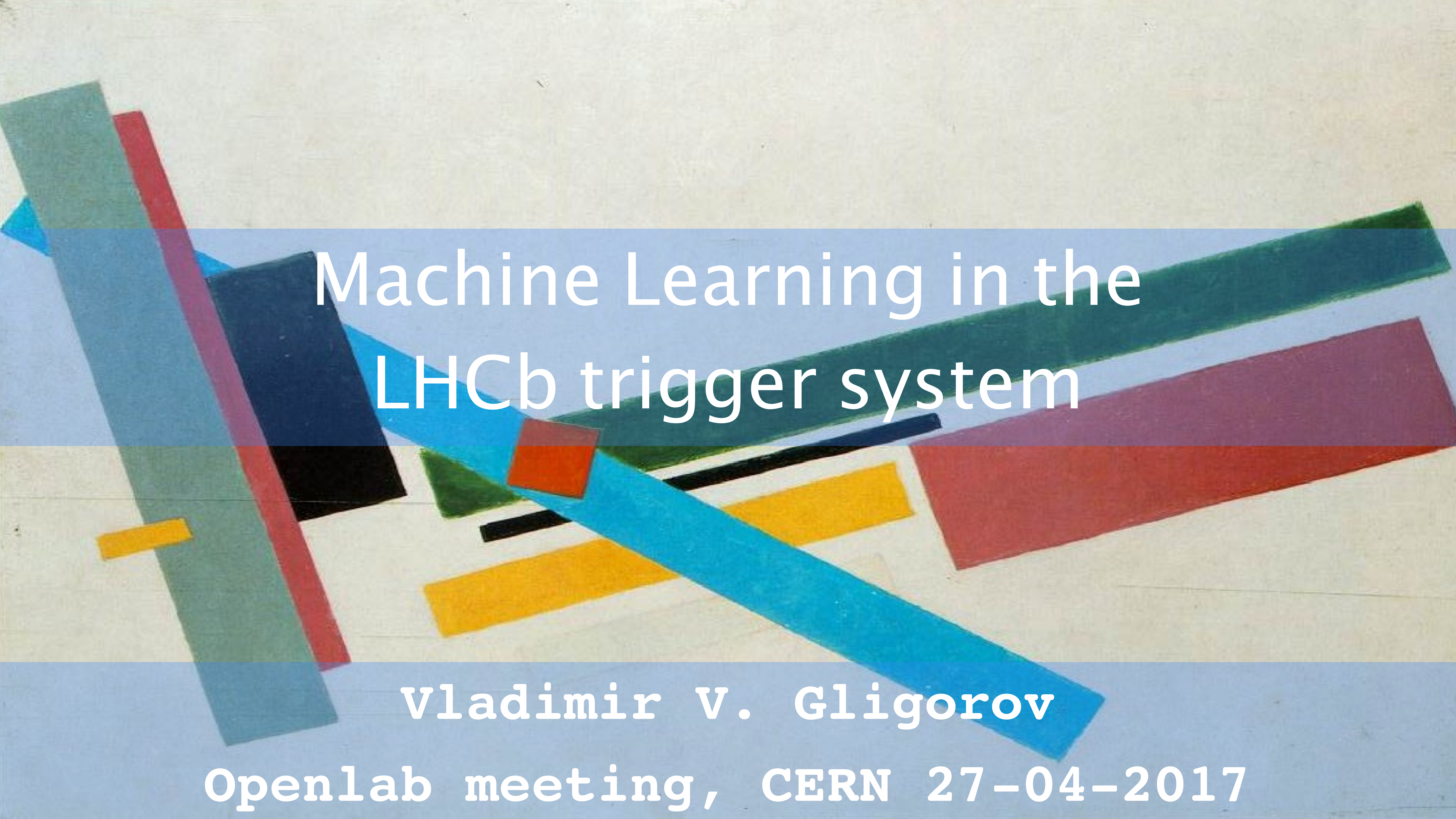


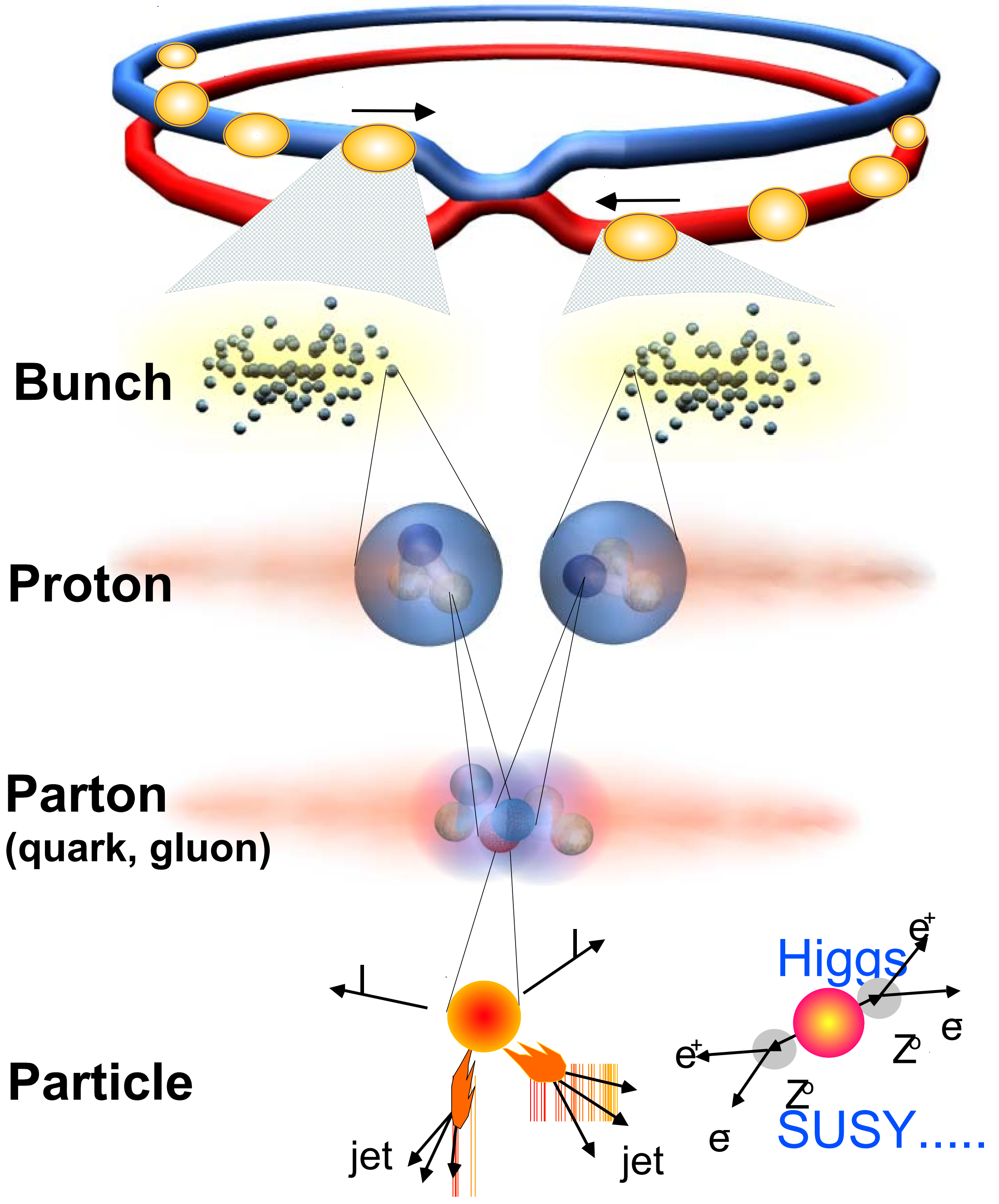
Machine Learning in the LHCb trigger system

Vladimir V. Gligorov

Openlab meeting, CERN 27-04-2017

Triggers : the heartbeat of the LHC

Collisions at the LHC: summary



Proton - Proton	2804 bunch/beam
Protons/bunch	10^{11}
Beam energy	7 TeV (7×10^{12} eV)
Luminosity	$10^{34}cm^{-2}s^{-1}$

Bunch crossing rate : 15-30 MHz

Between 1-200 proton-proton collisions per crossing (depends on experiment).

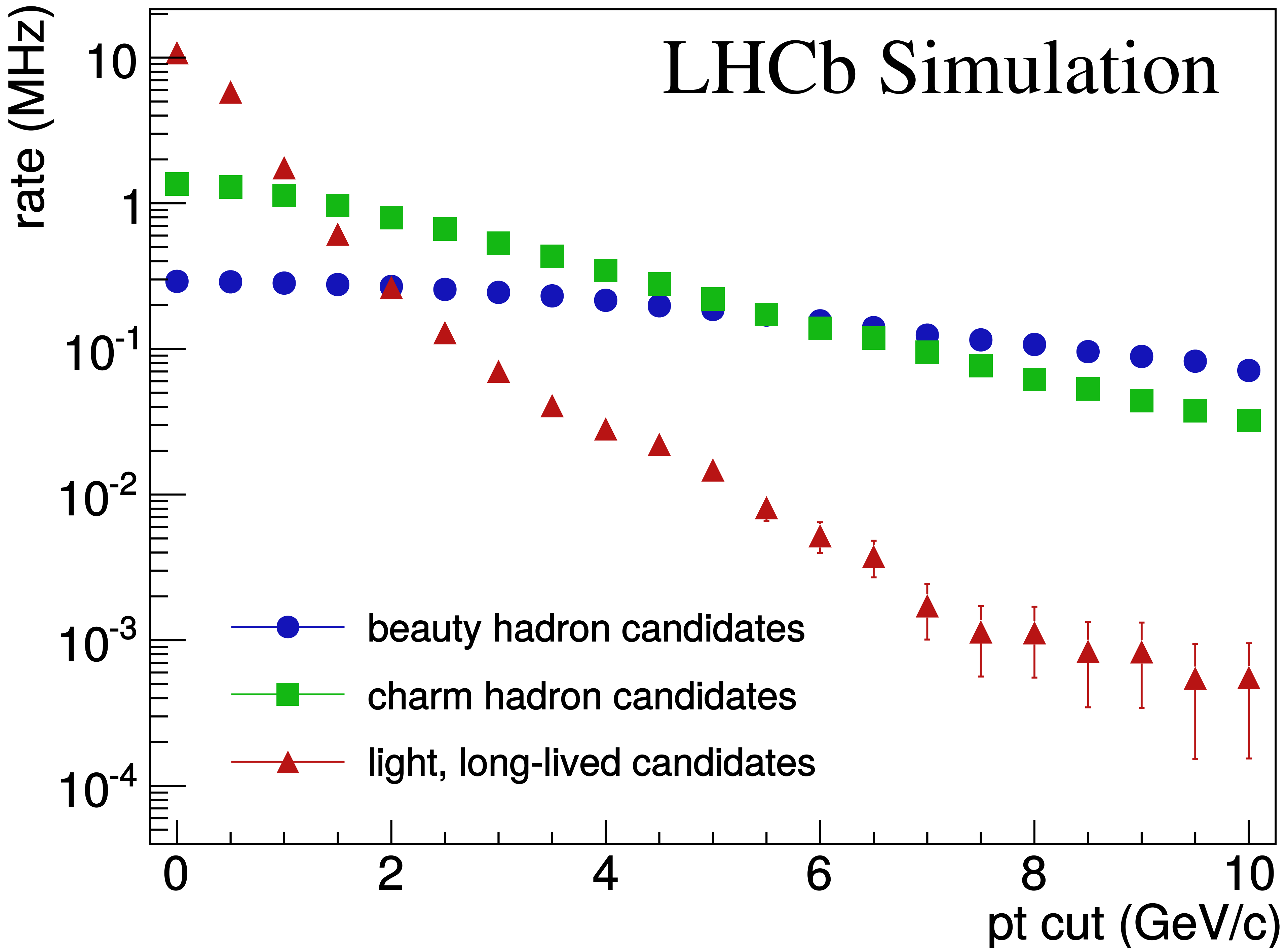
New physics rate ≈ .00001 Hz

Event selection:
1 in 10,000,000,000,000

Triggers today

Enter the MHz signal era

Fitzpatrick&Gligorov
LHCb-PUB-2014-027



In 2020, LHCb will increase the data rate by a factor 5 : at this point, almost every collision in the detector becomes interesting for physics analysis!

Triggers today

Real-time data analysis tomorrow

Short version of this talk

LHCb is moving most data analysis to the trigger

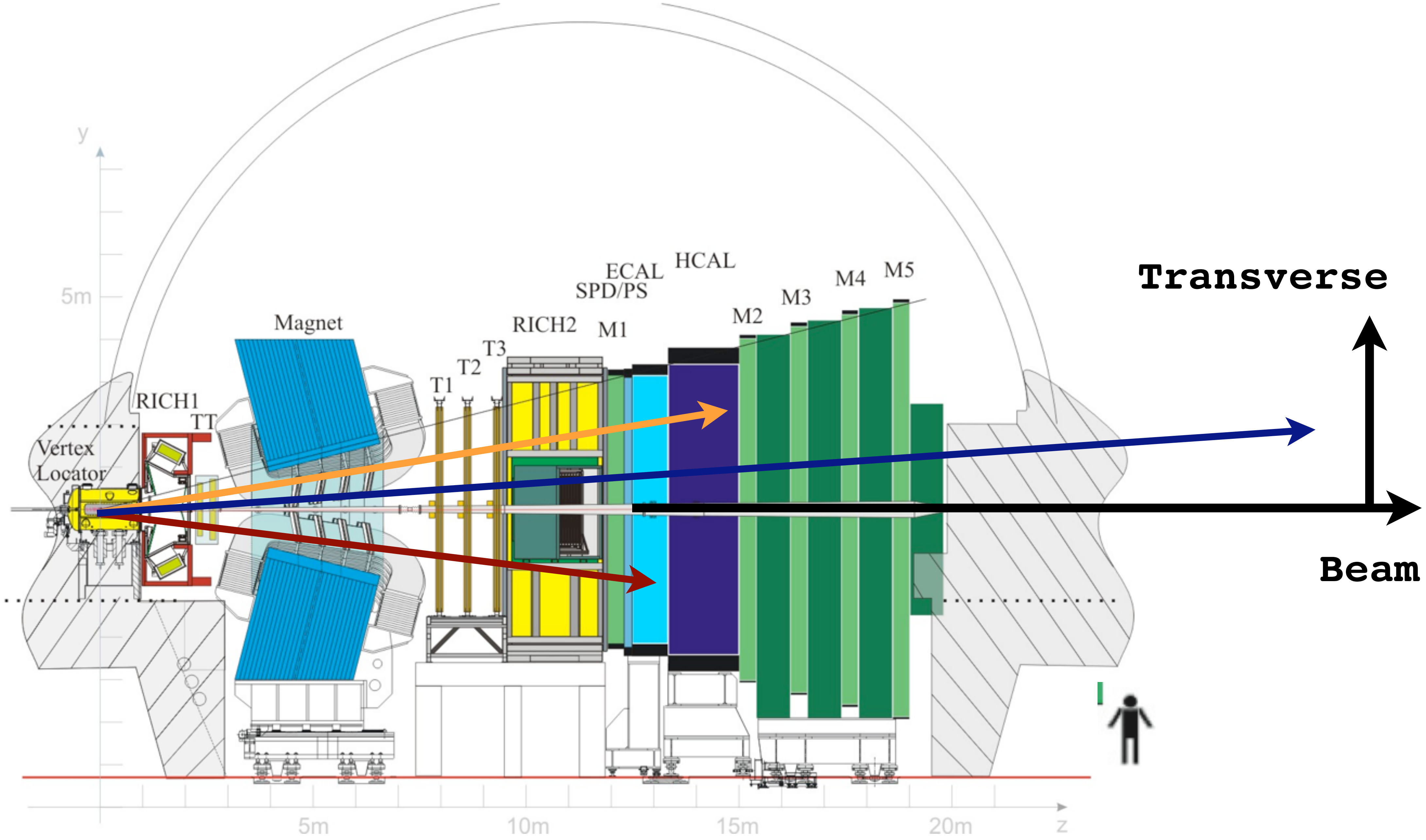
Almost all published LHCb analyses use at least a BDT

Ergo it either already is, or soon will be, ML/MVA based

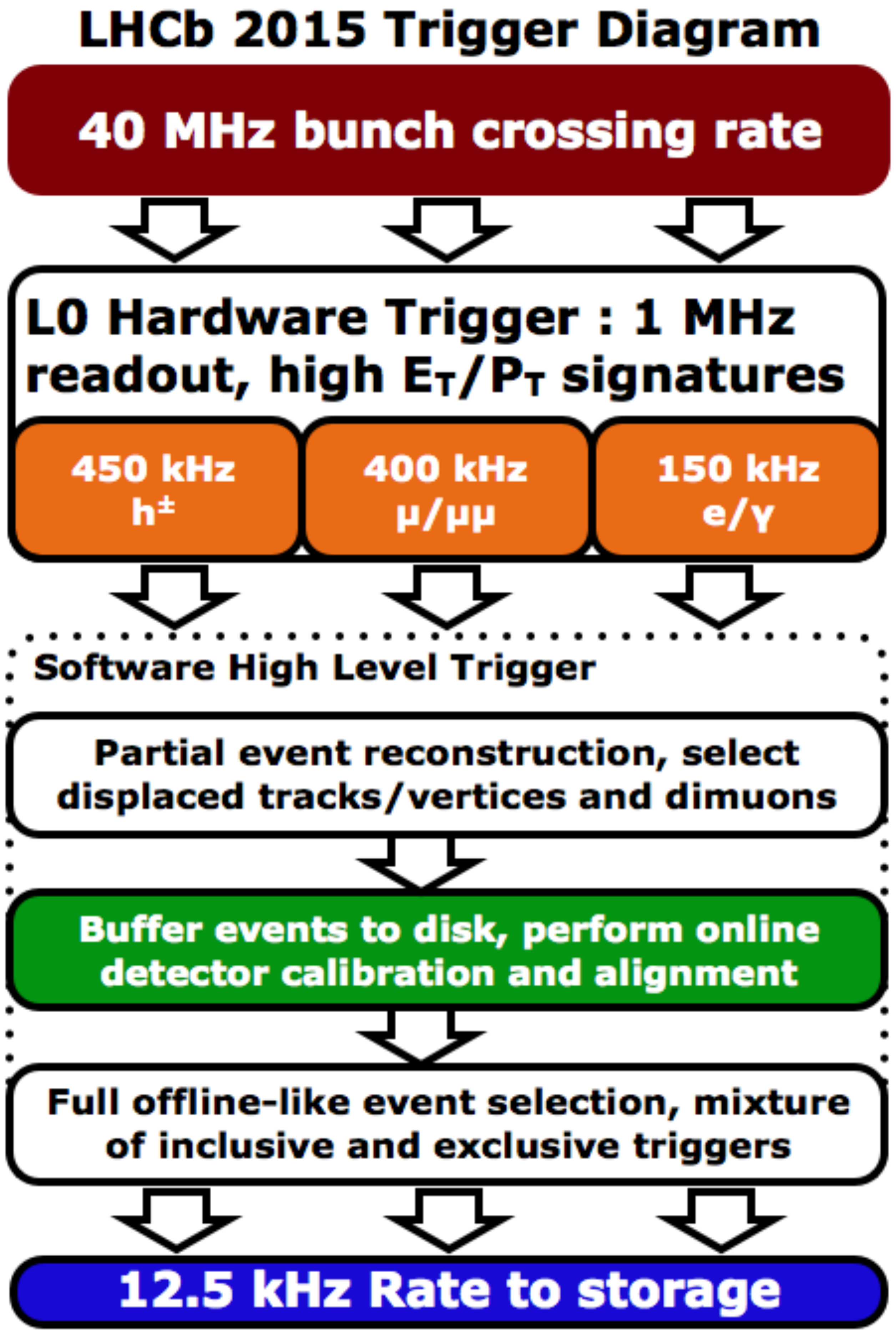
More interesting question : how?

LHCb

- ELECTRONS
- PHOTONS
- HADRONS
- MUONS



p_T = Transverse momentum
E_T = Transverse energy



See LHCb-PROC-2016-021

Where do we use ML/MVA today?

Within trigger, only in HLT. Some attempt made at using neural nets in hardware calorimeter trigger but no significant gain achieved, abandoned.

Reconstruction :

– Track reconstruction	DNN & BDT (in places)
– Fake track identification	NEURAL NET
– Particle identification	NN/BDT (in places)

Selection :

– First level inclusive charm/beauty triggers	FISHER, BDT
– Second level inclusive beauty triggers	BDT
– Second level inclusive D^* triggers	BDT
– Second level inclusive radiative decay triggers	BDT

Incidentally : these BDT triggers account for about 66% of the first-level trigger rate and around 30-40% of second-level trigger rate.

About 60% of Run I papers produced using BDT based trigger.

Key questions for using MVA in trigger

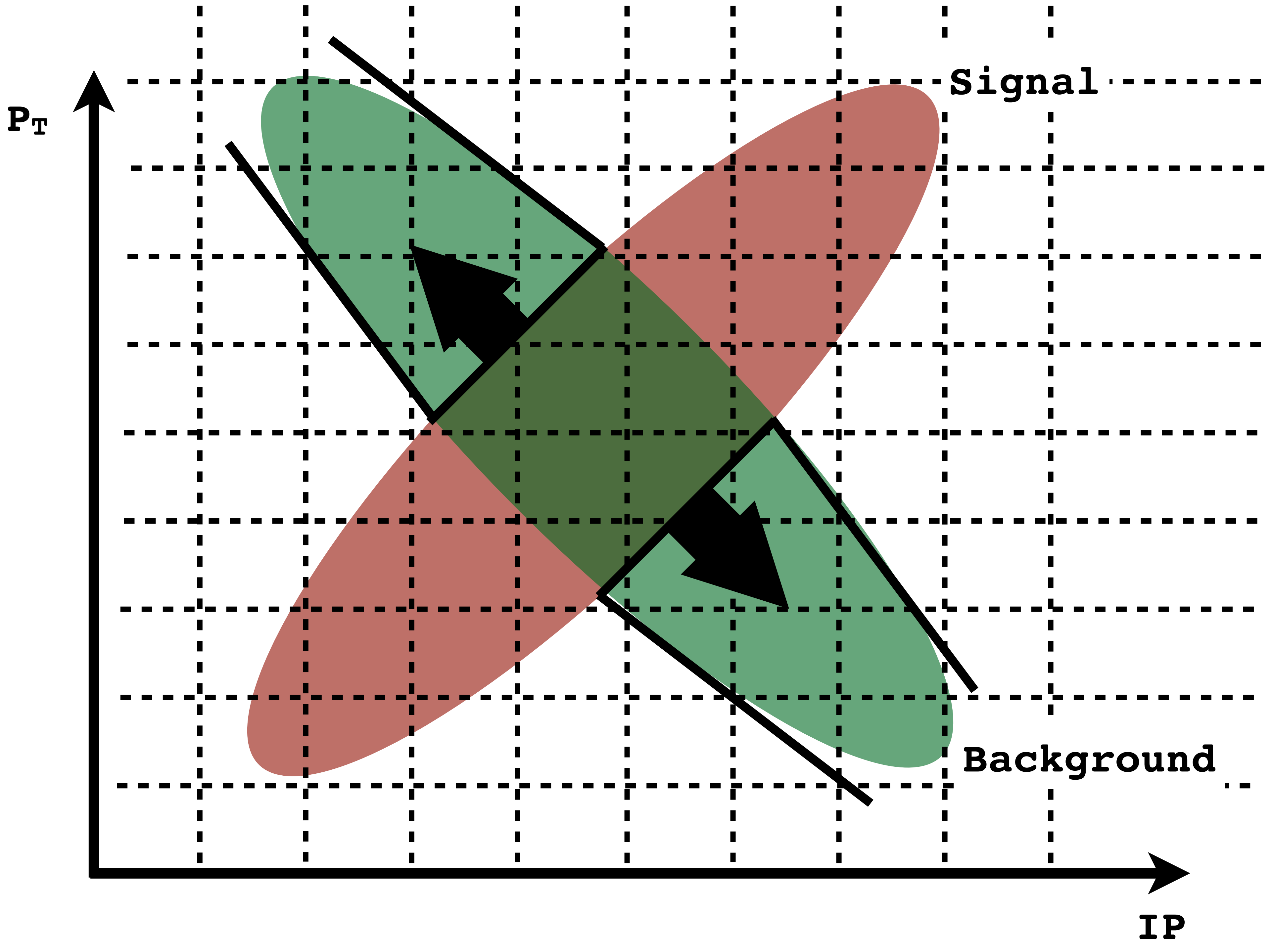
How to ensure stability wrt. varying detector conditions?

How to make the evaluation of the MVA instantaneous?

Nota bene : since the full event reconstruction is done upfront, the cost of feature building is not relevant for us.

A bonsai boosted decision tree

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



Gligorov&Williams http://arxiv.org/abs/1210.6861

A bonsai boosted decision tree

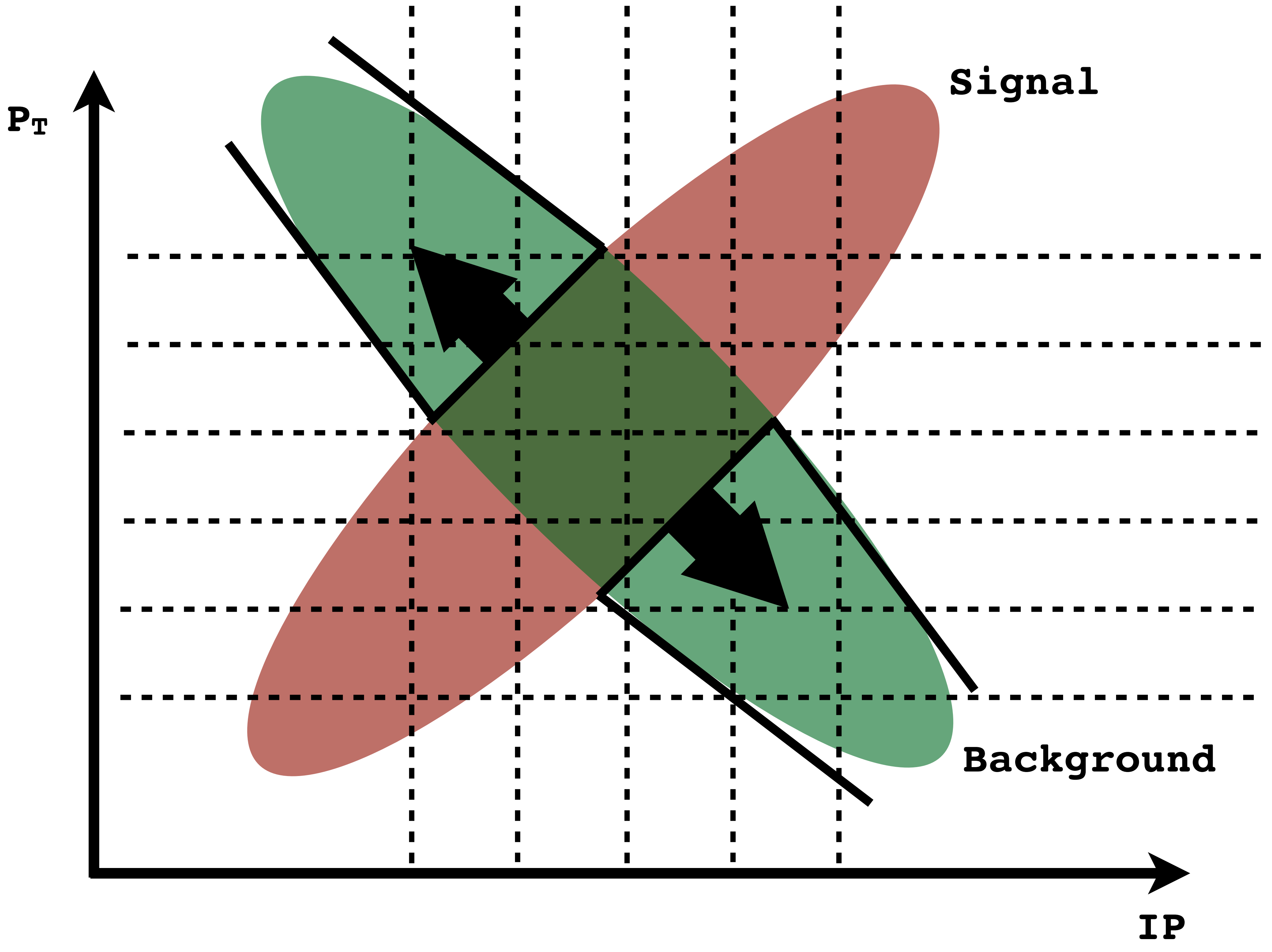
Consider a two-variable boosted decision tree : this is like a binned selection where the BDT algorithm picks the optimal bin sizes and boundaries

THEREFORE : discretize the variables yourself!

=> Pick a bin size (variable) based on the detector resolution.

=> Makes sure that the trigger is insensitive to resolution fluctuations.

=> Transforms the trigger into a 1D lookup table making it essentially infinitely fast.



Gligorov&Williams http://arxiv.org/abs/1210.6861

Topological performance

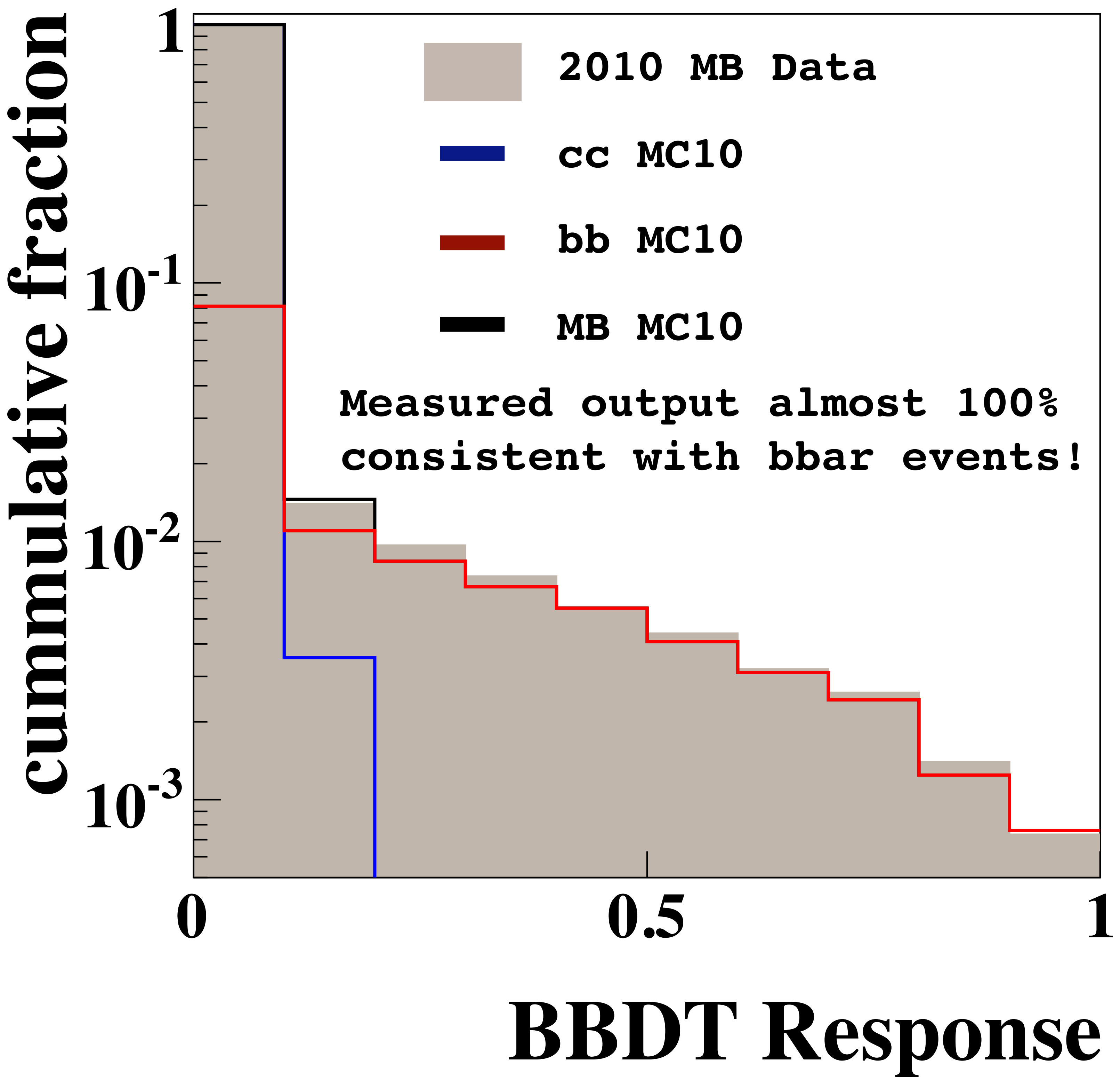
MVA applied after a cut-based preselection of secondary vertices, which is necessary to reduce the timing cost of forming the secondary vertices in the first place.

Prune less discriminating features, trade some absolute performance for simplicity.

Remarkable data/MC agreement in the BDT response function for bbar signatures.

No variation in performance during Run I (beyond intended retunings/improvements)

Note : heard a few times that this is an "obvious" idea and that's true today, but was not so obvious in HEP in 2010.



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

Gligorov&Williams http://arxiv.org/abs/1210.6861

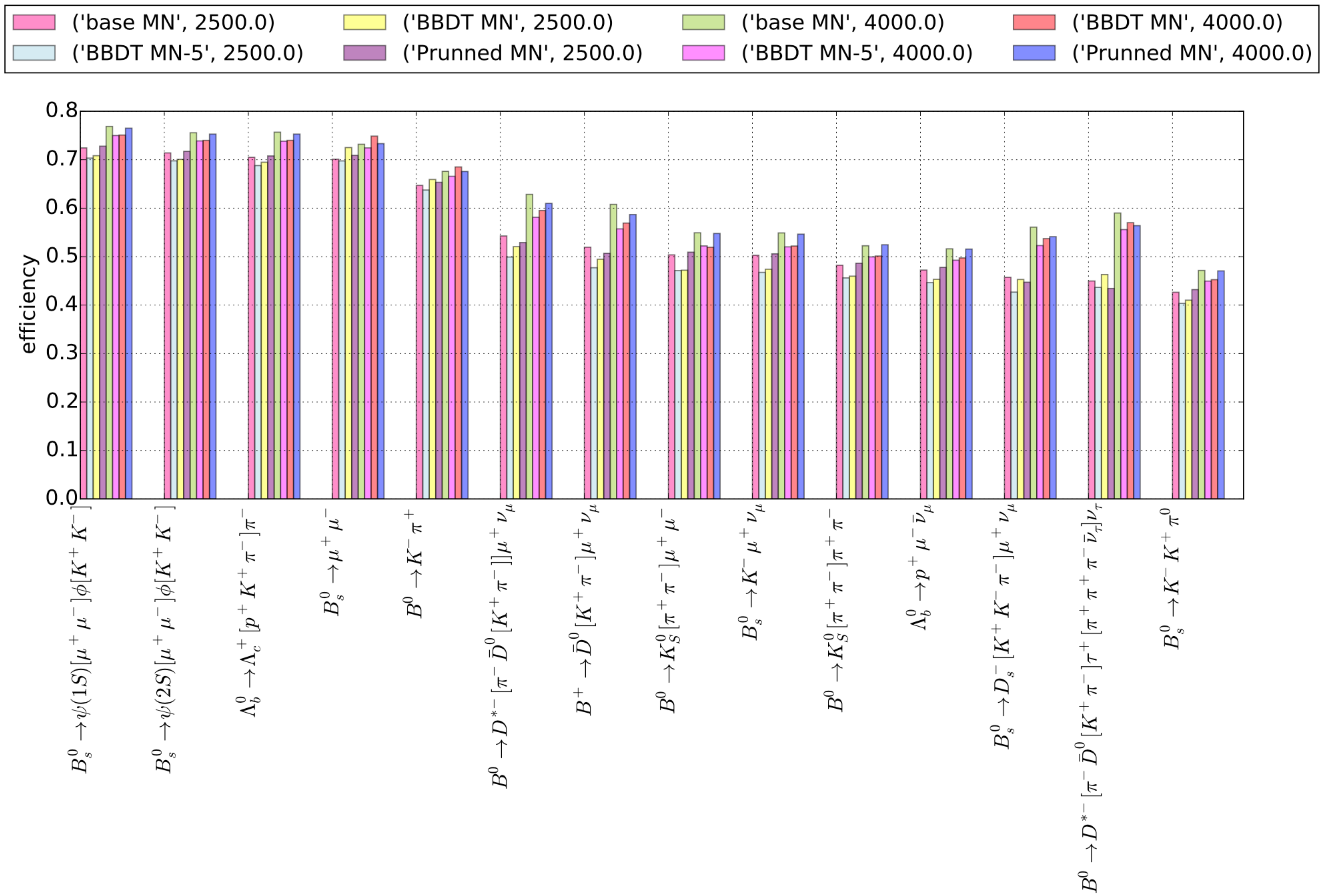
Implementation details

BDT weight files versioned separately to the code, which accesses the appropriate version at run-time.

Takes O(50 MB) loaded into memory as a 1D lookup table, but this goes in the shared memory so less of an impact when multithreading

Framework for developing/training/deploying new BDT based triggers minimizes effort for analysts

Topological reoptimization for 2015



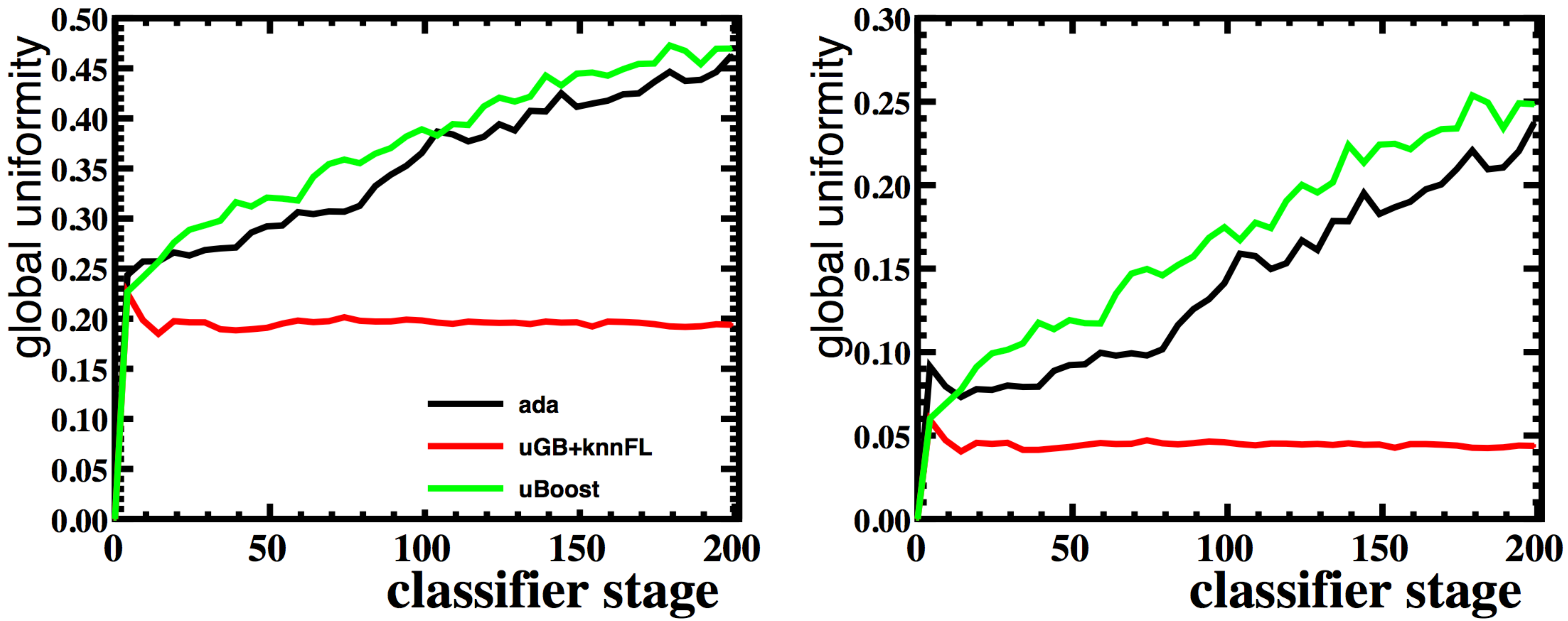
Compared numerous classifier types and configurations

Start with large number of features and deliberately prune down to 6-7 maximally discriminating variables; sacrifice some % of maximal performance for simplicity.

Gain 10-60% efficiency relative to the Run I trigger (depends on decay mode) for an equivalent output rate.

Likhomanenko et al. http://cds.cern.ch/record/2019813?ln=en

Yes but won't my classifier bias X,Y,... ?



We have developed classifiers which can be tuned to have a uniform efficiency or background rejection with respect to any variable/feature of interest for a marginal (few percent) loss in absolute performance. Not used yet in HLT as no need for now, but will probably be used by exclusive triggers in future.

Rogozhnikov et al. https://inspirehep.net/record/1322385

MVA in reconstruction

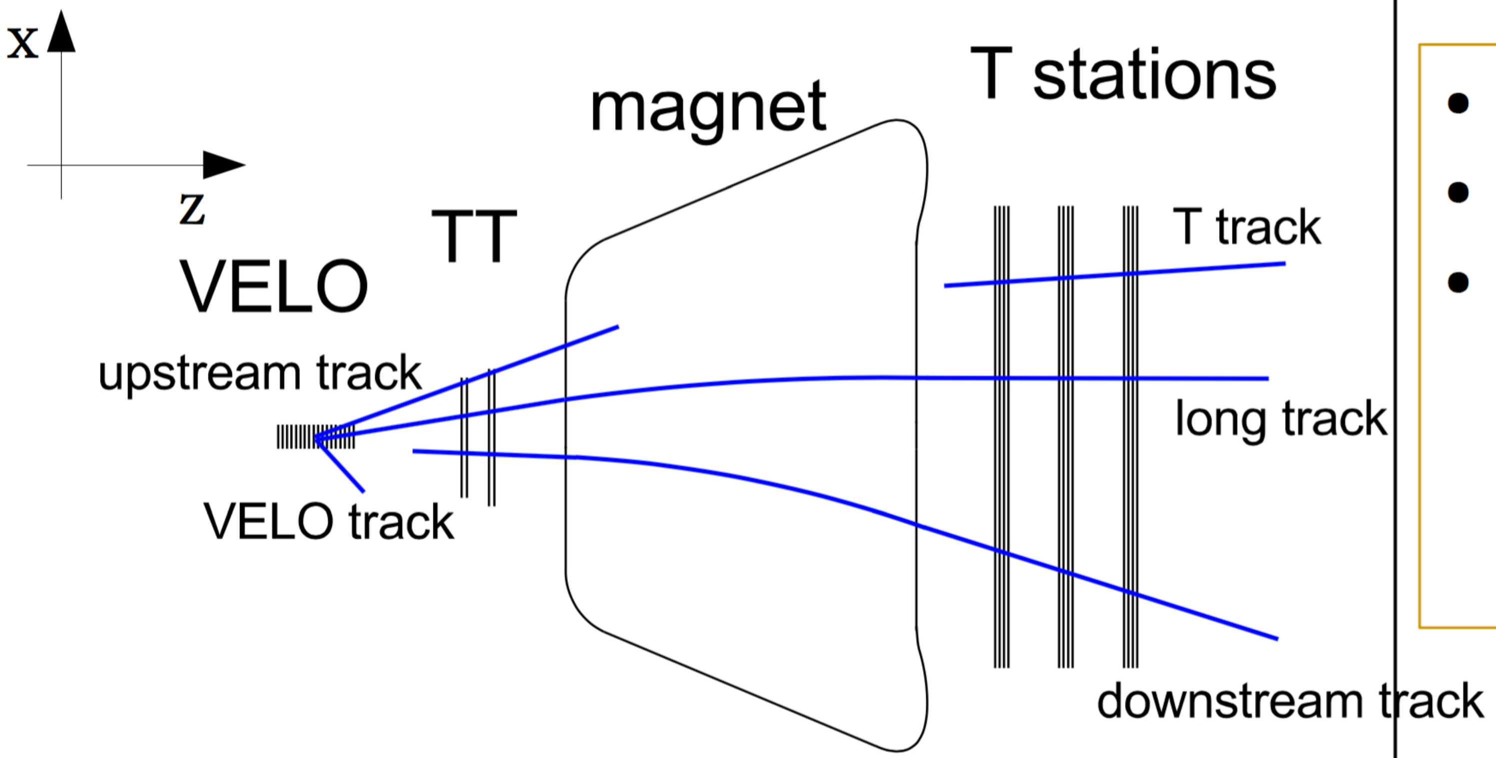
Using MVA for selections cool in 2010, now standard

Objective is now to introduce ML methods in the reconstruction of the detector

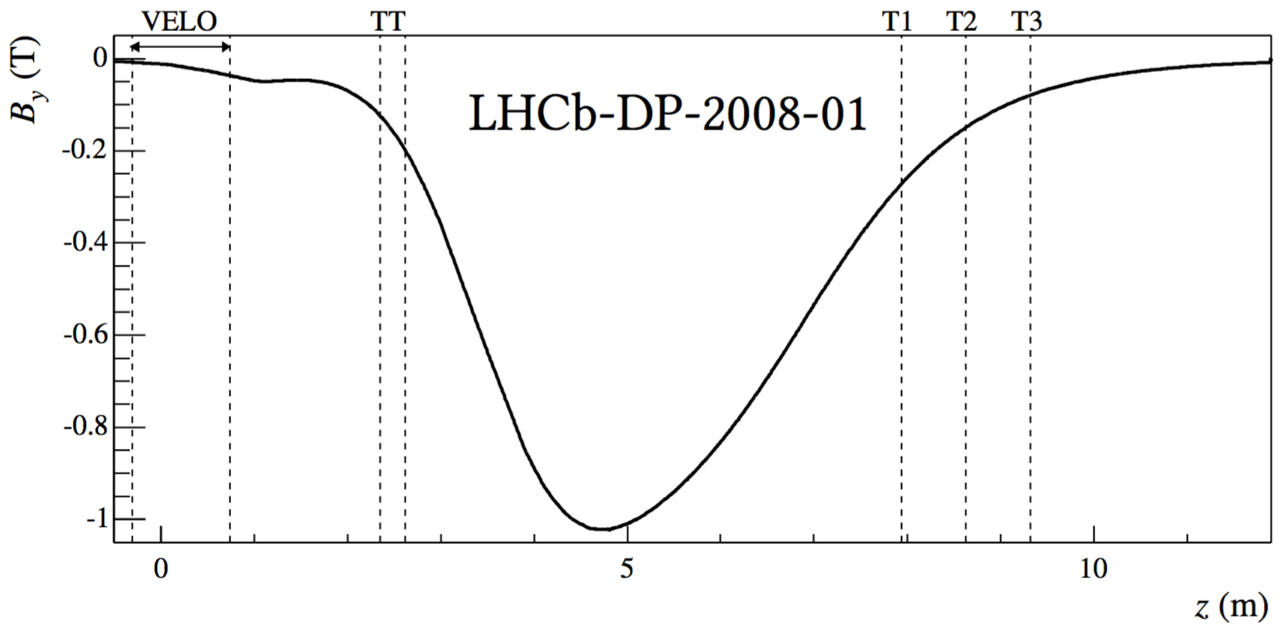
Will just give a couple of examples here, in the pattern recognition and fake track rejection

LHCb tracking 1-slide

Stolen from S. Stahl's talk to DS@HEP 2016 workshop at the Simons Foundation



Dipole magnet:

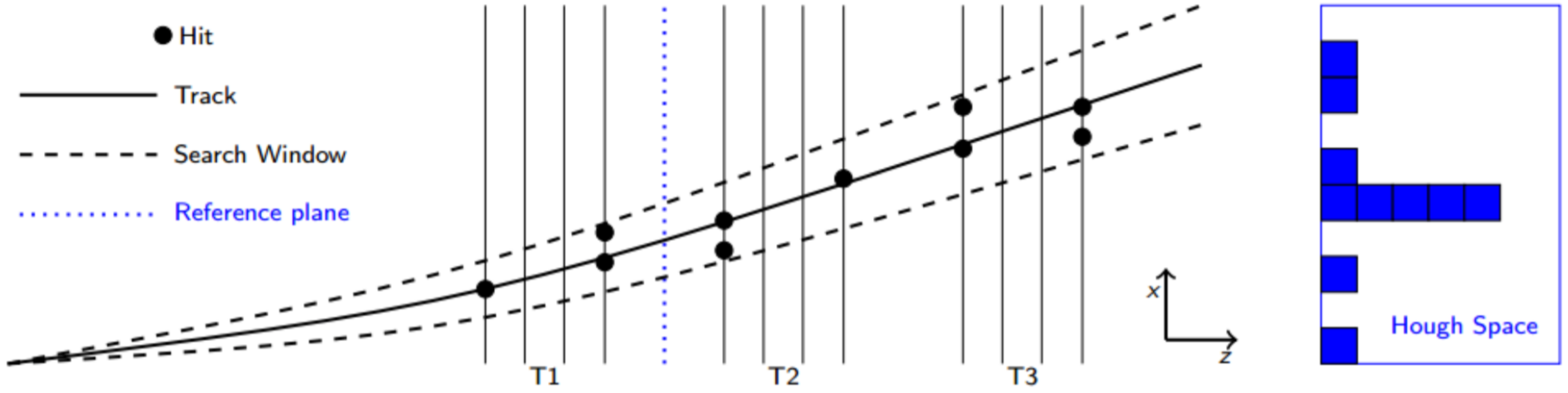


- Velo: Silicon strips (r,φ sensors)
- TT: Silicon strips (x, ±5° sensors)
- T-Stations
  - Inner region: Silicon strips
  - Outer region: Drift tubes
  - (x, ±5° sensors)

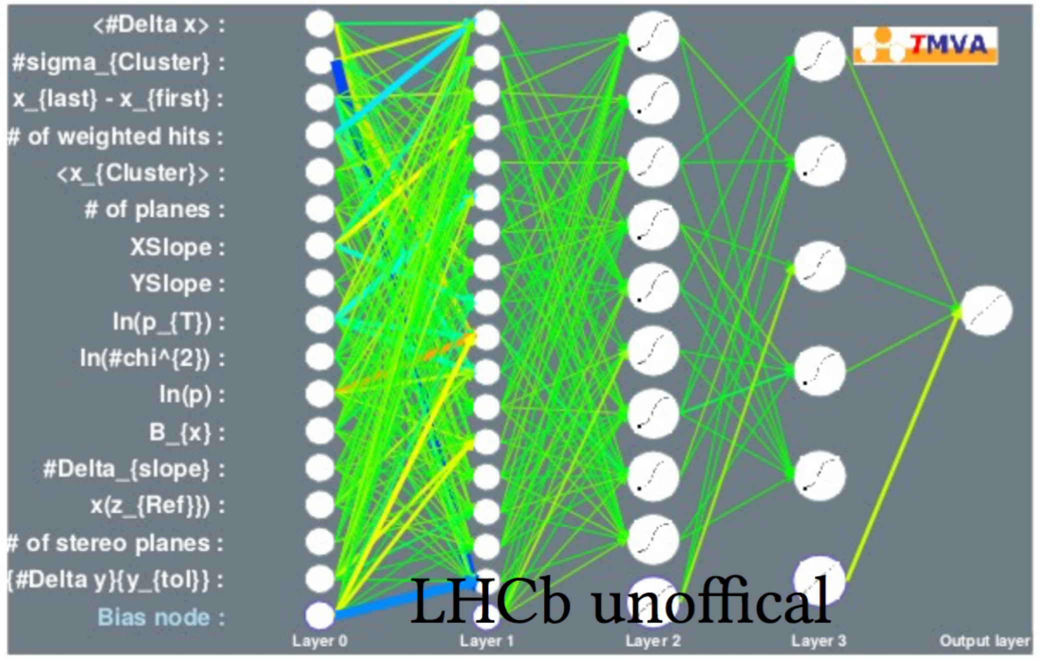
- Velo tracks:
  - Vertex reconstruction
- Long tracks:
  - Main input to analyses
- Downstream tracks:
  - e.g. for K_s and Lambda decay products

Using ML to speed it up

Stolen from S. Stahl's talk to DS@HEP 2016 workshop at the Simons Foundation



- Trained two neural networks:
  - For rejection of bad clusters with only 4 out of 6 layers
  - For final track selection
- Type:
  - MLP with hidden layers
  - Trained for background rejection at given efficiency (97% or 99 %)



Using ML to speed it up

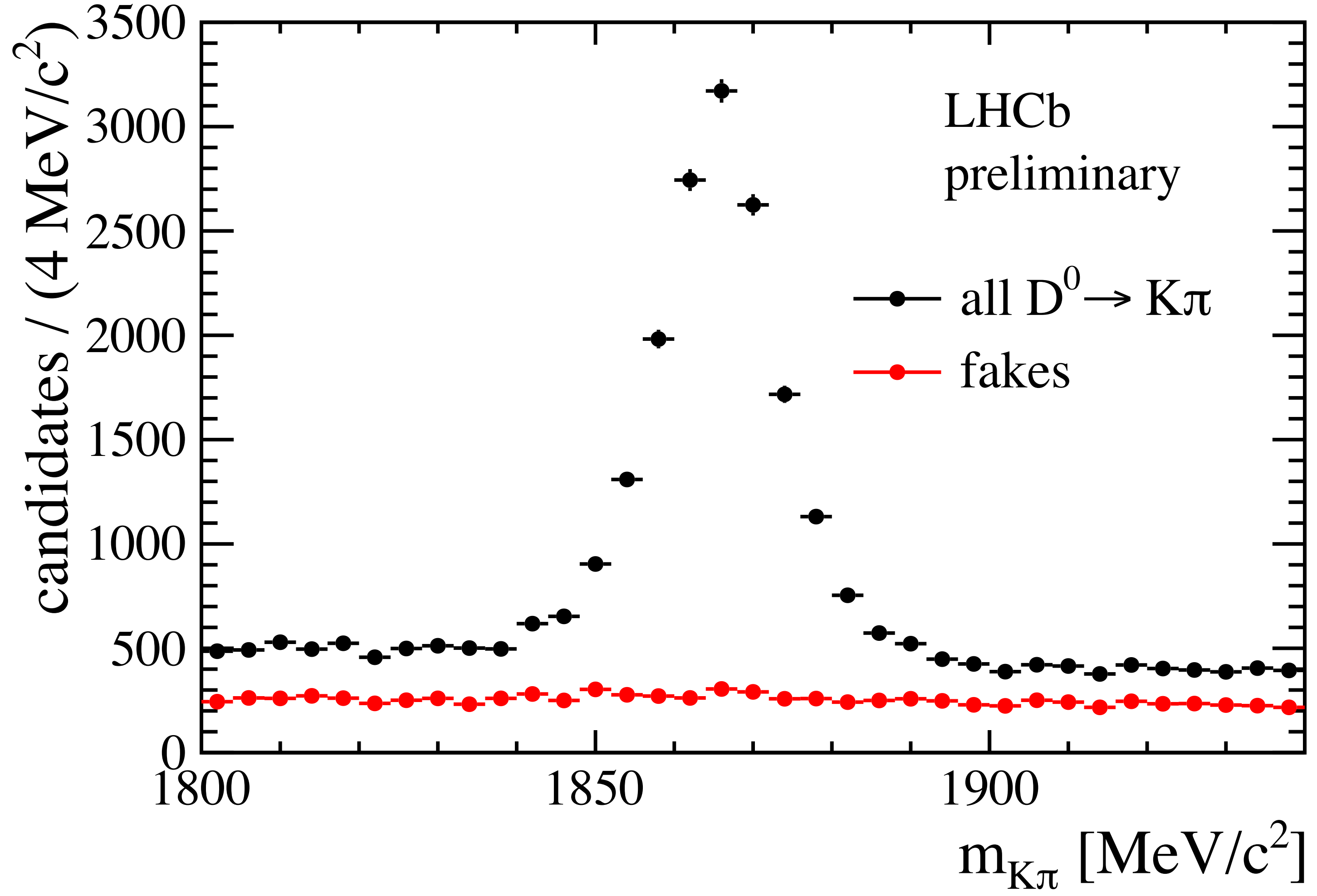
Stolen from M. Stahl's CHEP talk

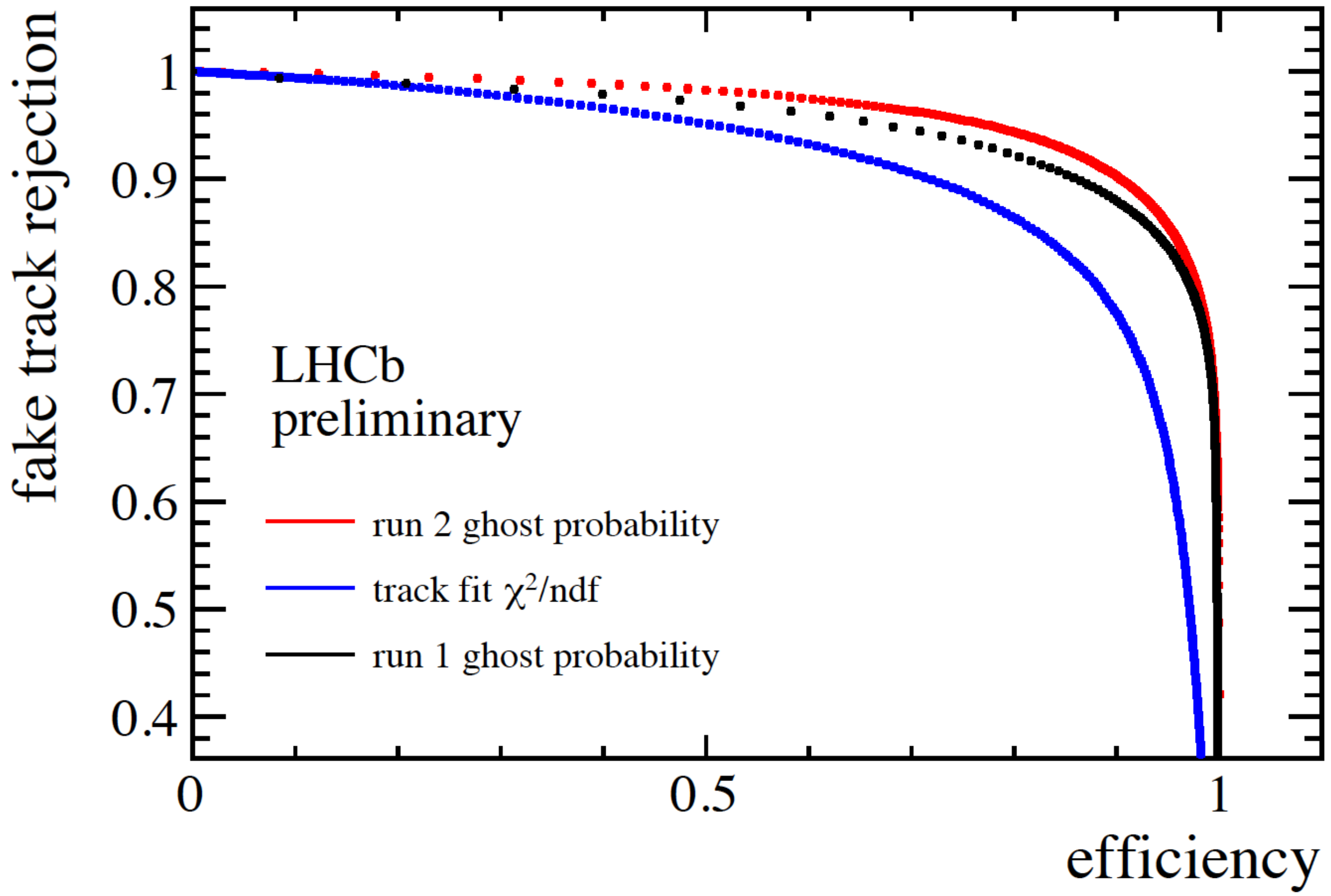
MC performance 2016 w.r.t. 2015	ν =1.6 w/ RL	ν =1.6 w/o RL
timing HLT1	± 0 %	
timing HLT2	$+4$ %	-38 %
fake rate	-27 %	-35 %
fake rate HLT1	-15 %	
ε long	$+0.5$ %	$+0.1$ %
ε long from B	$+0.2$ %	-0.2 %
$\varepsilon_{\text{HLT1}}$ long from B $p>3, p_T>0.5$ GeV	$+0.1$ %	

- Results:
  - Increased efficiency
  - Reduced fake rate considerably
  - Decreased speed compensated in later stages due to fake track rejection
  - NNs only contribute 2 % (HLT2), 0.5 % (HLT1) to timing of forward tracking algorithm

ML for fake track probability

De Cian et al.
LHCb-PUB-2017-011





Fake track probability based on TMVA NN (CE estimator), most important features are hit multiplicities and partial chi2 information in different tracking subdetectors. Main timing cost network evaluation, custom activation function for speed. Extensive use of code profiling and autovectorization to optimize the .C output of TMVA for speed.

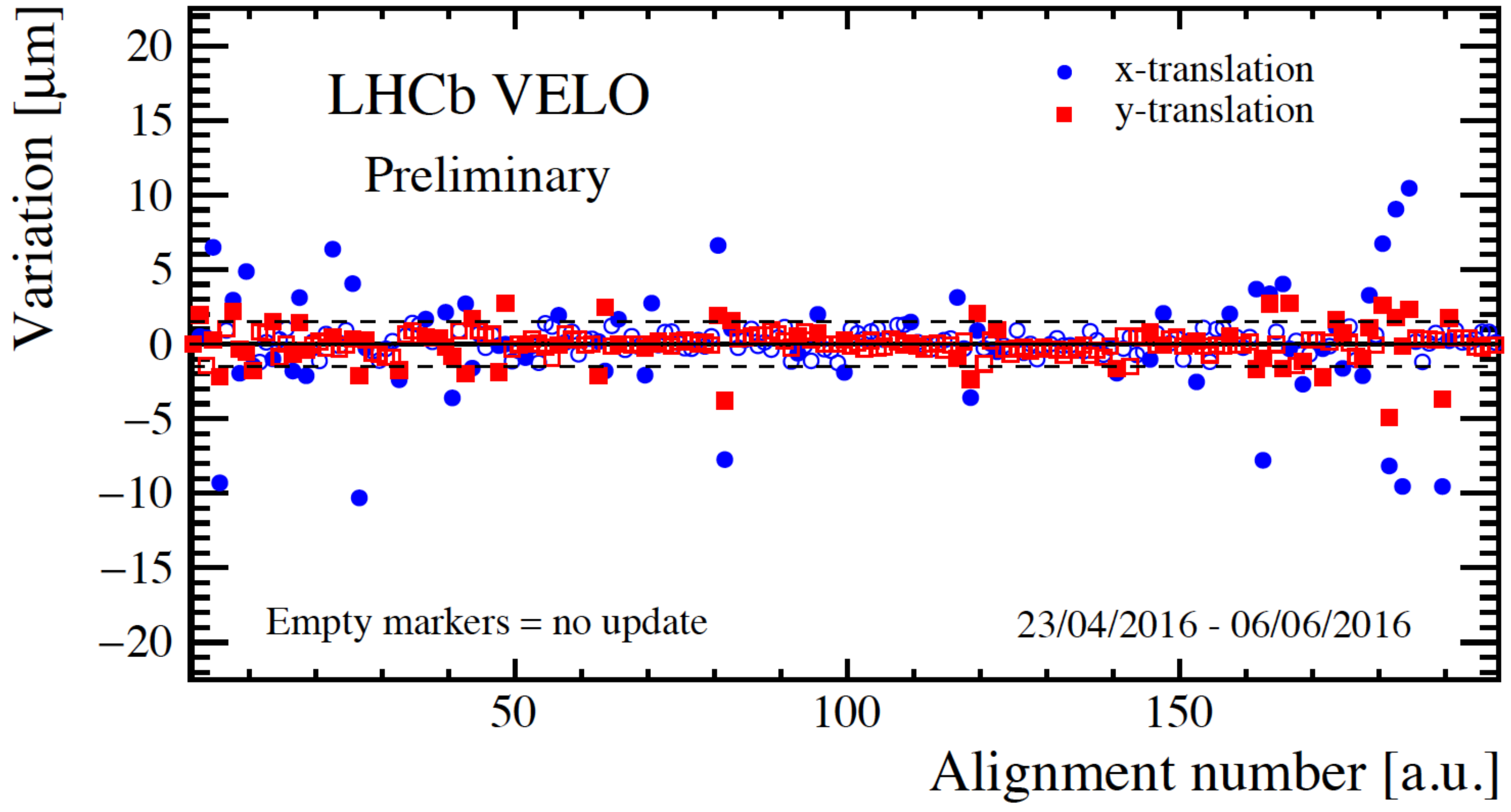
Interplay with detector calibration

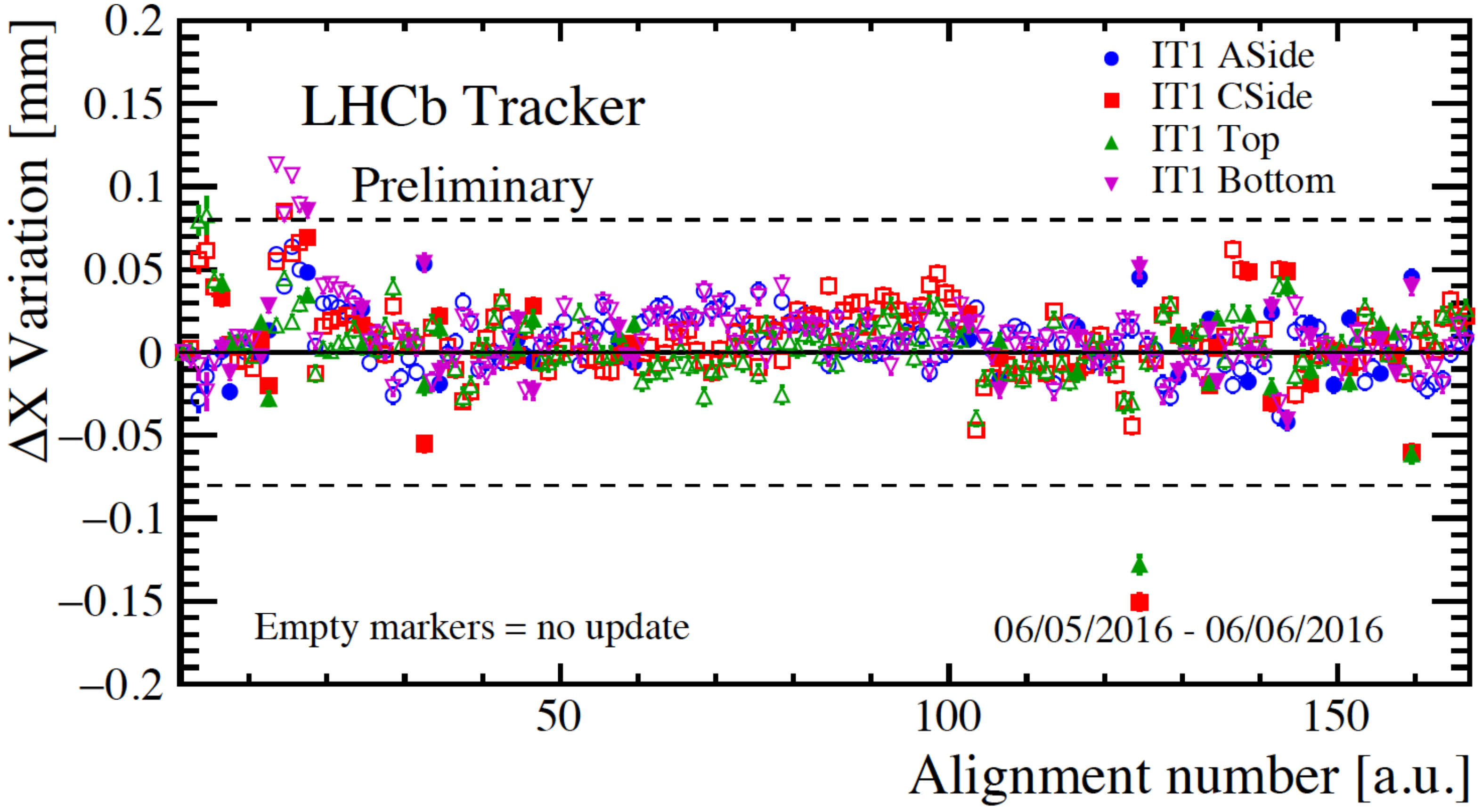
Real-time alignment and calibration is not a precondition for using MVA in the trigger, we introduced MVA in 2011 and we real-time alignment/calibration in 2015

However, it allows us to worry even less about stability issues. Full offline reconstruction in trigger allows MVA to carry the full selection burden, increasing efficiencies.

Real time alignment and calibration

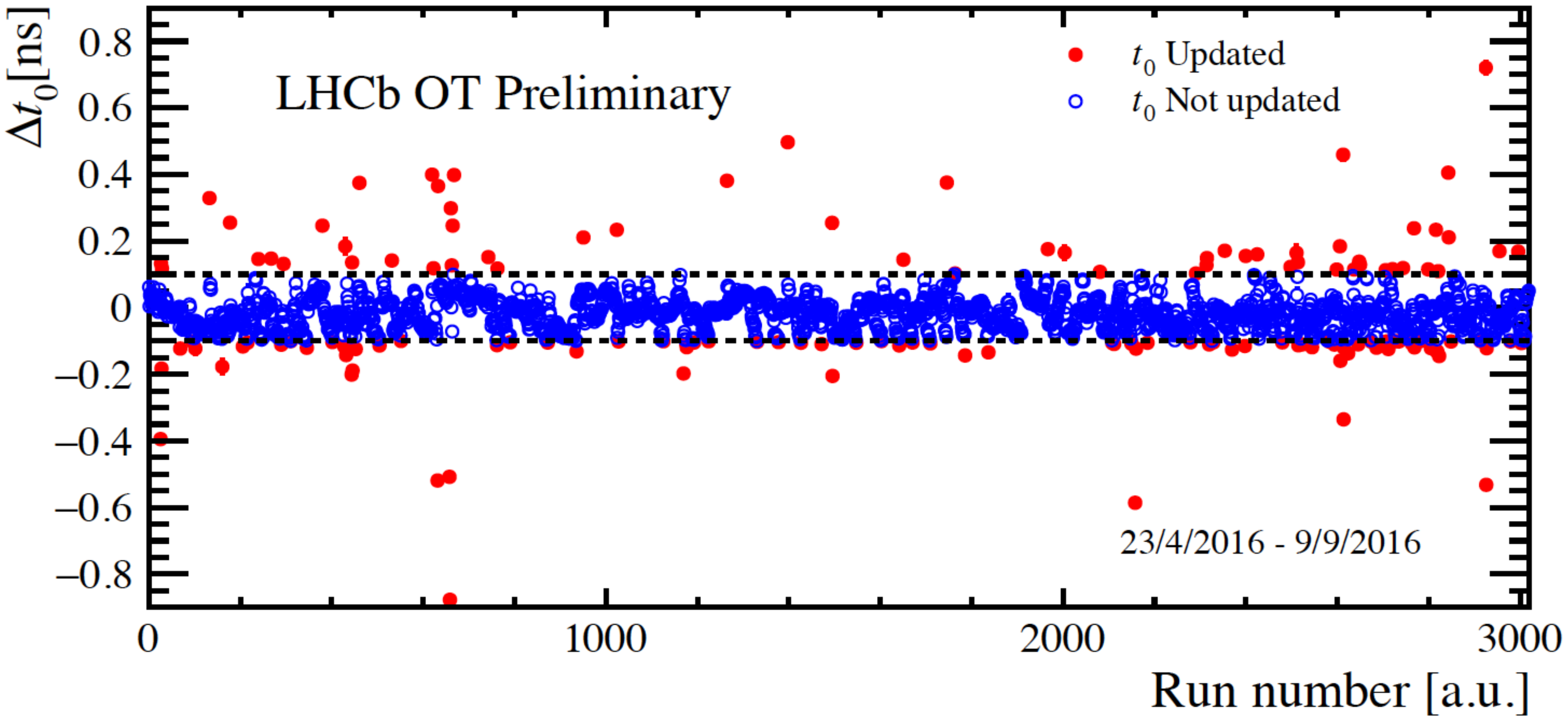
- update alignment constants only when above threshold (dashed lines)
  - VELO opens and closes each fill (protect sensors during injection): expect updates every few fills
  - tracking system (TT, IT, OT): expect updates every few weeks





Real time alignment and calibration

- update alignment constants only when above threshold (dashed lines)
  - VELO opens and closes each fill (protect sensors during injection): expect updates every few fills
  - tracking system (TT, IT, OT): expect updates every few weeks

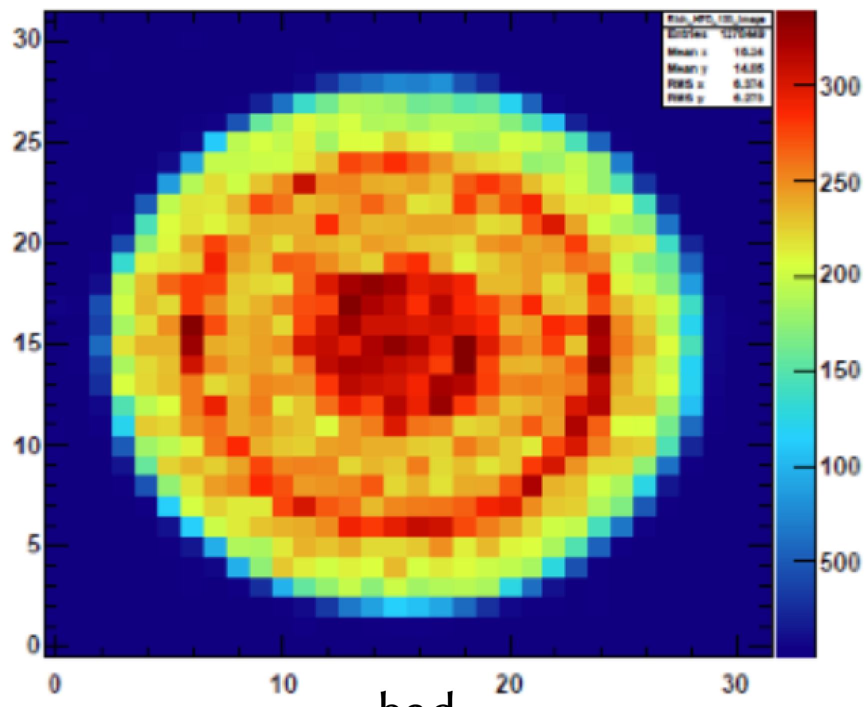


Real time alignment and calibration

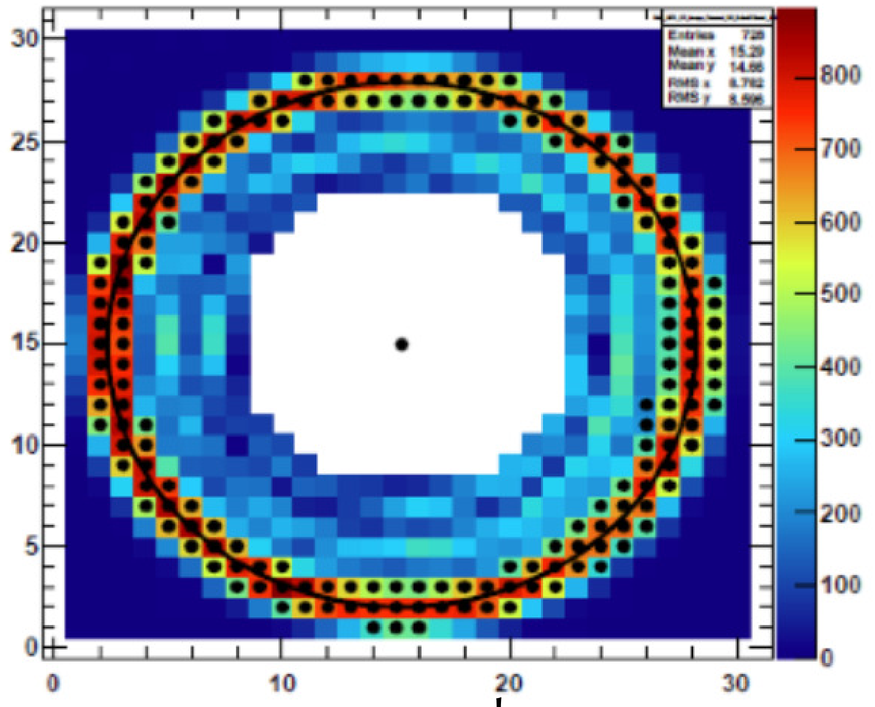
For optimal physics must calibrate and align the gaseous Ring-Imaging Cherenkov Detectors in real time.

Monitor and adjust the mirror alignment, the image distortion, and the refractive index of the gas.

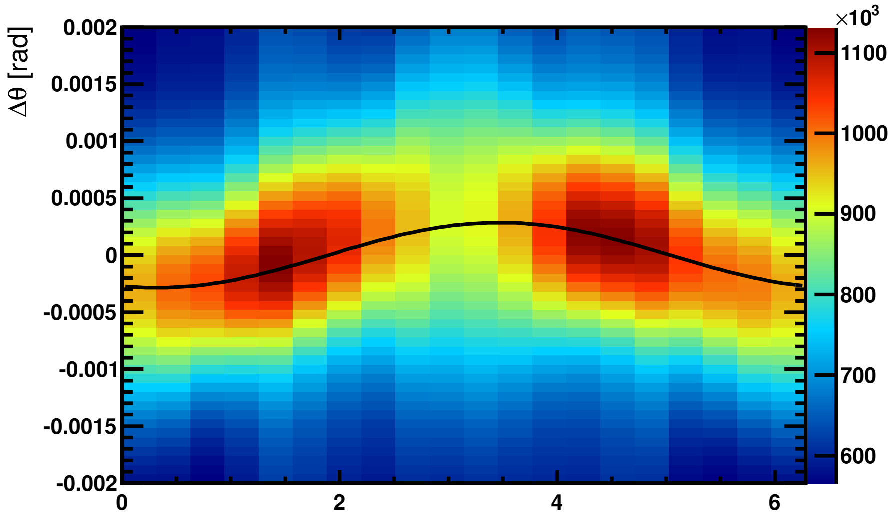
System almost entirely automated, can update image and refractive index parameters within less than a minute if needed. Alignment takes longer but also changes much less frequently (1-2 times per year).



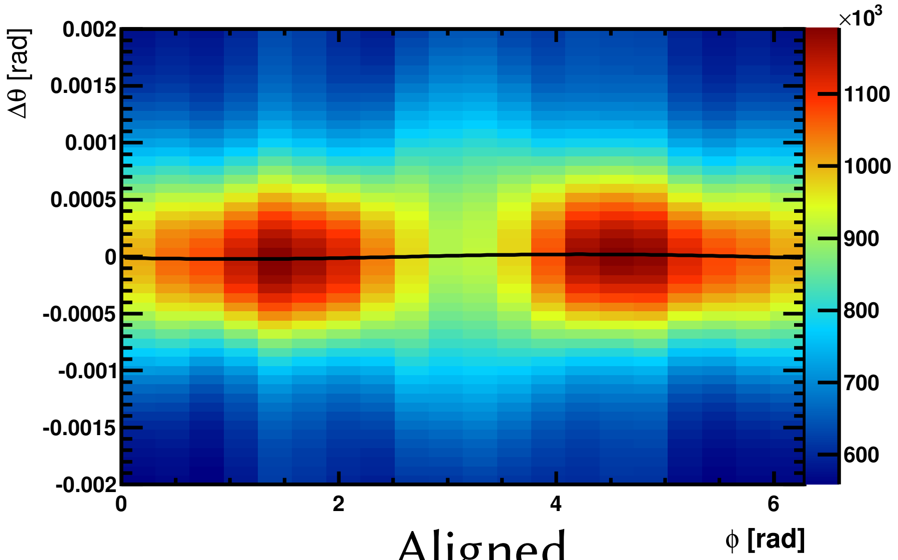
bad



good

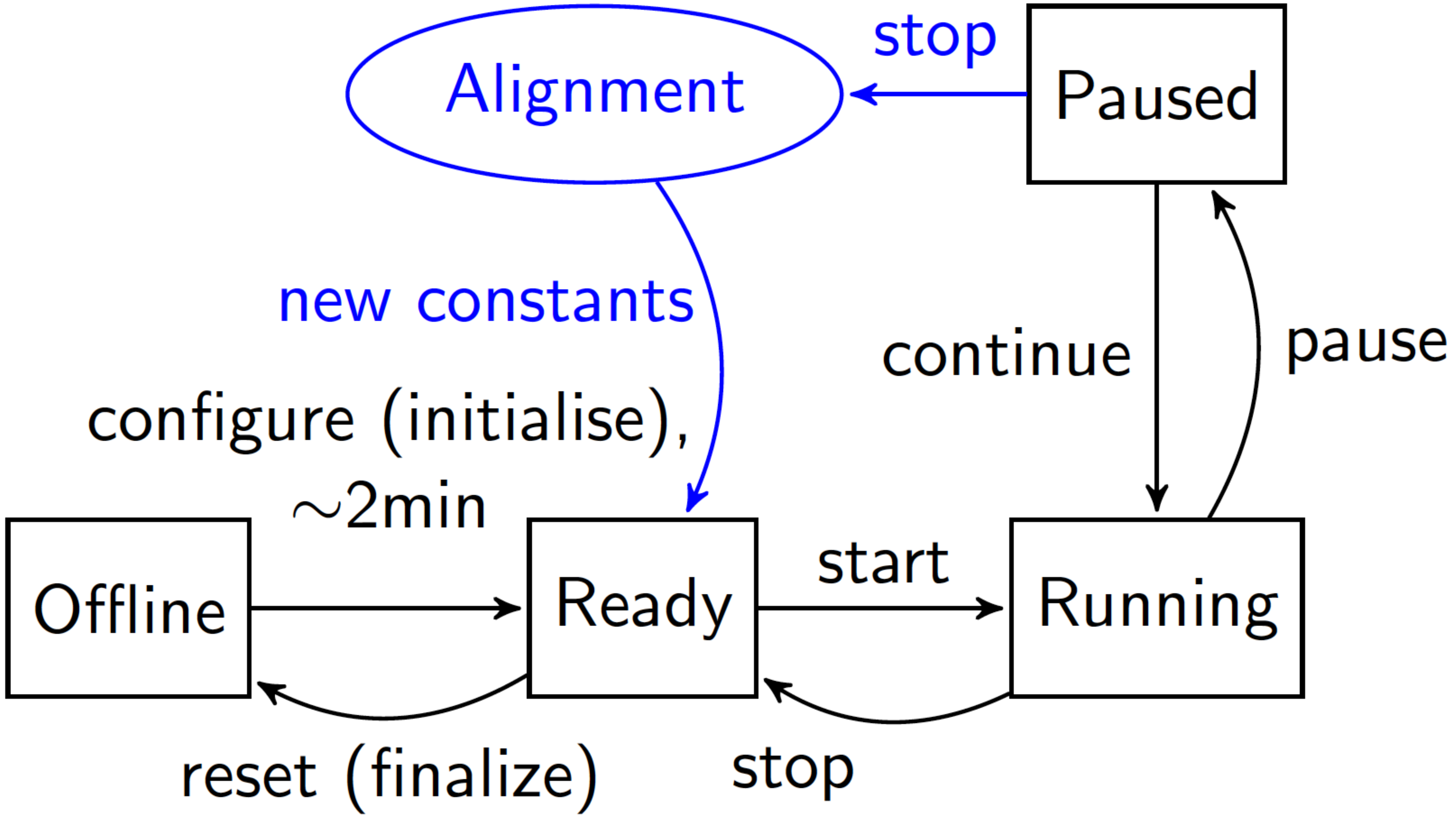


Misaligned



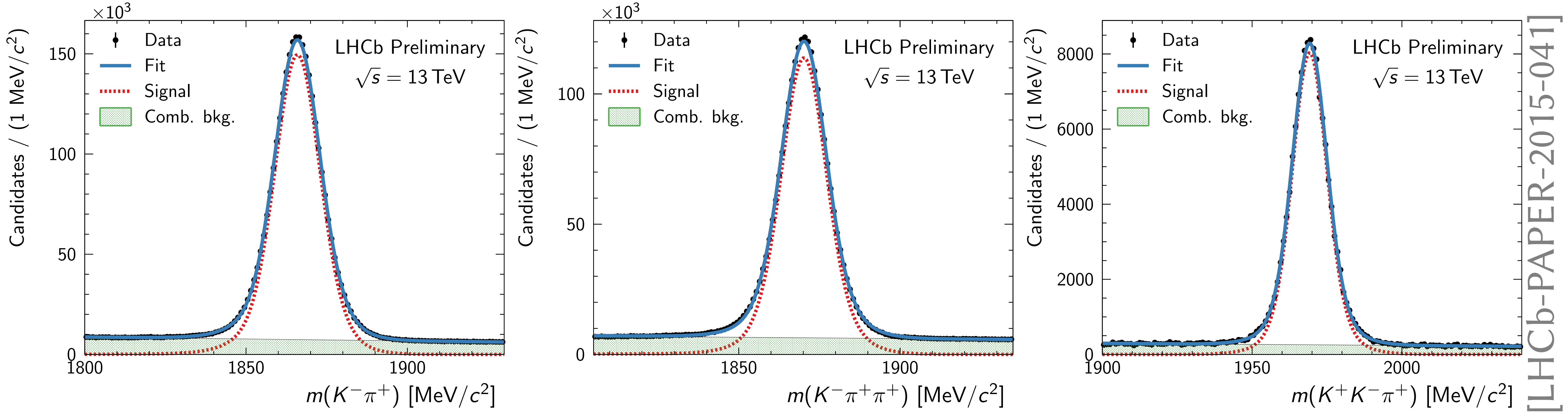
Aligned

Implementation details



Alignment tasks distributed across the 1700 HLT farm nodes, each processes a few events, a central analyzer task collects the results and performs the minimization. This is how we can align the detector automatically in the space of a few minutes (with negligible impact on HLT)

Real time signals in 2015



$D^0 \rightarrow K^- \pi^+$ $D^+ \rightarrow K^- \pi^+ \pi^+$ $D_s^+ \rightarrow K^+ K^- \pi^+$

Trigger level signal purities and resolutions for charged particles identical to the best possible offline ones. Published first papers 2 weeks after data taken!

Conclusion and Outlook

LHCb trigger uses machine learning throughout

We have developed ways to make machine learning techniques are safe for use in a real-time environment

We anticipate vast majority of trigger algorithms will be at least BDT based by Run III. Goal is to make it easy for Run III offline analysts to deploy whatever the most advanced machine learning methods of 2020 are.

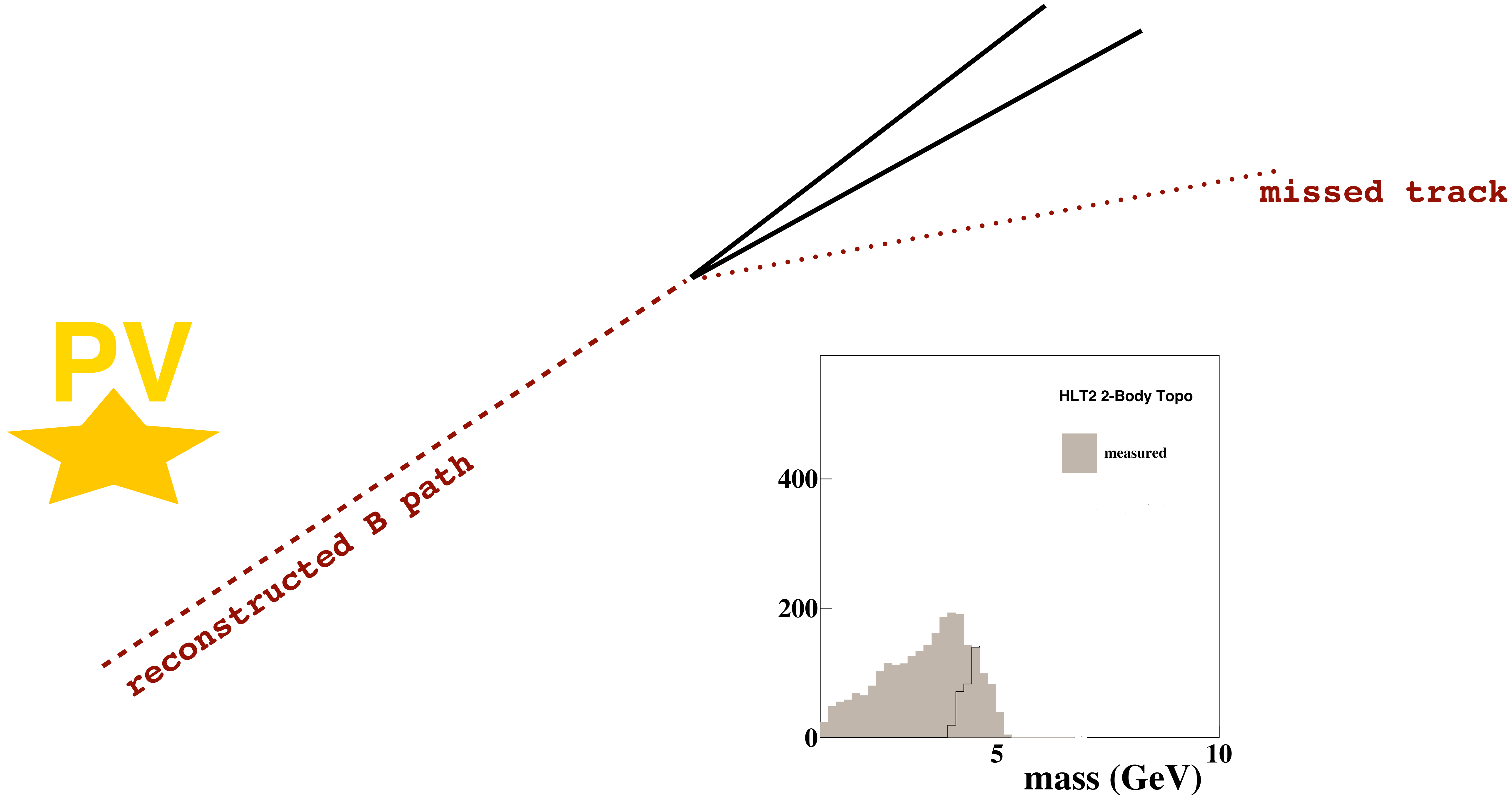
Any questions?


The Future Is Now!

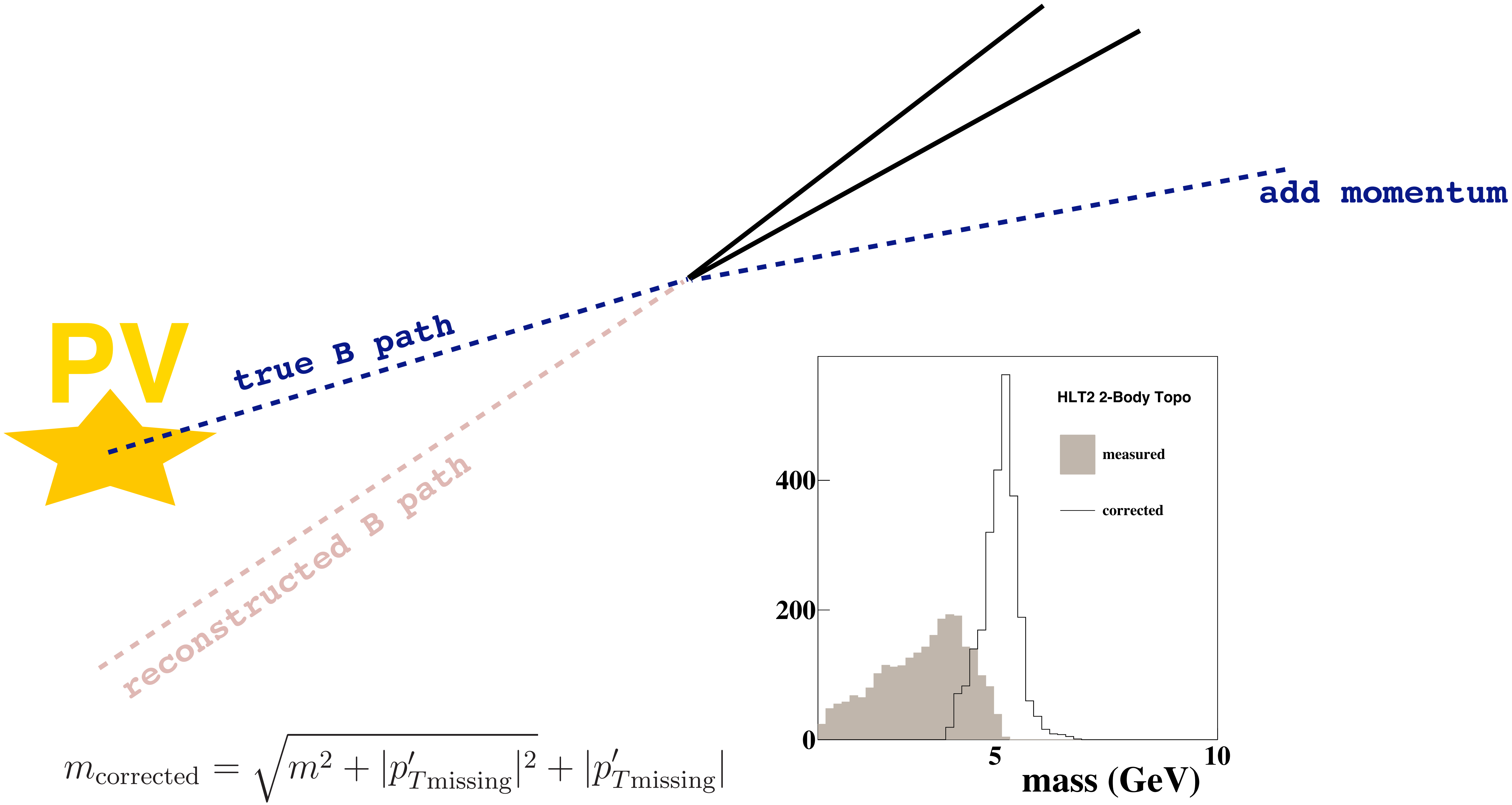
Topological variable binning in Run I

variable	cuts(2,3,4-body)	allowed splits
$\sum \|p_T\|$	$>$ 3,4,4 GeV	3.5,4,4.5,5,6,7,8,9,10,15,20 (GeV)
mass	$<$ 7 GeV	2.5,4.75 (GeV)
DOCA	$<$ 0.2 mm	0.05,0.1,0.15 (mm)
IPχ^2		20
corrected mass		2,3,4,5,6,7,8,9,10,15 (GeV)
$p_T^{\min}$	$>$ 0.5 GeV	0.6.0.7.0.8,0.9,1,1.25,1.5,1.75,2,2.5,3,4,5,10 (GeV)
FDχ^2	$>$ 100	2,3,4,5,6,7,8,9,10,25,50,100 $\times$100

Example : inclusive beauty trigger

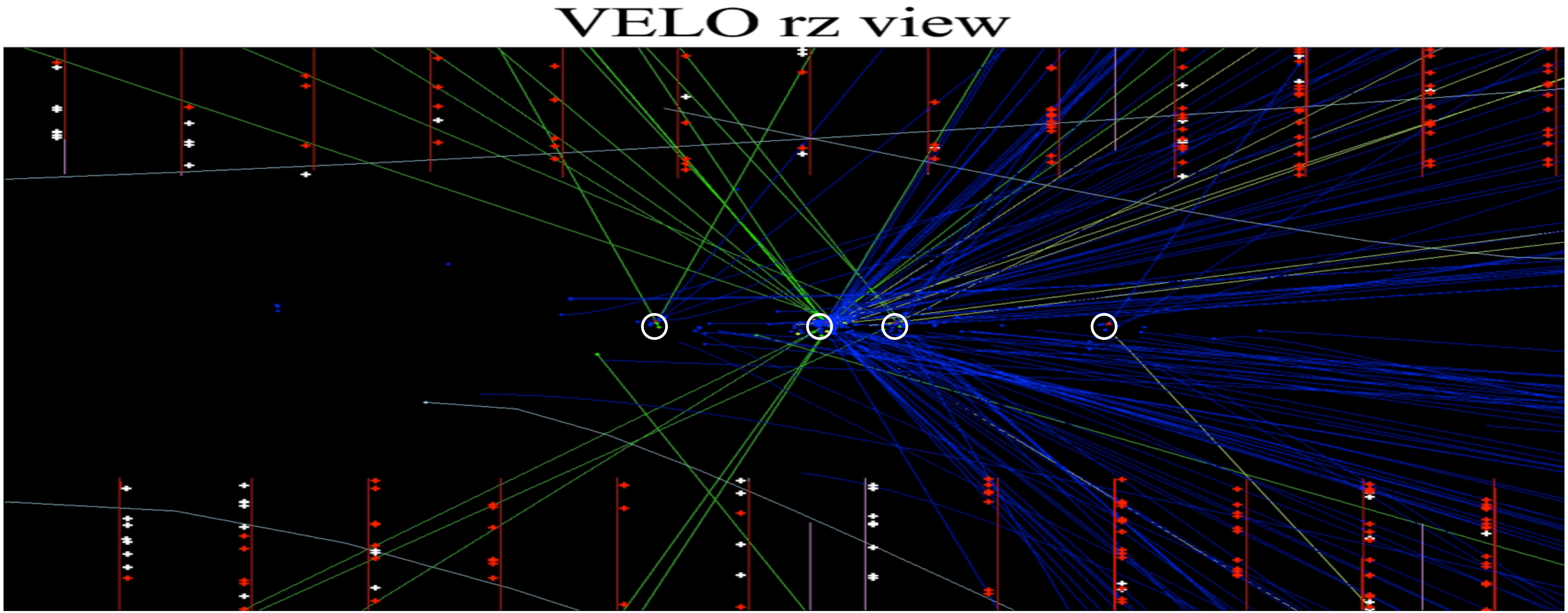


Example : inclusive beauty trigger



$$m_{\text{corrected}} = \sqrt{m^2 + |p'_{T\text{missing}}|^2} + |p'_{T\text{missing}}|$$

LHCb environment



In 2010-2012 : 15 MHz of bunch crossings, ~1.5 proton-proton interactions per bunch crossing, ~30 particles produced in the detector acceptance per interaction

In 2015, run with ~1.1 pp interactions per bunch crossing, similar multiplicity because of the increased collision energy.