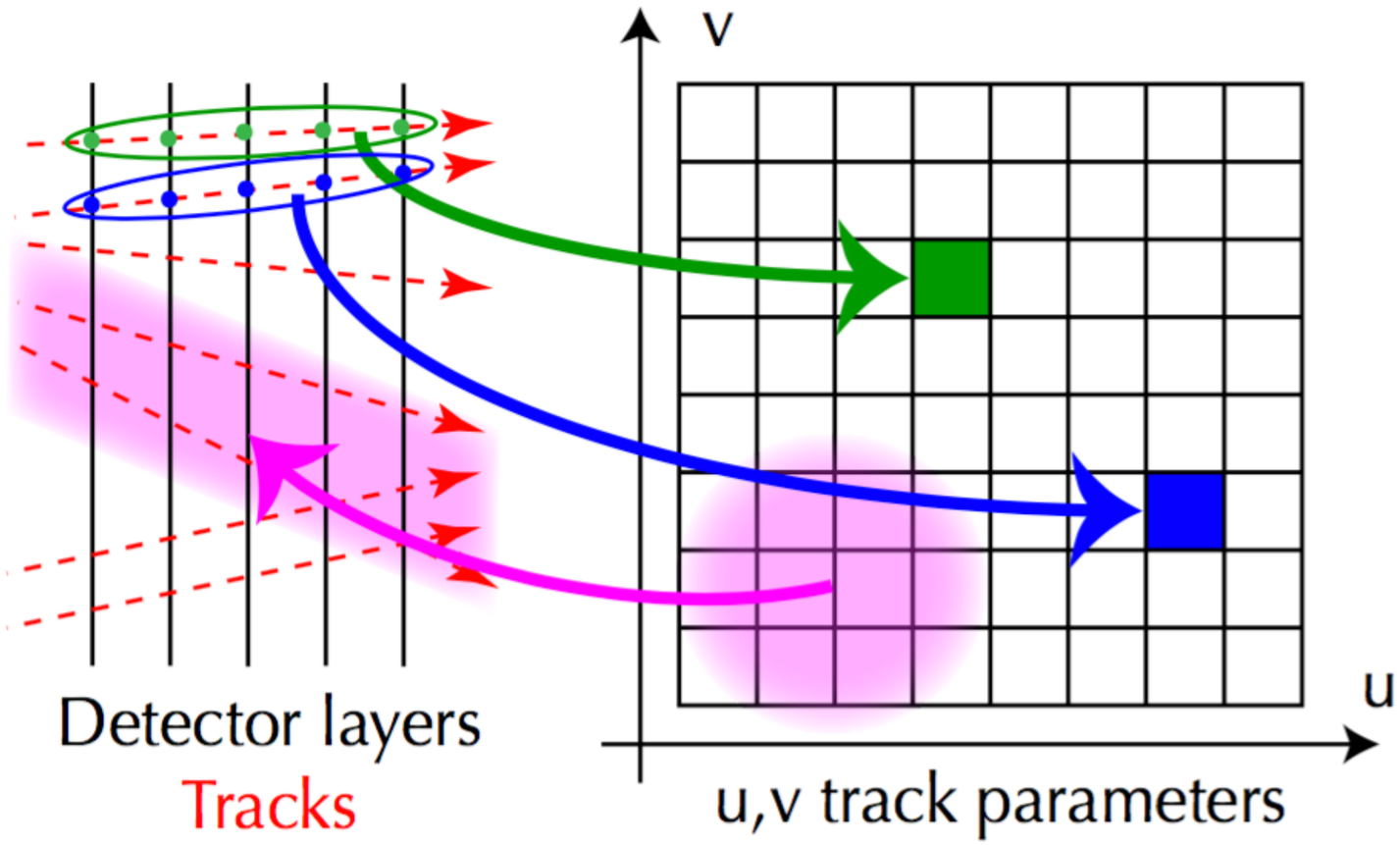
Truth: Prediction
0: 0.000
0: 0.000
0: 0.000
0: 0.000
0: 0.000
0: 0.000
0: 0.000
1: 0.165
0: 0.165
0: 0.000
0: 0.000
0: 0.196
0: 0.000
0: 0.201
1: 0.220

Cannot seem to find 3-hit tracks that are not continuous

Sort the hits in φ, use LSTMs to find patterns corresponding to tracks. Performance not on a par with traditional algorithms for now, but gives an idea of thinking.

Can custom processors help?

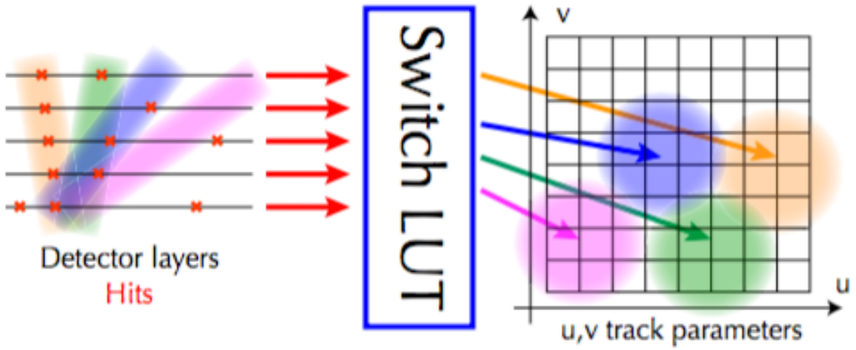
Cenci et al. CTD2016



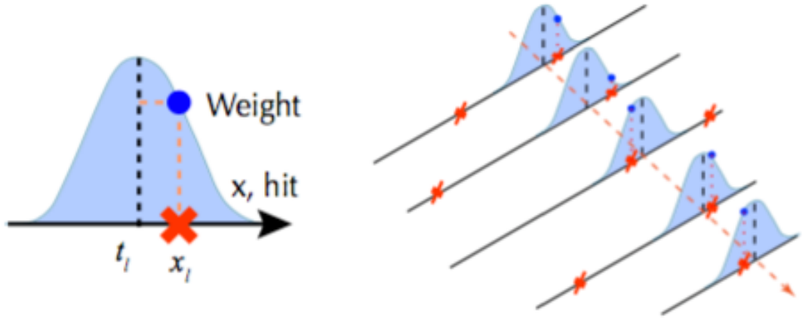
Build a custom processor using FPGAs and exploiting natural detector/physics symmetries to map hits onto particle trajectories. Enables track finding with microsecond latencies

Can custom processors help?

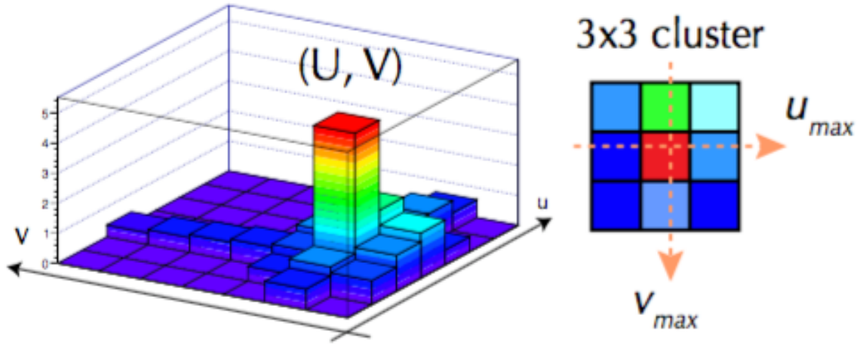
Cenci et al. CTD2016
Cenci et al. TWEPP17



(a) Step 1



(b) Step 2

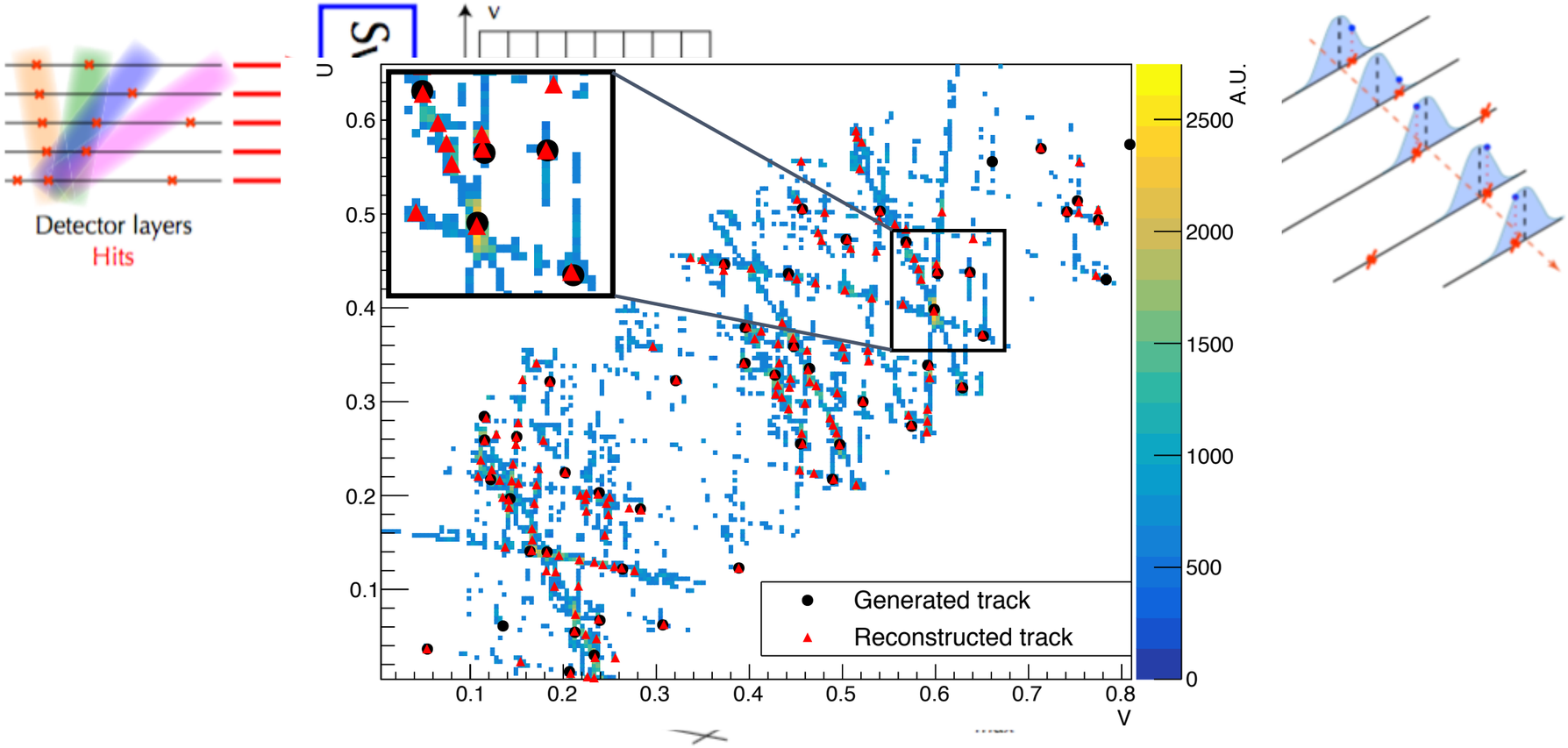


(c) Step 3

Prototype has demonstrated the ability to find tracks in a simplified setup, studies are ongoing to understand if deploying such a processor in the following years will be more globally cost-effective than traditional algorithms or x86 based solutions.

Can custom processors help?

Cenci et al. CTD2016
Cenci et al. TWEPP17

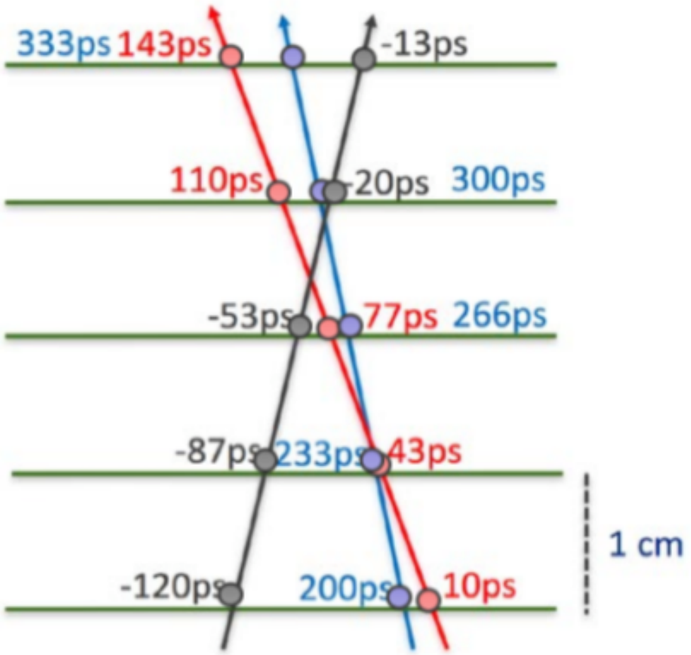


(c) Step 3

Example of track finding for the LHCb tracker using simulated events at $2*10^{33}$!
Algorithm owes its performance to an NN-like web of local connections on the FPGA.

Adding timing information

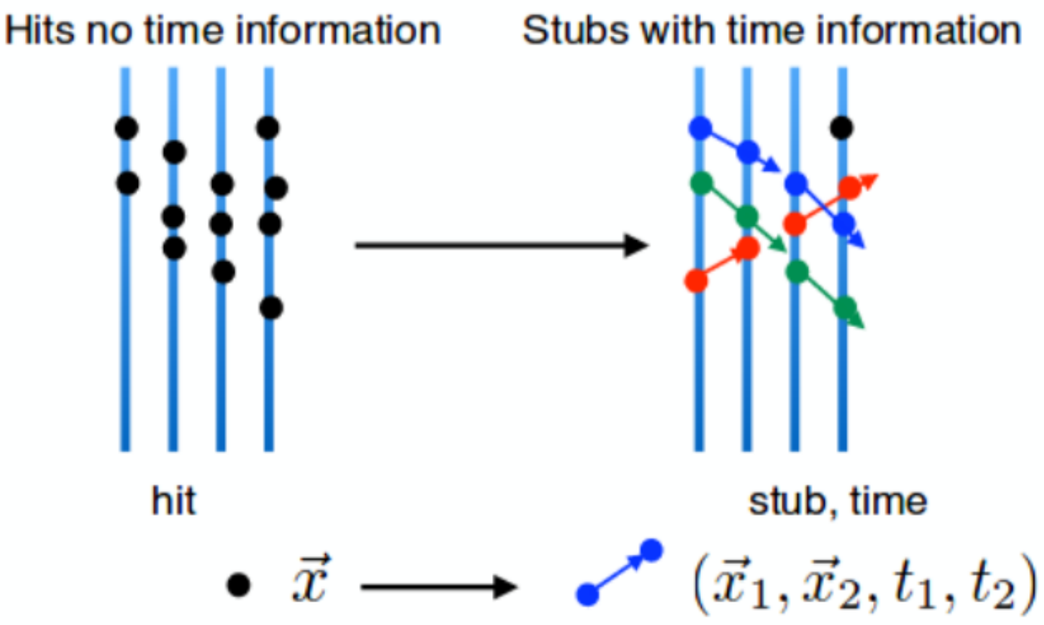
Petruzzo et al. Slides

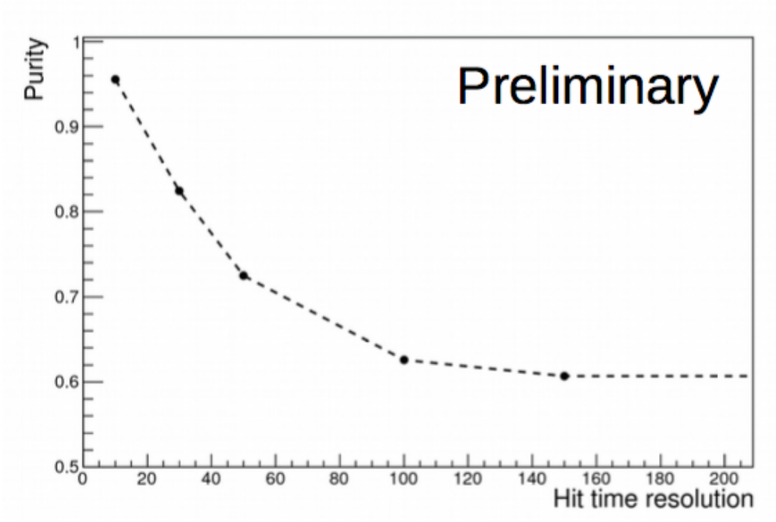


In future may need to go from 3D to 4D pattern recognition in order to separate trajectories in the detector. Studies ongoing for future LHCb upgrades within the custom processor approach with promising preliminary results.

Adding timing information

Petruzzo et al. Slides

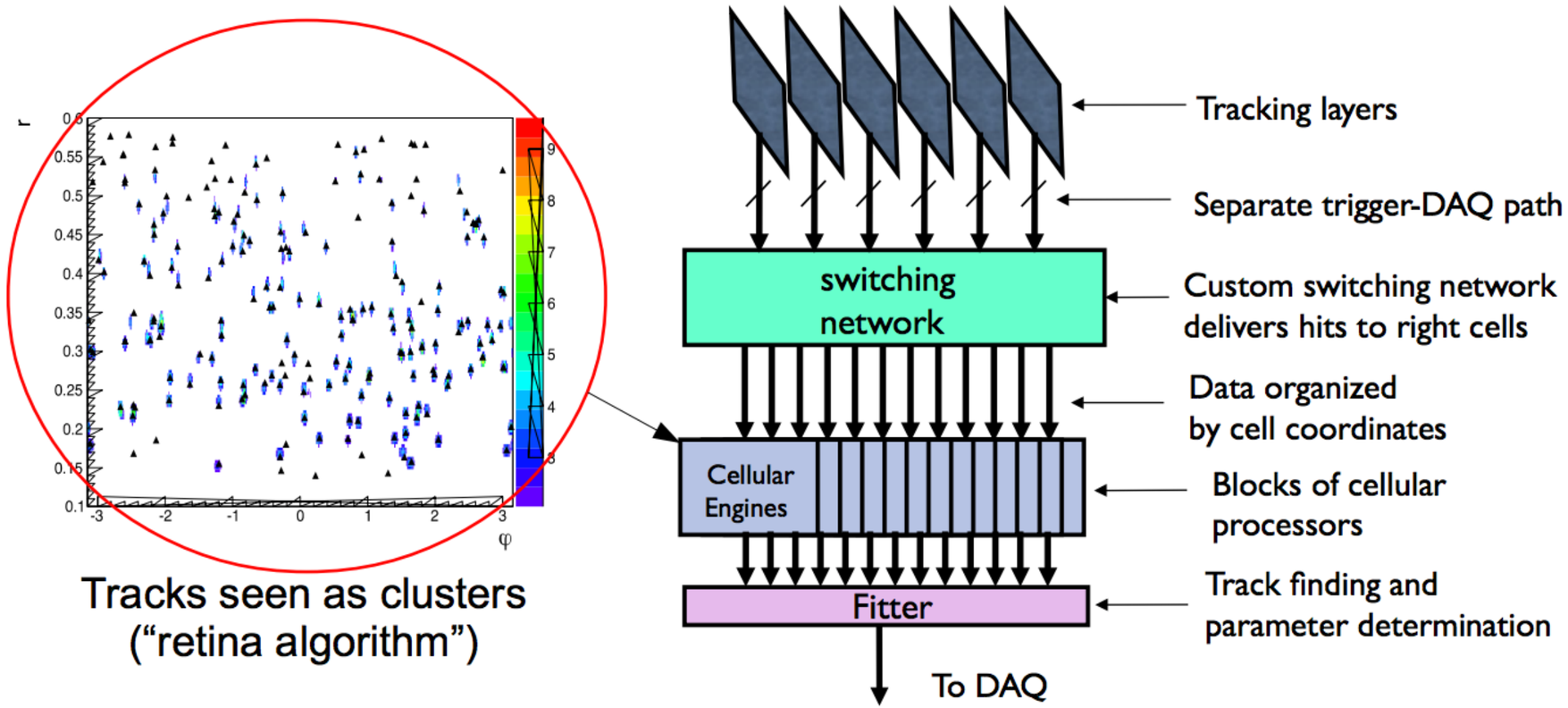




In future may need to go from 3D to 4D pattern recognition in order to separate trajectories in the detector. Studies ongoing for future LHCb upgrades within the custom processor approach with promising preliminary results.

Learning to find physics on detector?

Punzi Slides



[see talk by GP @INSTR-2014 (Novosibirsk) + related talks @WIT-2014]

Extend custom processor concept from tracking to all parts of the detector. Send only high-level "physics summaries" for further processing. Can information gained by combining information from different detectors still be obtained efficiently? Does the data volume actually reduce? Does it save resources? To be discussed in next years!

Data processing is *the* challenge

AI/ML transformed classification, is pattern recognition next?

Is AI/ML key to efficient use of massively parallel computing?