Machine Learning in the LHCb Trigger and Beyond

Mike Williams

Department of Physics & Laboratory for Nuclear Science
Massachusetts Institute of Technology

July 19, 2017


The Large Hadron Collider
LHCb
70 institutes
16 countries
700 physicists
Almost 400 papers!

The Short-Short Version

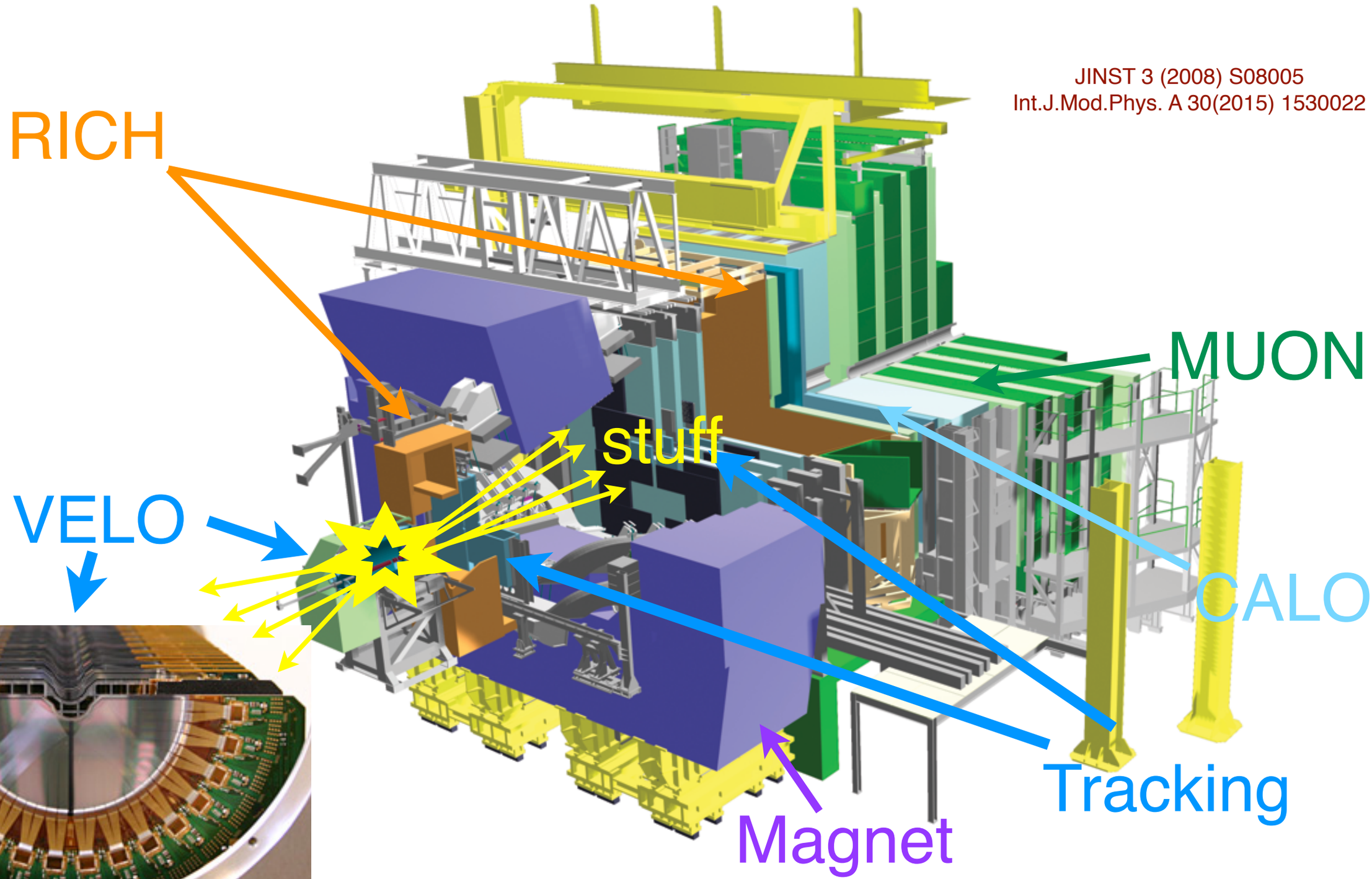
We use ML almost everywhere, and we've moved to a real-time calibration system putting much "analysis" online—to enable great science!

LHCb Detector

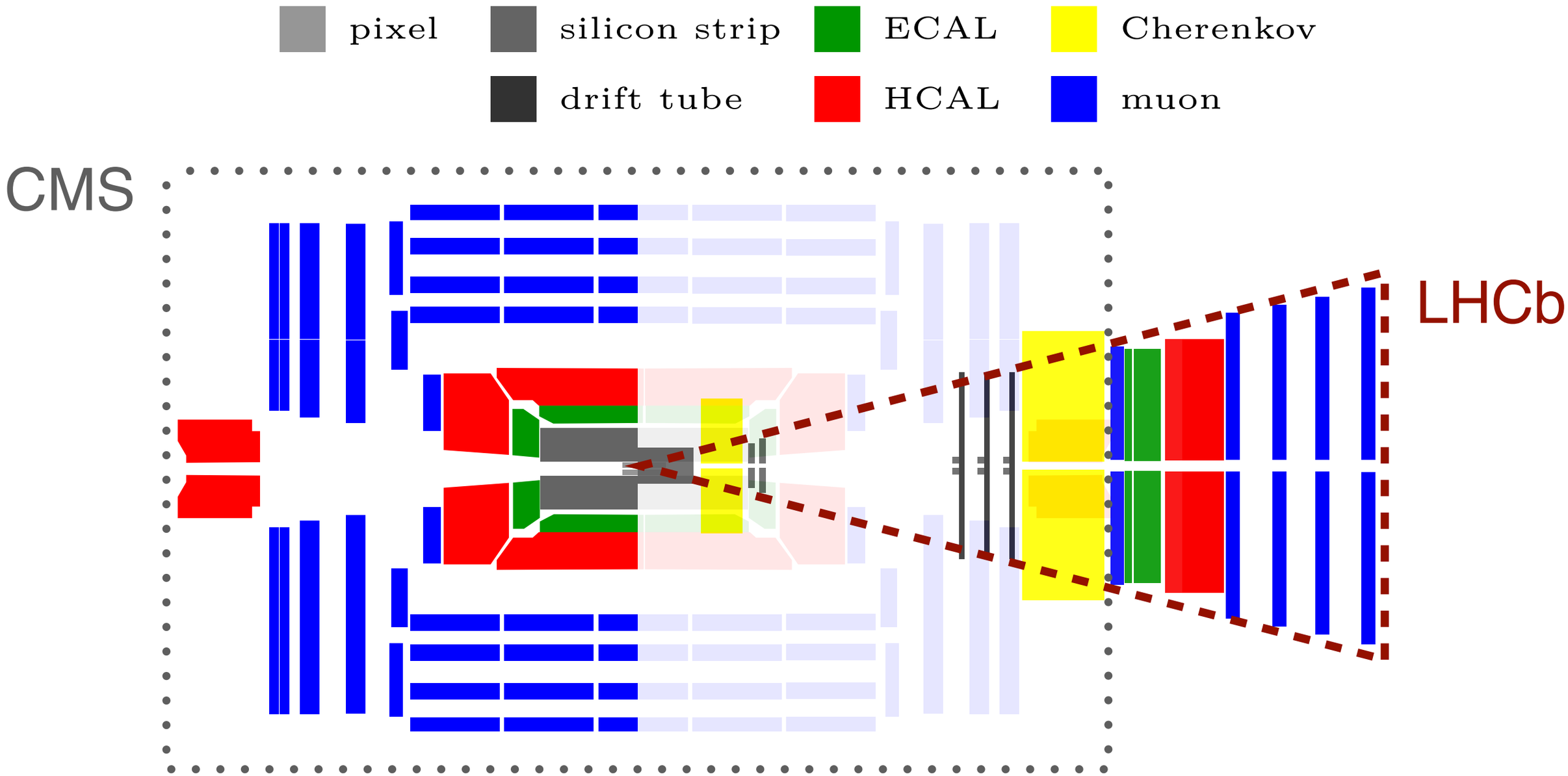
LHCb is a forward Spectrometer ($2 < \eta < 5$)

(roughly 1-15^o)

JINST 3 (2008) S08005
Int.J.Mod.Phys. A 30(2015) 1530022



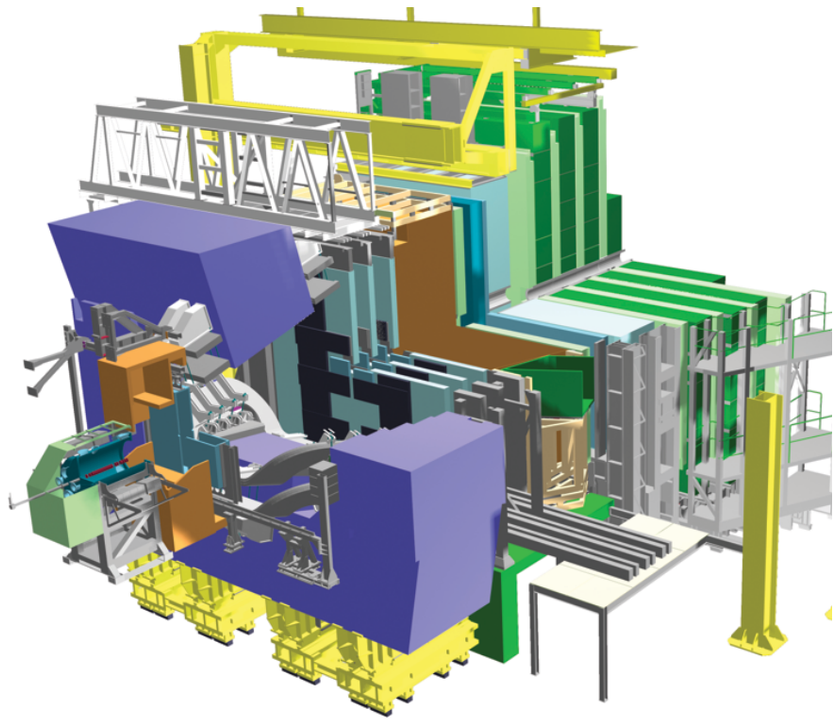
LHCb Detector



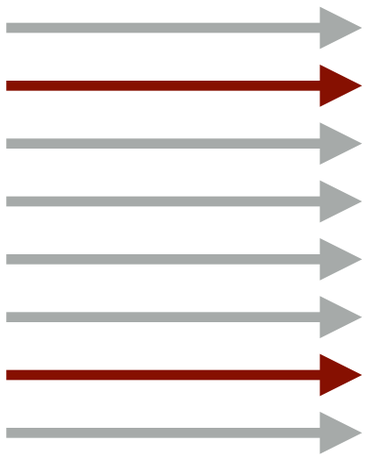
Complimentary kinematical coverage to CMS & ATLAS.

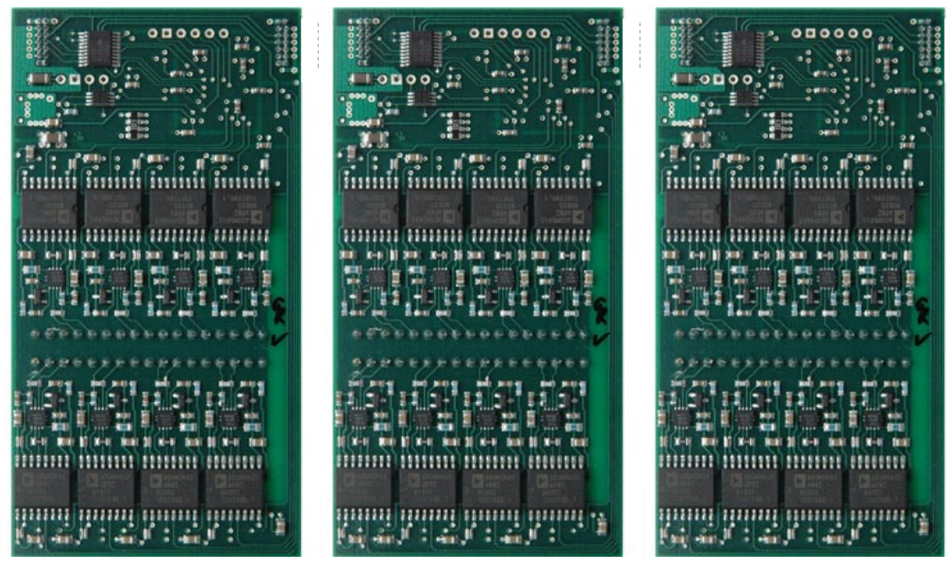
Big Data & Triggers

A streaming readout of all channels would be 100 TB/s, so perform ~lossless compression by using zero-suppression on the frontend electronics.



100 TB/s 1 TB/s



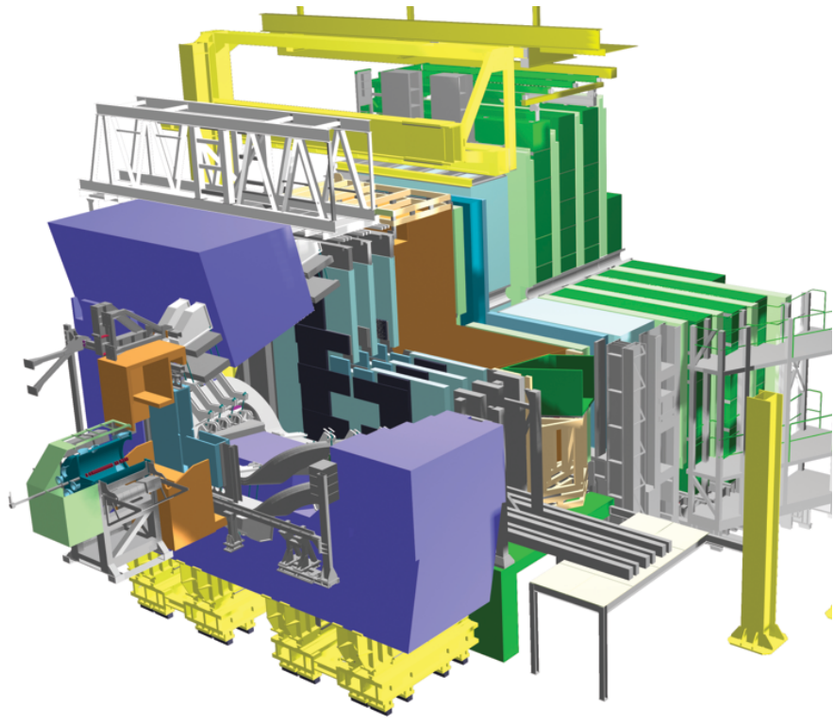


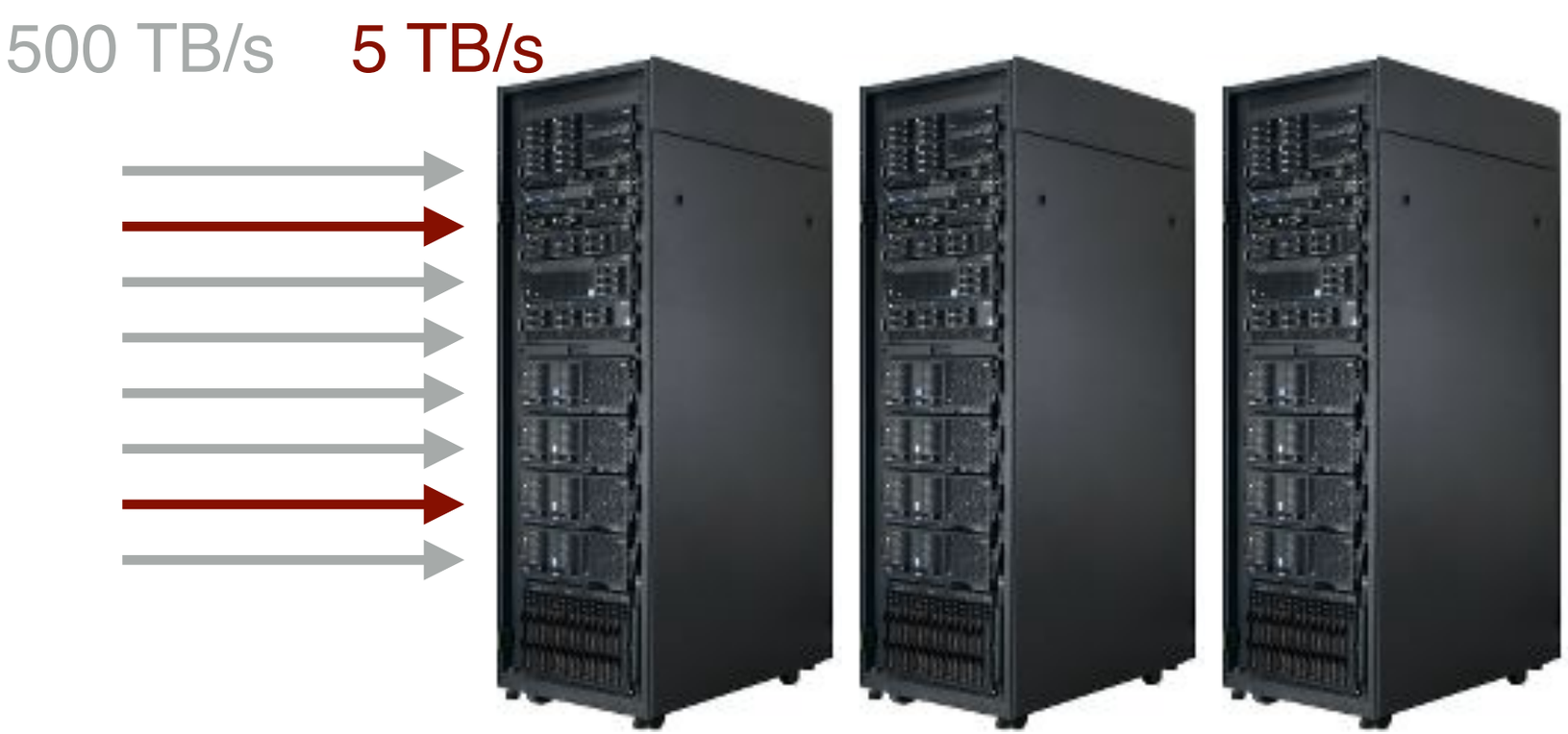
Current electronics only permit reading out 50 GB/s, data reduced first using hardware (e.g. FPGA) with fixed latency.

That still leaves 250 PB per year! Need to reduce further (use CPUs), but what data should we keep?

Big Data & Triggers

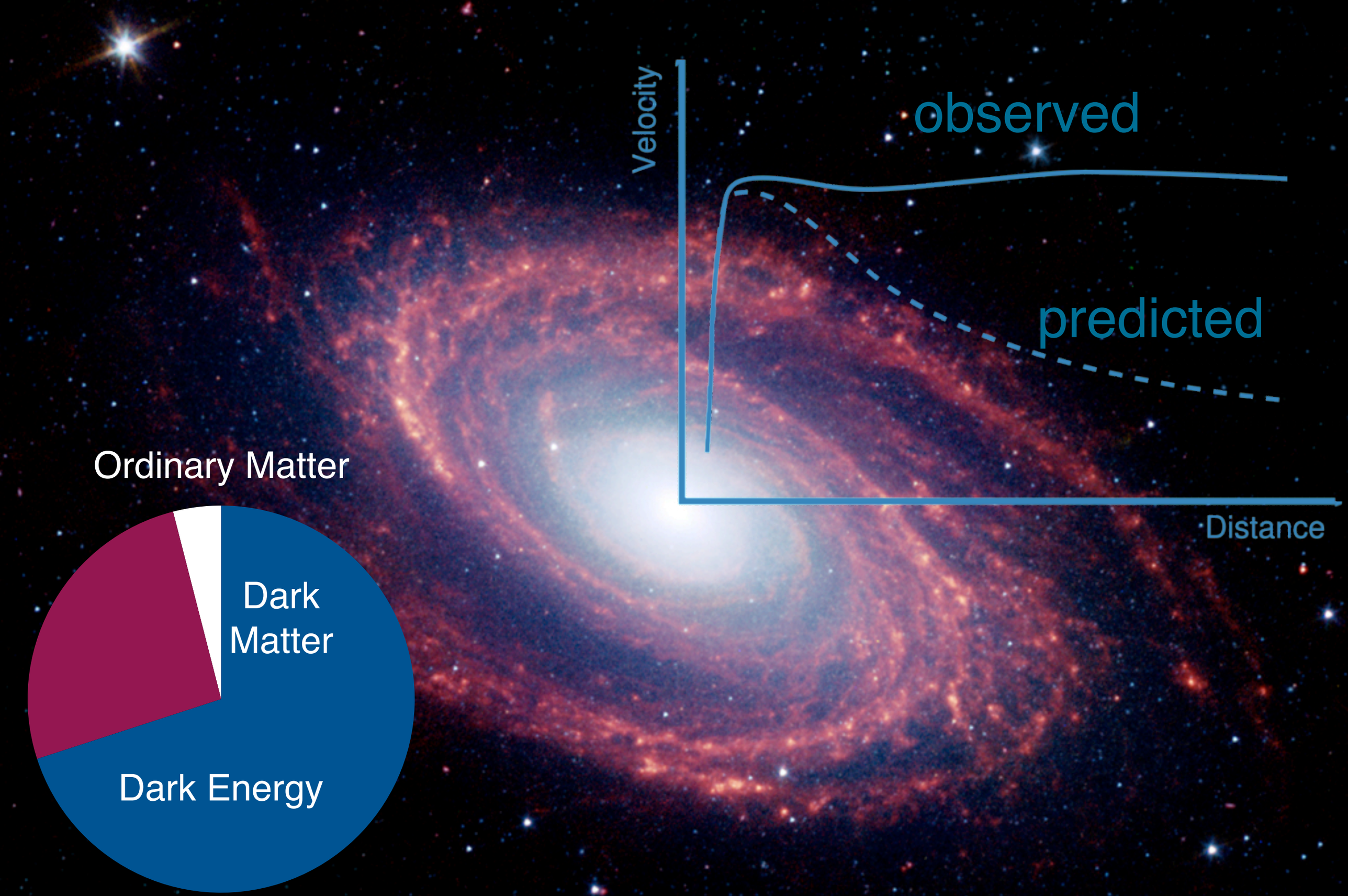
A streaming readout of all channels would be 500 TB/s in Run 3, so perform ~lossless compression using zero-suppression on the frontend electronics.





The electronics are being upgraded to permit reading out the full zero-suppressed data rate — and the luminosity is being increased by a factor of 5 in Run 3 (2021-2023). This will be 25 EB per year!

This data potentially contains some extremely interesting signals, and LHCb is the best-equipped detector ever built to study them. However, identifying events containing such signals in real time is an incredible challenge — an opportunity to enable great science using ML.

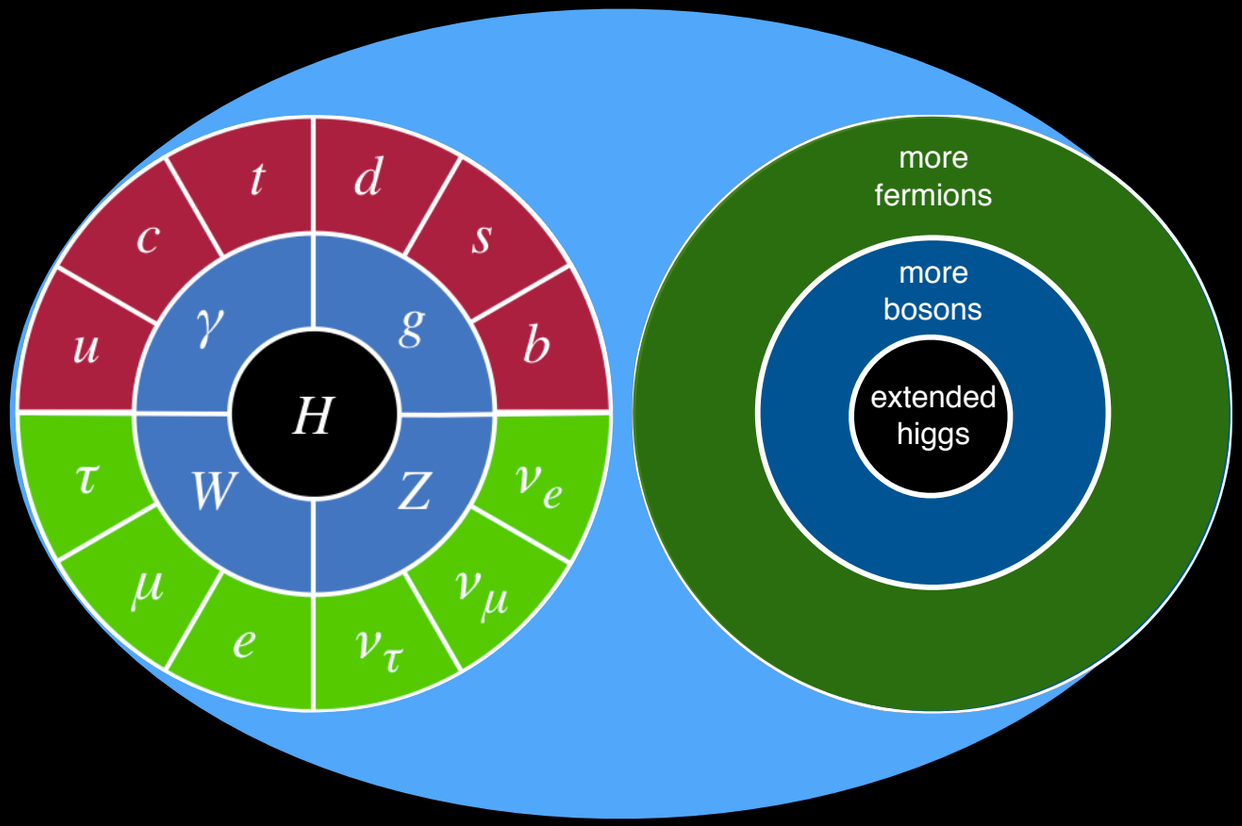


The SM is frustratingly successful at describing ordinary matter, but it provides no viable dark matter candidate. What is the microscopic nature of DM?

Dark Matter Paradigms

(apologies to axions)

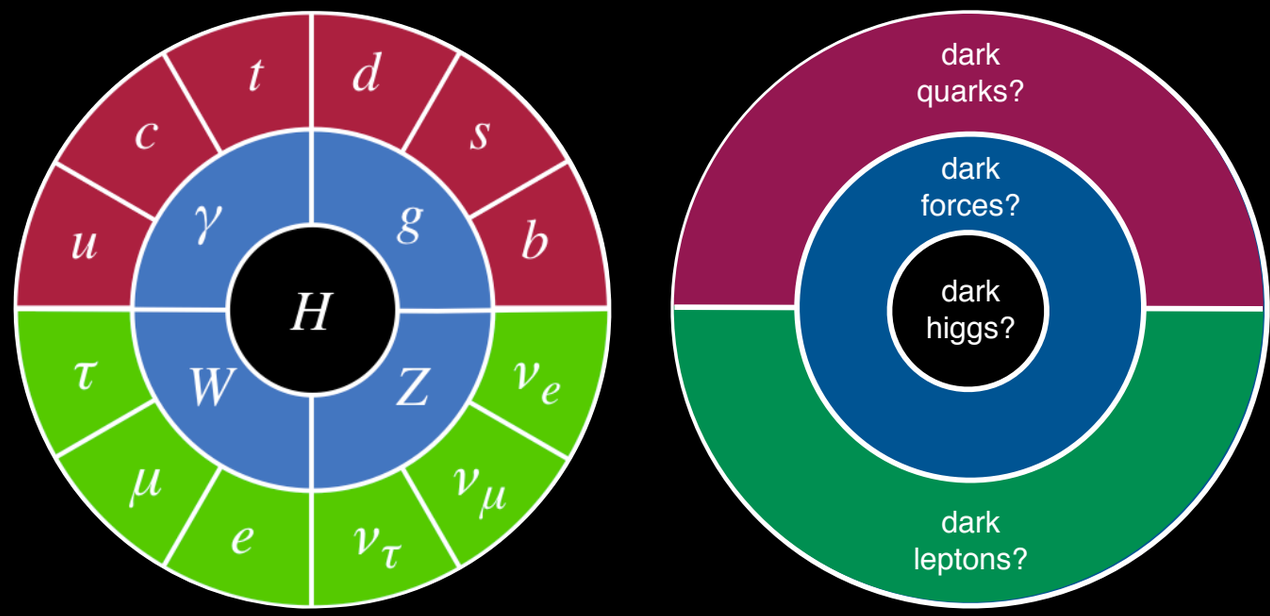
WIMP



SM and DM particles are part of a larger unified theory at the TeV scale.

LHCb searches for indirect evidence of this via quantum effects (flavor physics, aka core physics program).

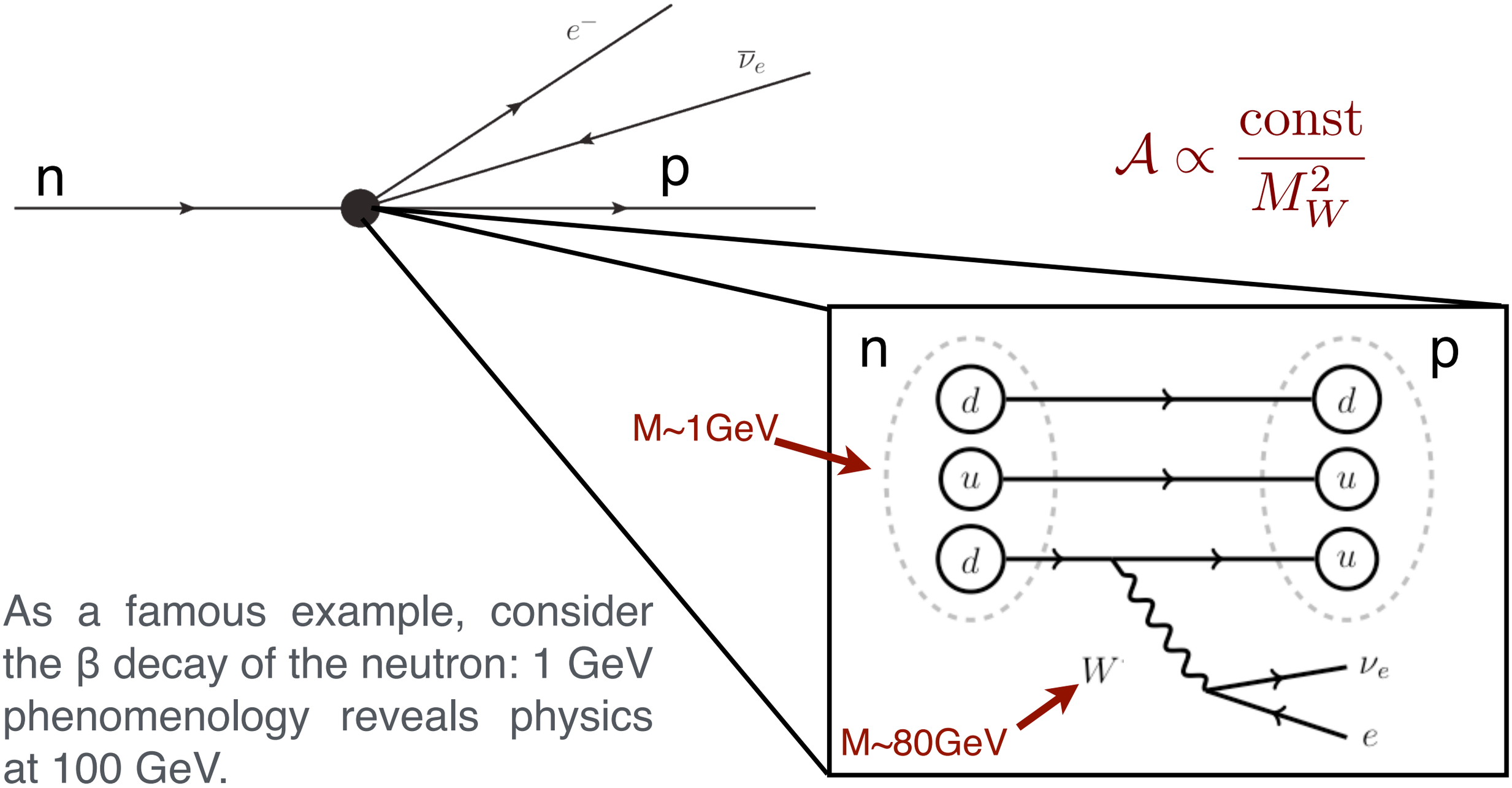
Hidden Sector(s)



No direct SM-DM connection. LHCb searches for this directly, and has (or will have) world-leading sensitivity in certain regimes.

Indirect Observation

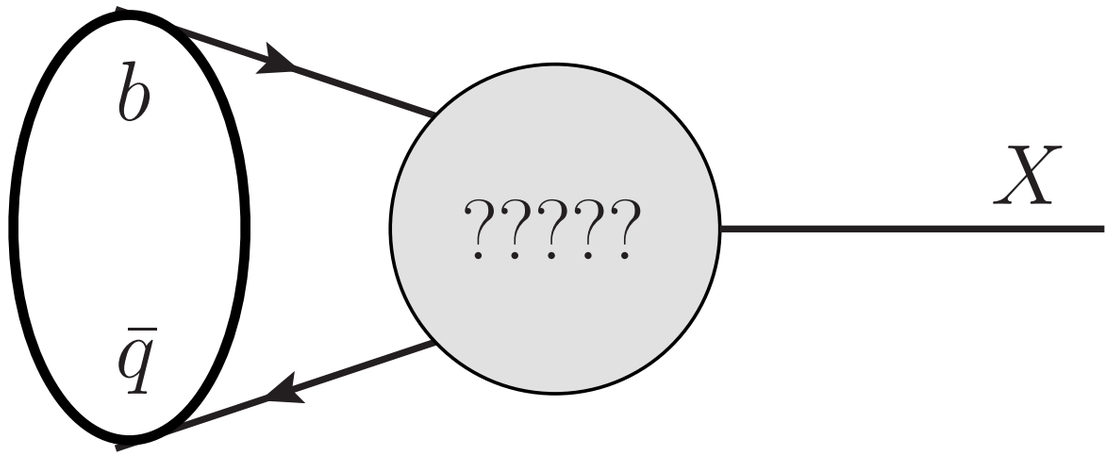
Indirect observations have historically been used to infer the existence of new particles before experiments with sufficient energy to produce them have existed.



As a famous example, consider the β decay of the neutron: 1 GeV phenomenology reveals physics at 100 GeV.

Probing High Mass Scale

The most sensitivity to high mass scales occurs in processes where the SM amplitude is small and precisely calculable. Also seek to probe possibilities not well covered by previous experiments.



$$\mathcal{A} = \mathcal{A}_{\rm SM} + \mathcal{A}_{\rm BSM}$$

operator product expansion

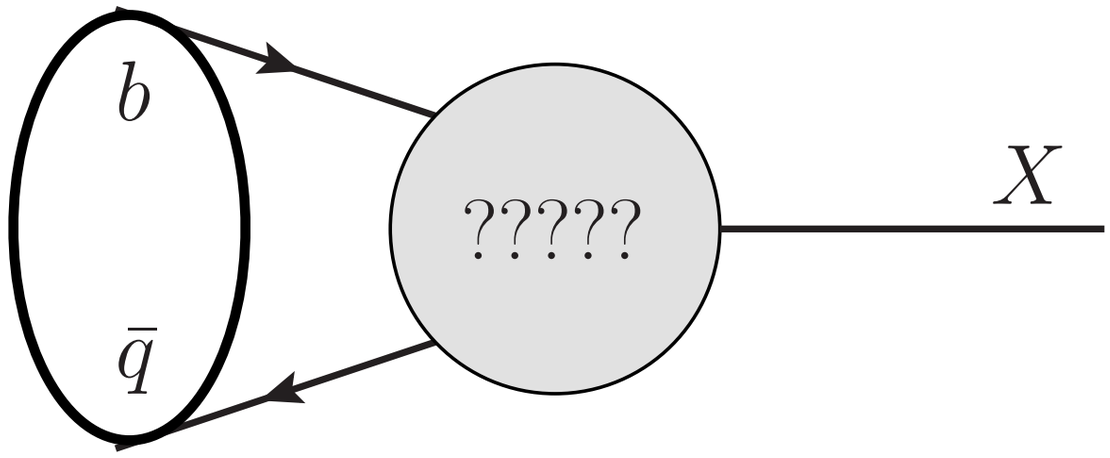
$$\mathcal{H}_{eff} \propto \sum_i \left(C^i_{SM} + C^i_{BSM}\right) \mathcal{O}_i$$

Model-independent information on new physics can be obtained from all quark-flavor-changing-current data, then interpreted within specific models.

N.b., in principle sensitive to any mass scale, limited only by experimental and theoretical precision (e.g. kaon oscillation data probes 10^5 TeV!).

Probing High Mass Scale

Primarily focus on QCD bound states containing c and b quarks, which have masses of 2-5m(proton) and lifetimes O(ps). At LHC energies, O(10%) of pp collisions produce charm and O(1%) produce beauty.



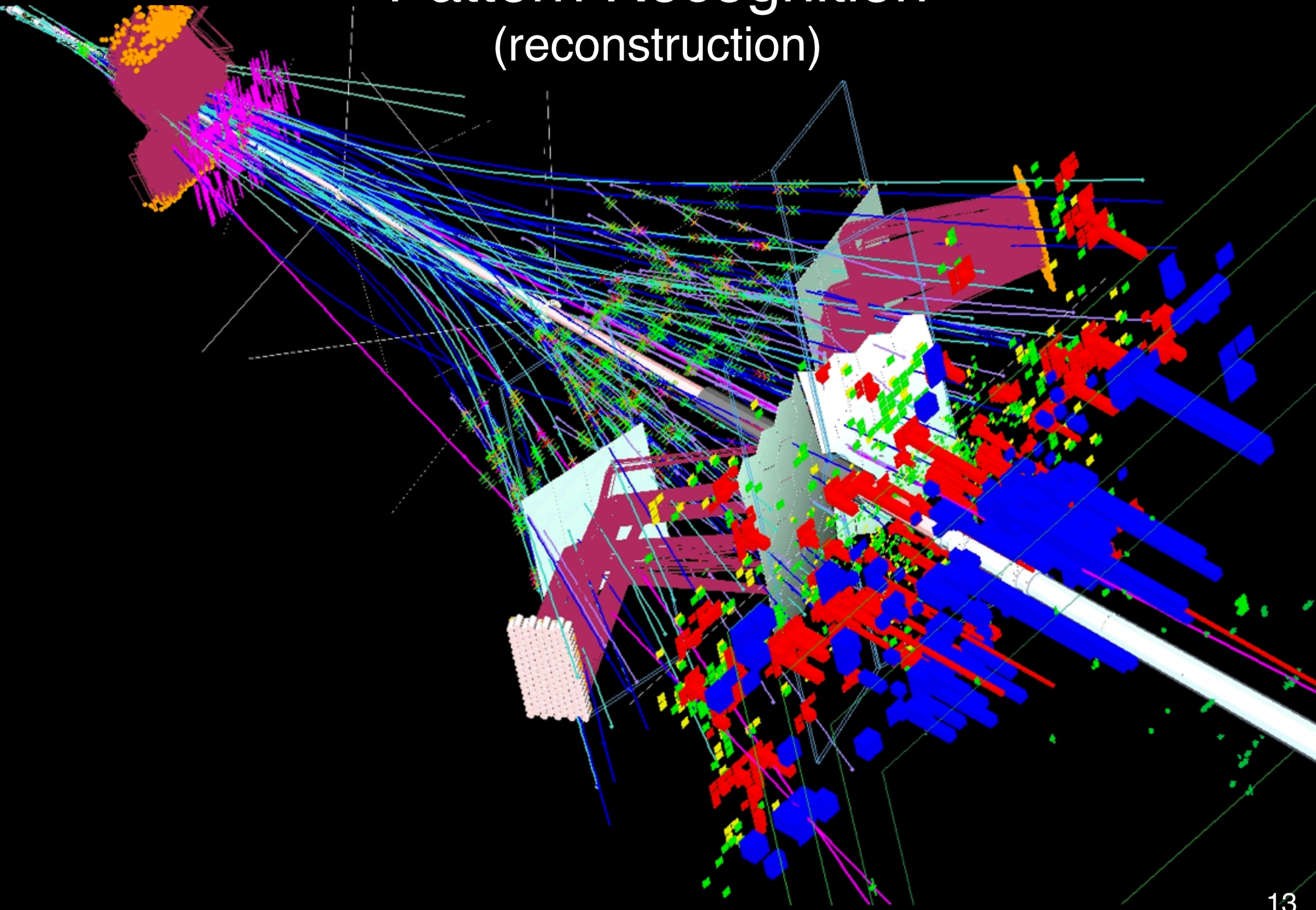
$$\mathcal{A} = \mathcal{A}_{\mathrm{SM}} + \mathcal{A}_{\mathrm{BSM}}$$

operator product expansion

$$\mathcal{H}_{eff} \propto \sum_i \left(\mathcal{C}^i_{SM} + \mathcal{C}^i_{BSM}\right) \mathcal{O}_i$$

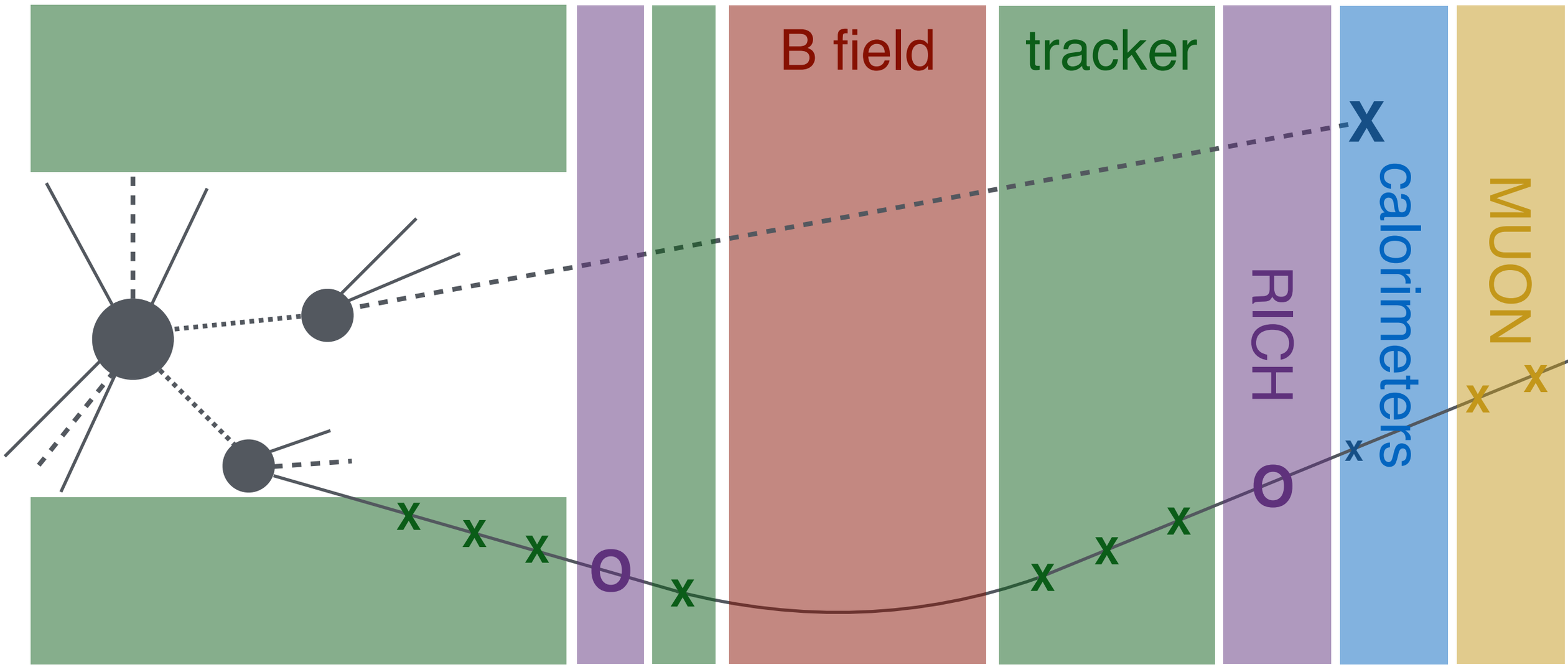
Can't even afford to keep all such decays (too much data). Furthermore, due to the relatively low masses and short lifetimes, the signatures of these events are not as striking as, e.g., a Higgs or SUSY ones (traditional trigger paradigm doesn't apply).

Pattern Recognition
(reconstruction)



Signatures

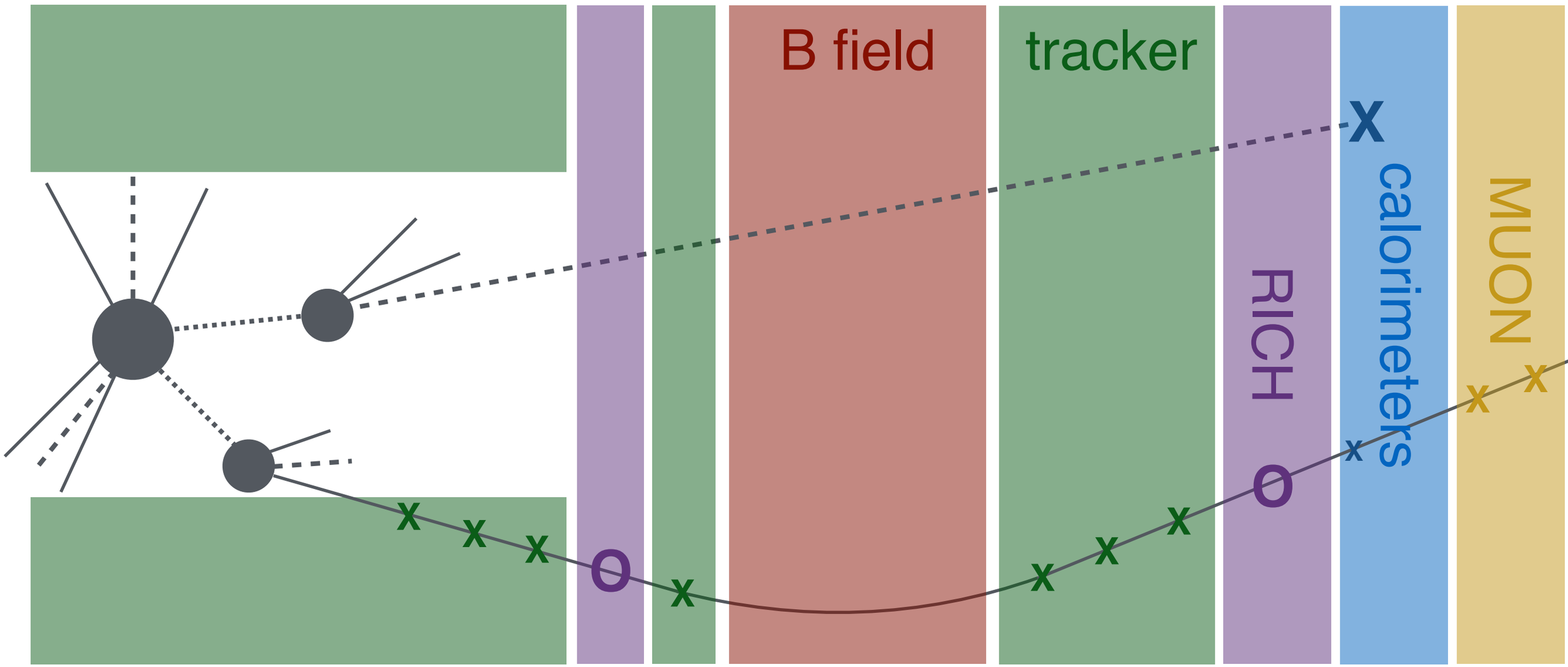
Not all charged particles produce hits in all systems, and not all information takes equal time to obtain. Have to decide quickly which events to keep.



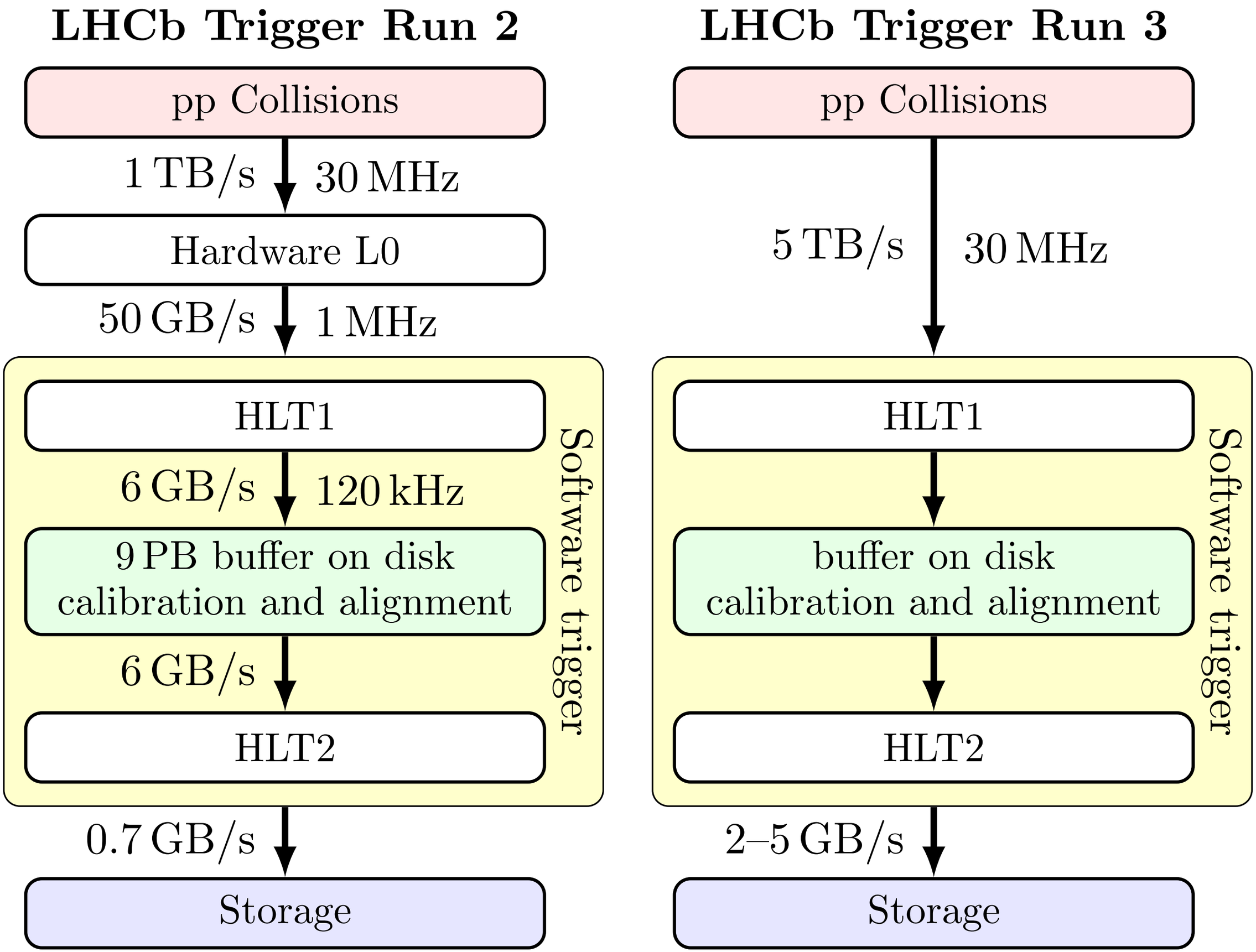
Signatures are tracks with large “impact parameter” (IP), especially leptons, SVs with certain masses, etc. All require substantial reconstruction to identify.

Hardware Trigger

Current detector must reduce the data rate by a factor of 40 before the detector can be read out. Only option is to use p_T thresholds in the MUON, ECAL, and HCAL systems.

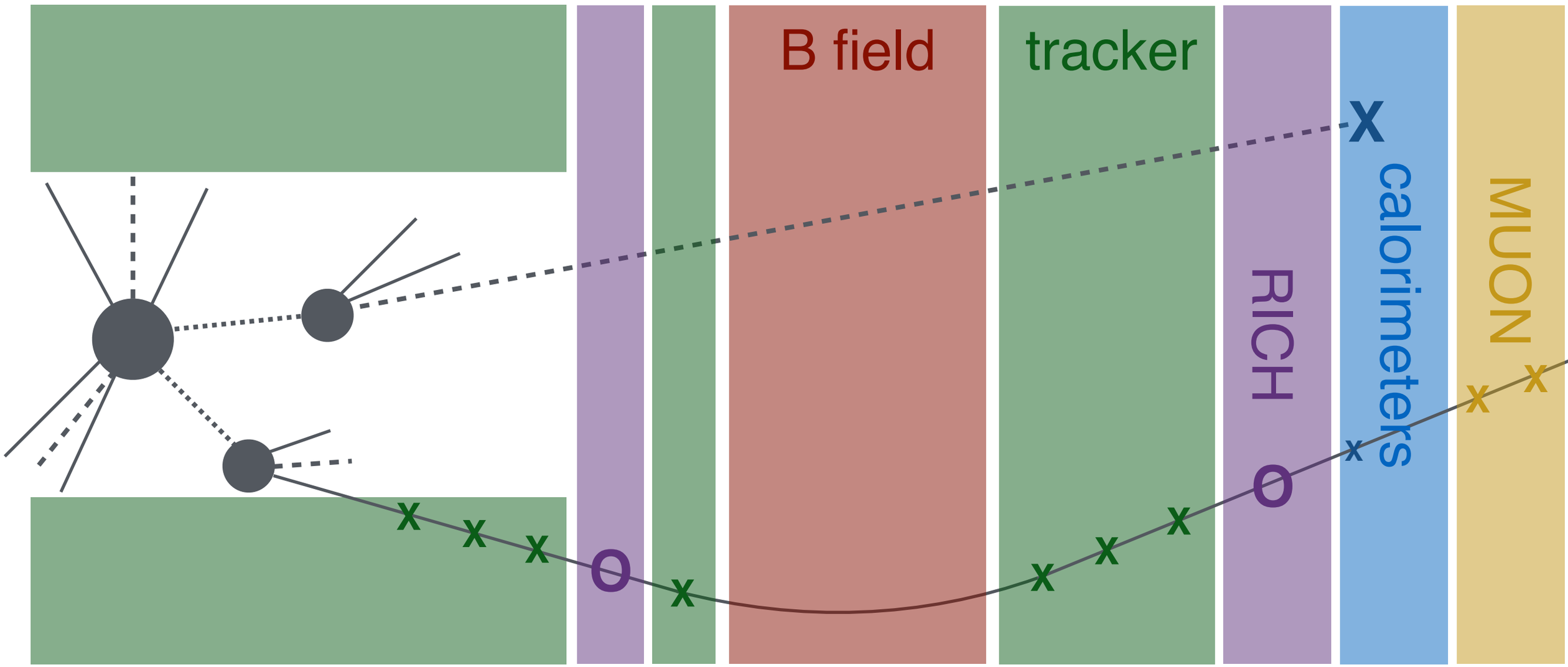


Removal of the hardware trigger stage in Run 3 will have a huge impact on the physics potential of LHCb.


LHCb Trigger Run 2
pp Collisions
1 TB/s
30 MHz
Hardware L0
50 GB/s
1 MHz
HLT1
6 GB/s
120 kHz
9 PB buffer on disk
calibration and alignment
6 GB/s
HLT2
Software trigger
0.7 GB/s
Storage
LHCb Trigger Run 3
pp Collisions
5 TB/s
30 MHz
HLT1
buffer on disk
calibration and alignment
HLT2
Software trigger
2–5 GB/s
Storage

HLT1

HLT1 has 25k physical cores (>50k logical cores) and access to all raw data, but cannot afford to do full event reconstruction. Choose to do charged-particle tracking with a threshold of $p_T > 0.5$ GeV (included PV making).

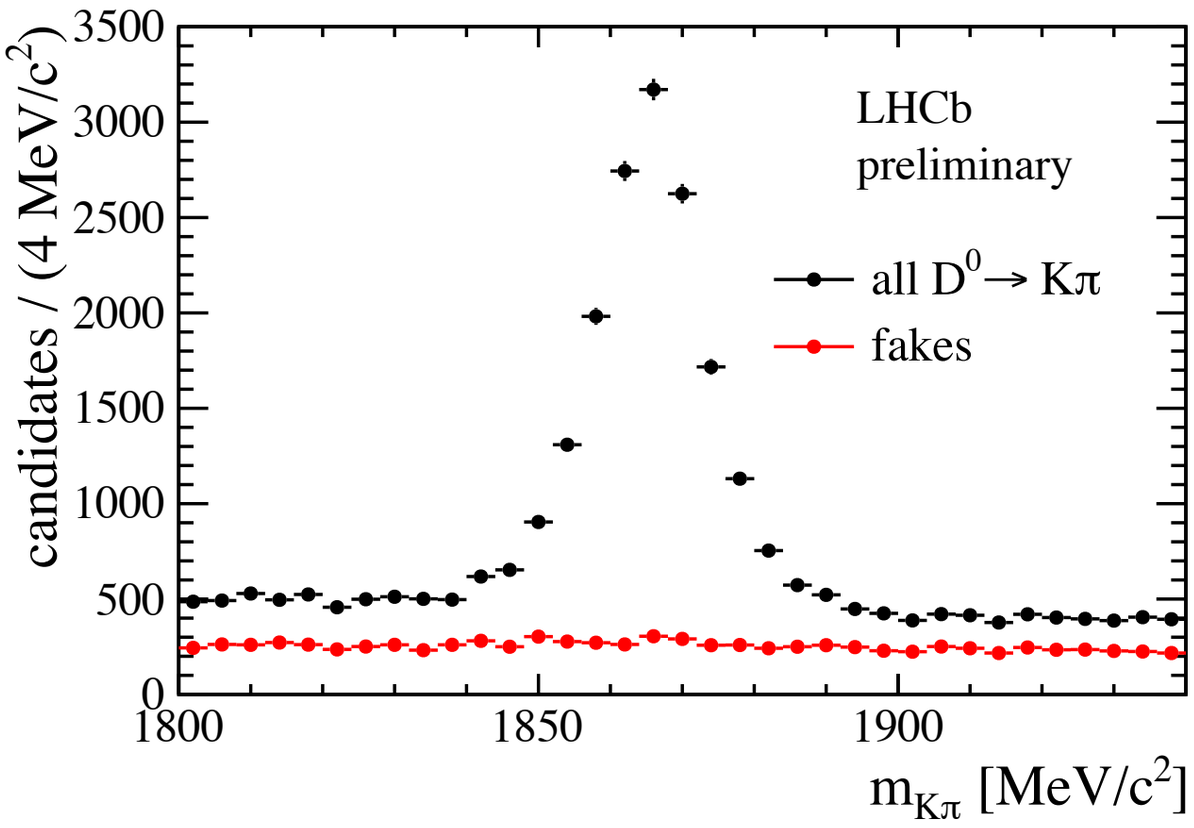


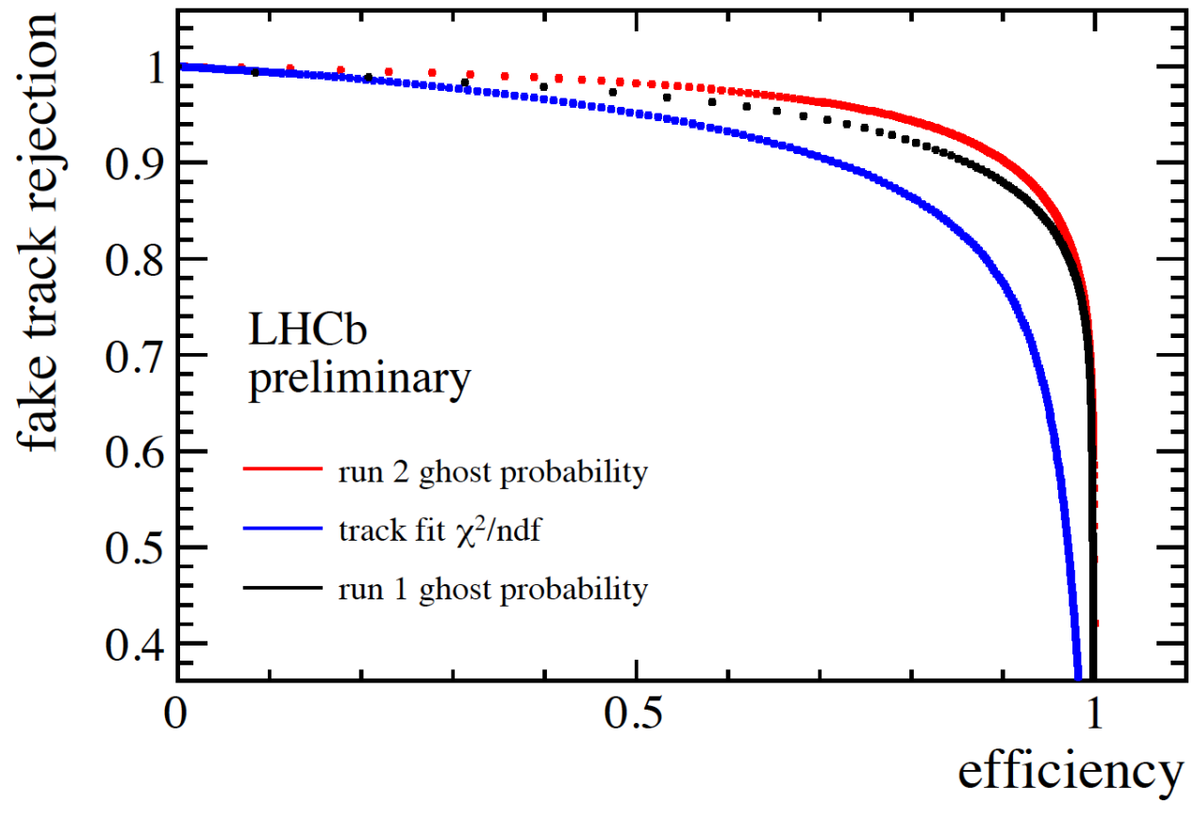
LHCb builds VELO segments first, then extends these to the next station, then beyond the B field to the final station before Kalman filtering all tracks.

Fake-Track Killer

Fake-track-killing neural network, most important features are hit multiplicities and track-segment chi2 values from tracking subsystems.

LHCb-PUB-2017-011

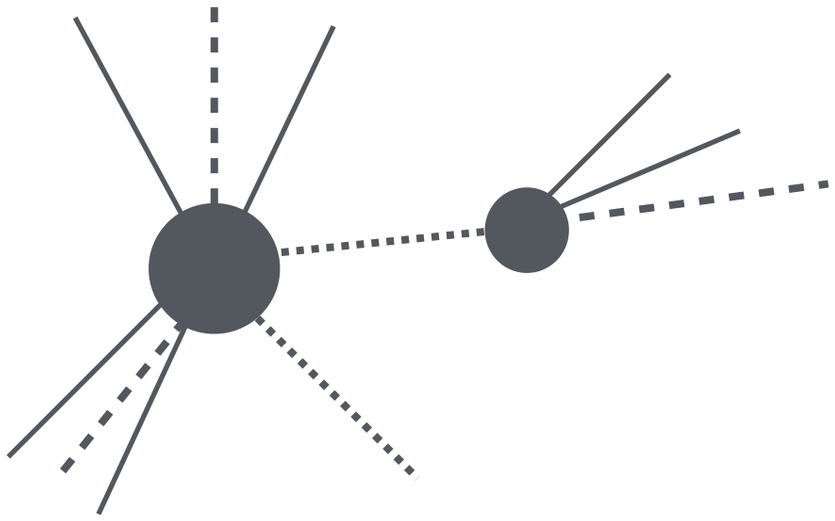




Run in the trigger on all tracks, so must be super fast. Use of custom activation function and highly-optimized C++ implementation (ROOT's TMVA package provides stand-alone C++ code to run the trained algorithm).

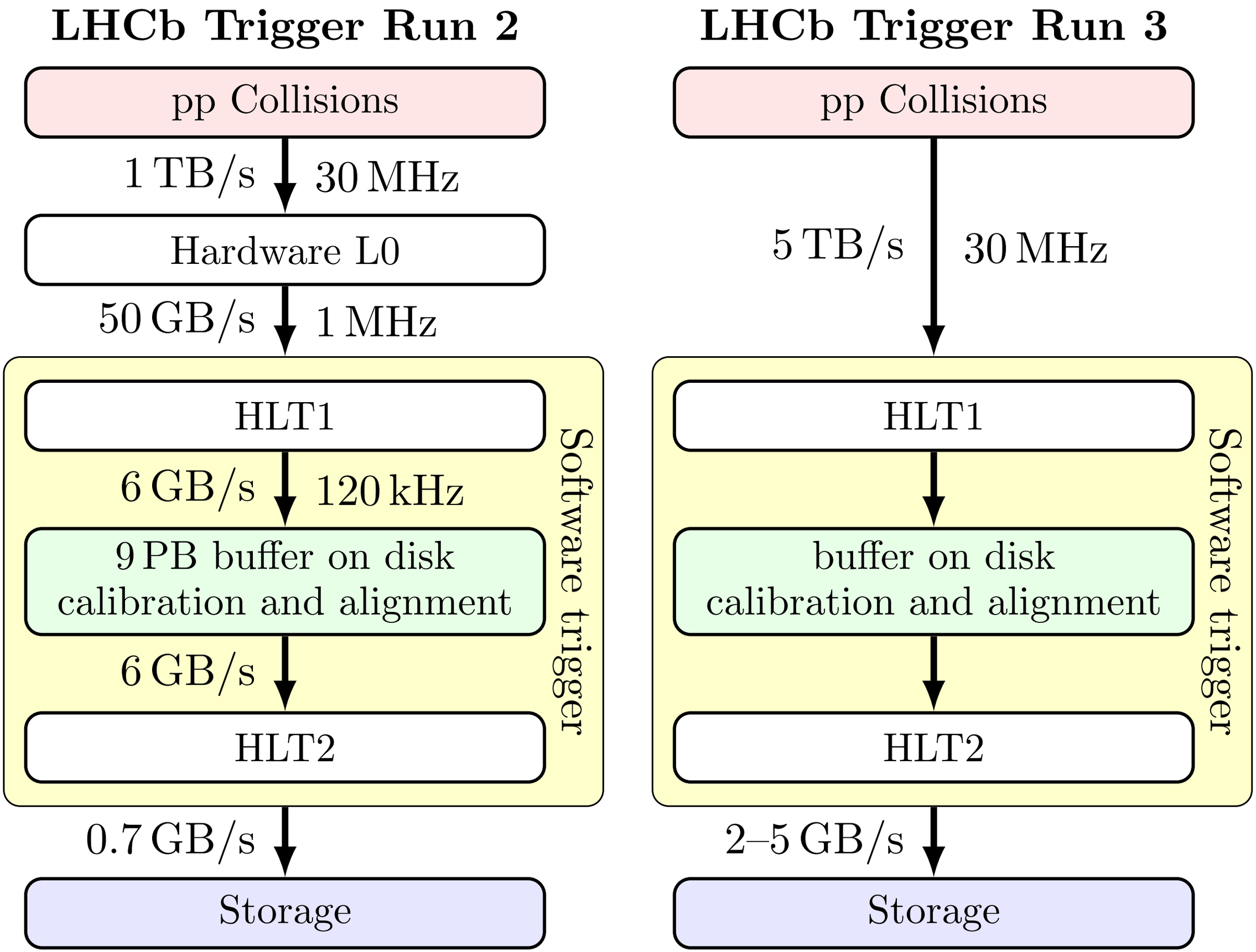
HLT1 ML Selections

About 70% of the output bandwidth from HLT1 is taken up by inclusive selections that seek to efficiently select almost any heavy flavor decay that could be of interest.



- A one-track algorithm based on the p_T and IP x^2 (track-quality criteria applied as pre-selection; there is also a version of this that only considers muons).
- A two-track (SV) algorithm based on vertex x^2, flight distance x^2, scalar track pT sum, and n(small IP x^2) tracks (also has a heavy-flavor-like preselection).

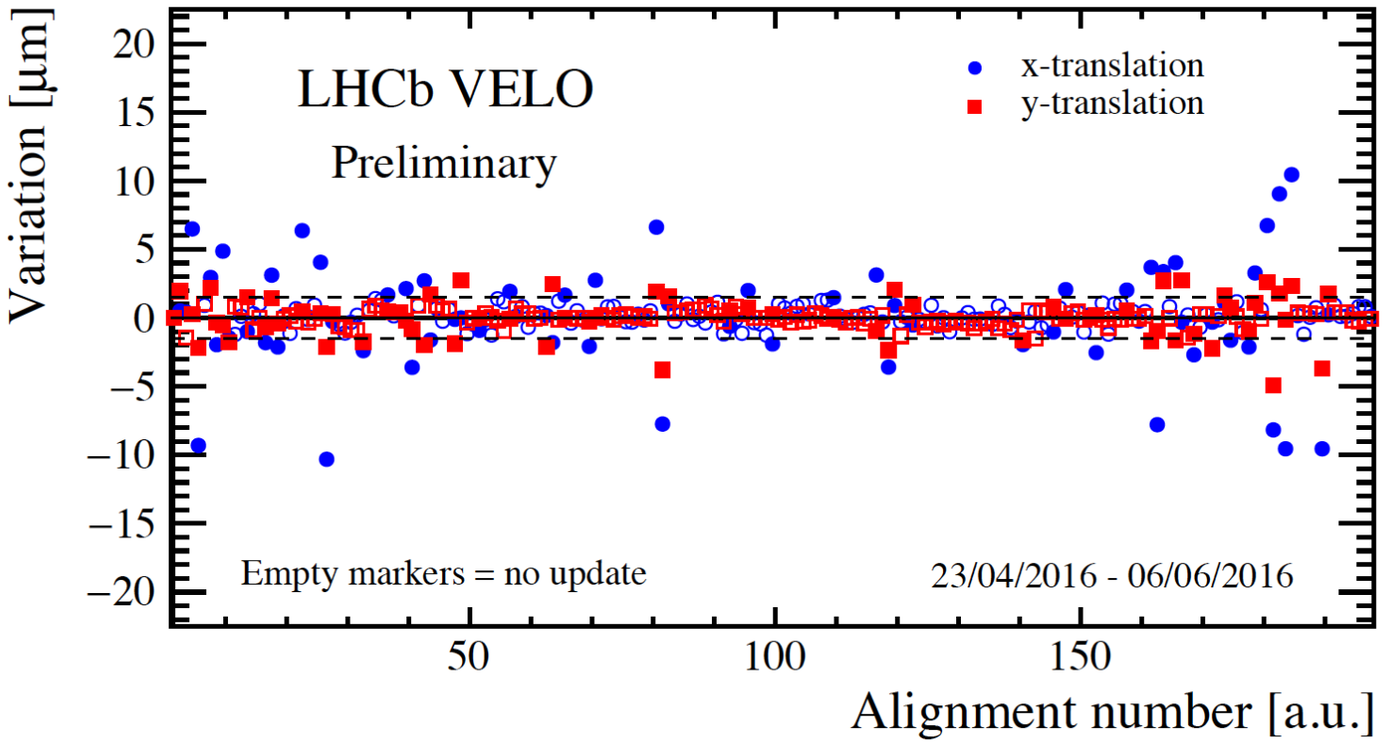
The majority of the LHCb physics program uses data selected in HLT1 by these algorithms, which use MatrixNet (trained by our Yandex friends).

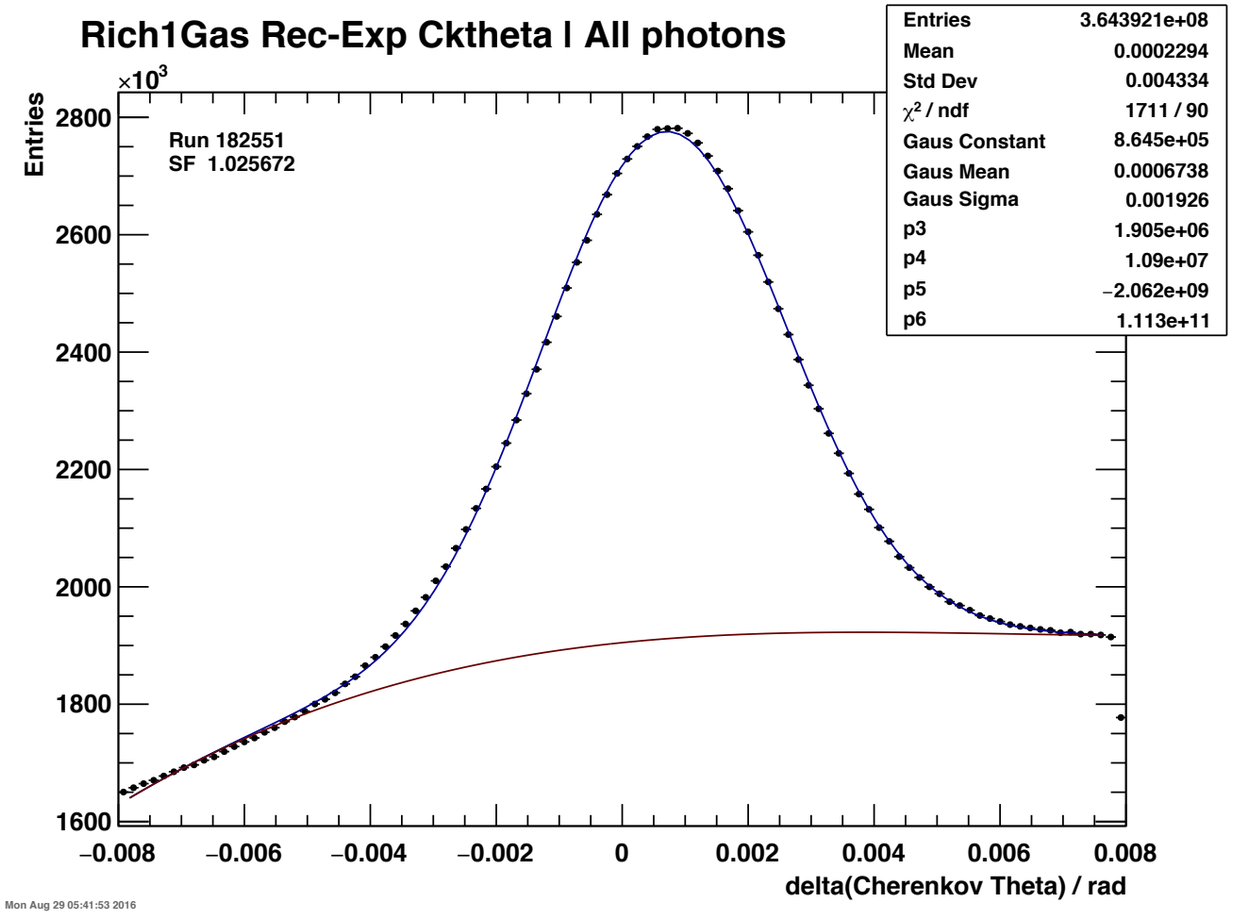

LHCb Trigger Run 2
pp Collisions
1 TB/s
30 MHz
Hardware L0
50 GB/s
1 MHz
HLT1
6 GB/s
120 kHz
9 PB buffer on disk
calibration and alignment
6 GB/s
HLT2
Software trigger
0.7 GB/s
Storage
LHCb Trigger Run 3
pp Collisions
5 TB/s
30 MHz
HLT1
buffer on disk
calibration and alignment
HLT2
Software trigger
2–5 GB/s
Storage

Real-Time Calibration

VELO opens/closes every fill, expect updates every few fills. Rest of tracking stations only need updated every few weeks.

RICH gases indices of refraction must be calibrated in real time; requires ~1 min to run, and new calibrations are required for each run.



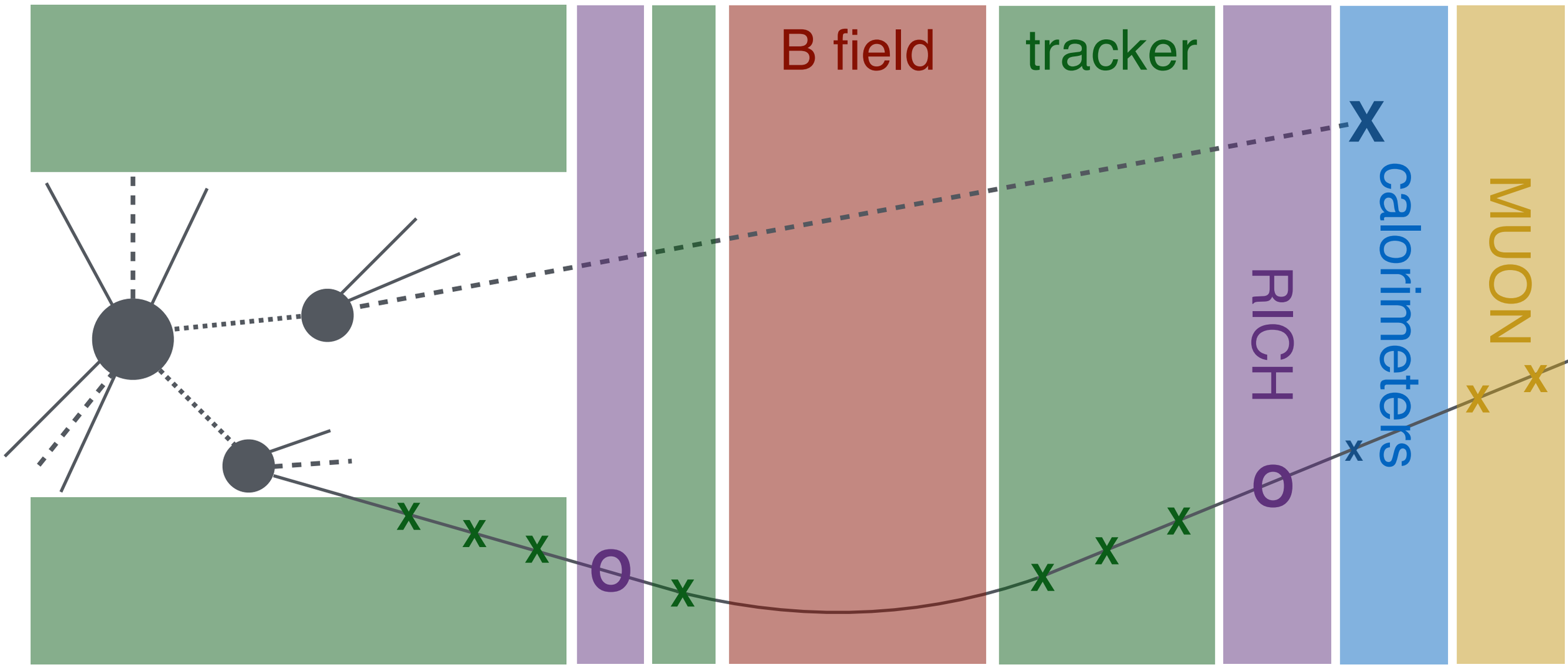


Calibration data is sent to a separate “stream” from the physics data after the first software-trigger stage. This permits running the calibrations on the online farm simultaneously with running the trigger.

(Near) real-time publication: σ(cc)[13TeV] shown @ EPS (2015) within a week of recording the data (measured using online-reconstructed data). We achieved better mass and lifetime resolution online than we had offline in Run 1. LHCb-PAPER-2015-041

HLT2

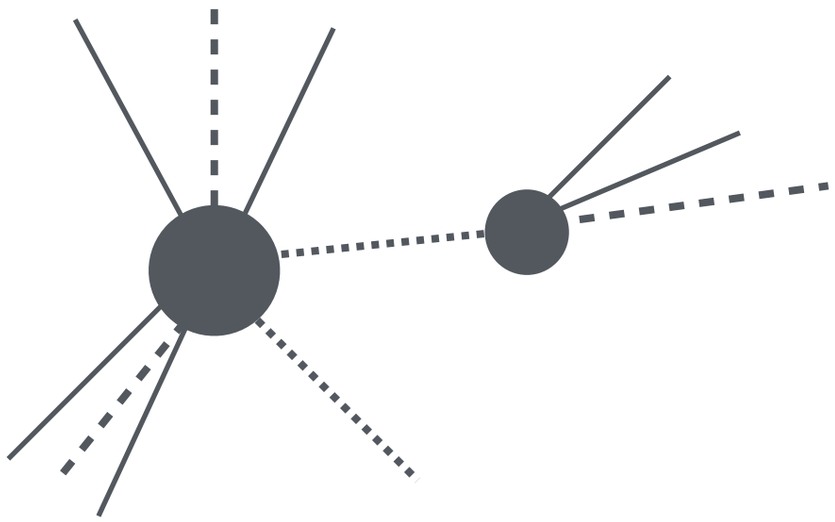
HLT2 typically runs out of fill, but must run fast enough to prevent the buffer from overflowing. LHCb runs the full event reconstruction in HLT2, including all charged particles down to $p_T > 0.07$ GeV.



The fake-track-killer NN is again used here, as are a number of other important ML-based algorithms.

HLT2 Topological Trigger

About 40% of the final output bandwidth is given to inclusive selections that seek to efficiently select almost any b-hadron decay that could be of interest.



- An SV algorithm that considers 2, 3, and 4-track vertices (seeded by HLT1 ML selections).
- The ML uses corrected mass, vertex χ^2, scalar track p_T sum, flight distance χ^2, pseudorapidty (PV-SV), min(track p_T), n(small IP tracks), IP χ^2, n(very b-like tracks).
- All features are discretized in the ML for stability, robustness, etc.

V.Gligorov, MW, JINST 8 (2012) P02013.

This algorithm has run since the start of 2011, and has collected the data used by ~200 papers! It was re-tuned for Run 2 by Yandex (now based on MatrixNet, was a BDT in Run 1).

T.Likhomanenko et al [1510.00572]

N.b., real-time alignment and calibration is NOT required to use ML in an online system.

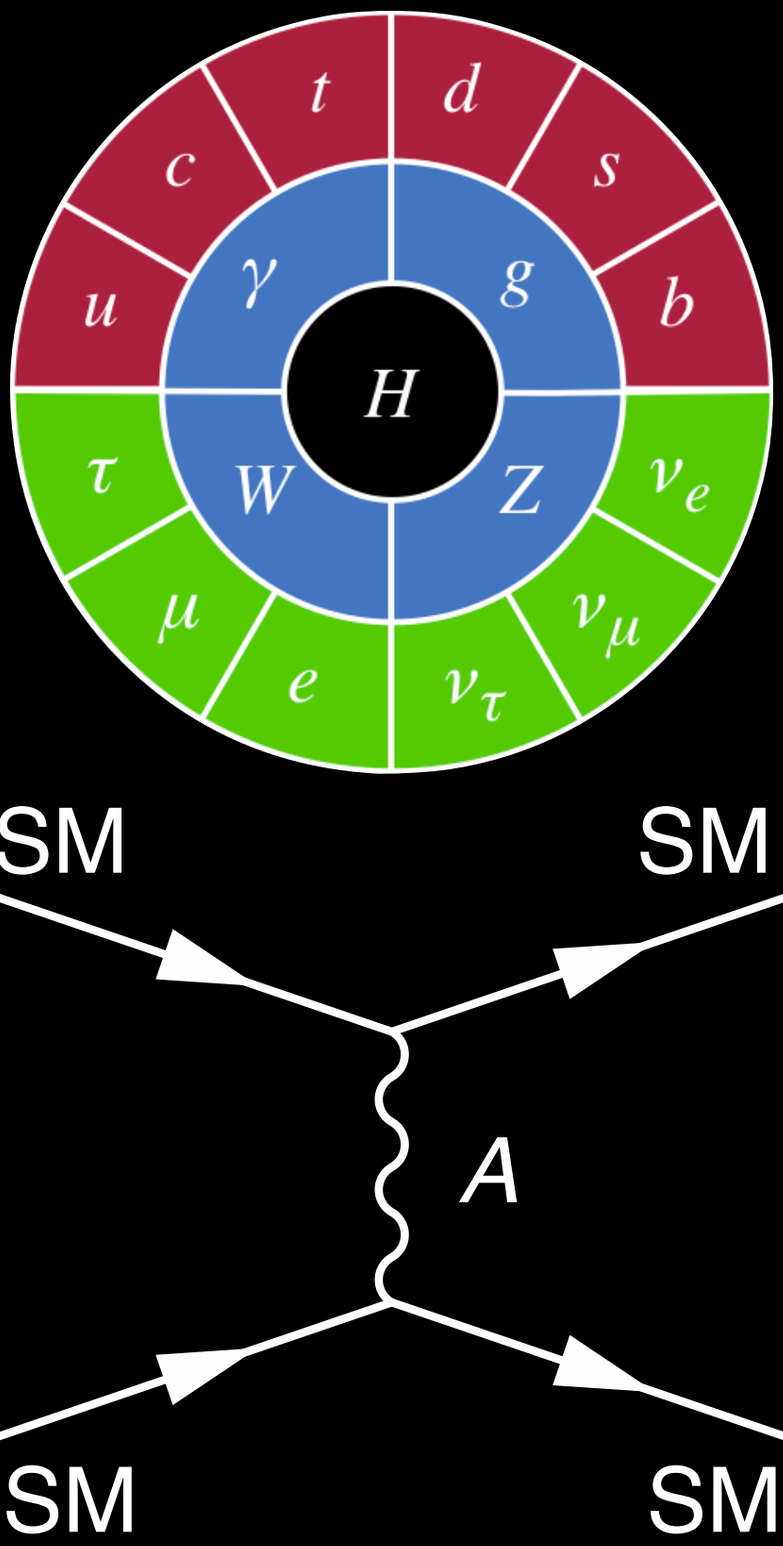
We first introduced ML into our primary event-classification algorithm at the start of 2011 data taking, but real-time calibrations were not implemented until 2015.

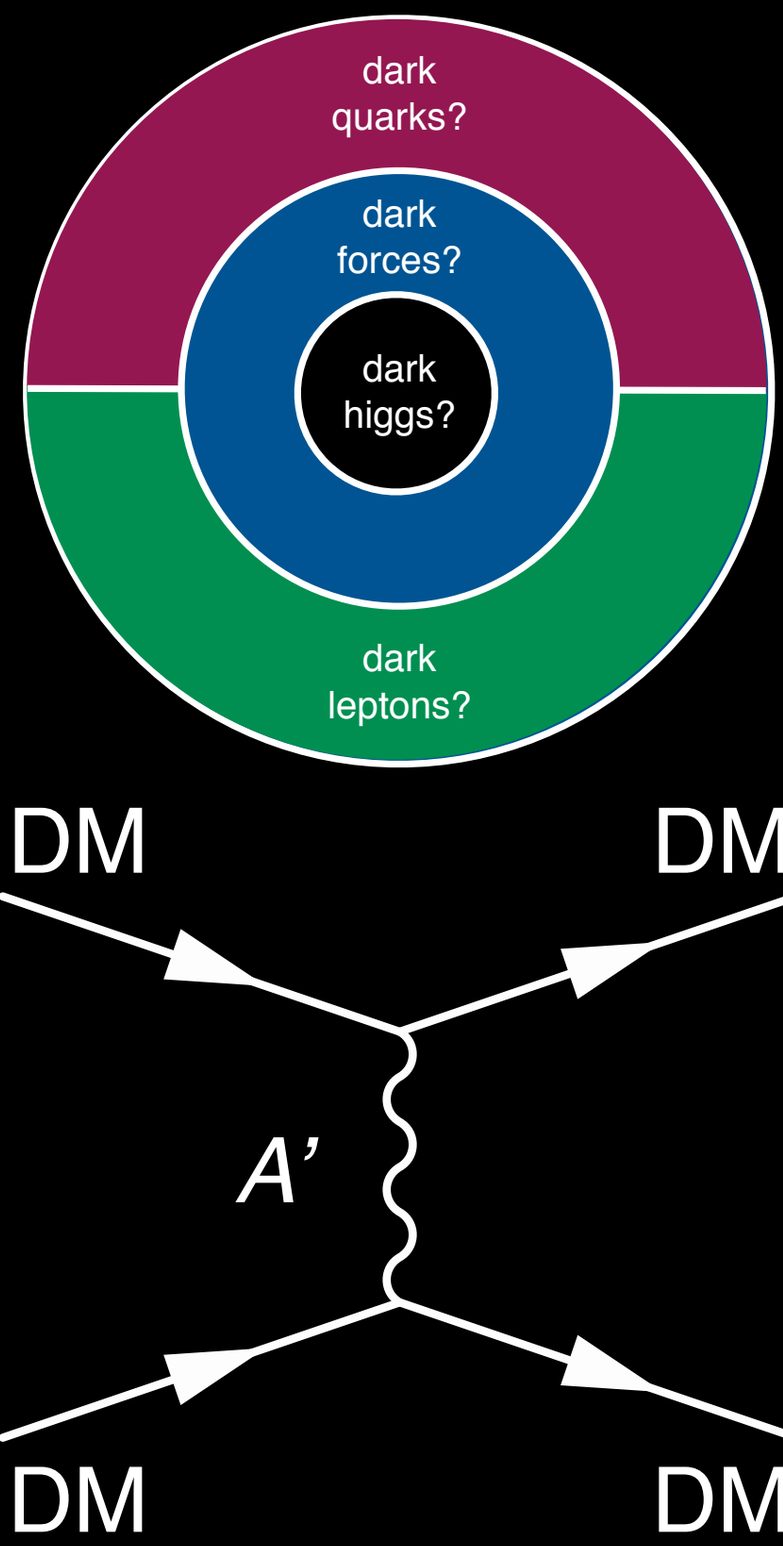
Our Run 1 ML-based trigger algorithm collected the data used in about 200 papers to date — and it was run on imperfect data (but designed to be robust against run-time instabilities).

DM-DM interactions would affect galactic formation, DM halos, galaxy mergers, etc.

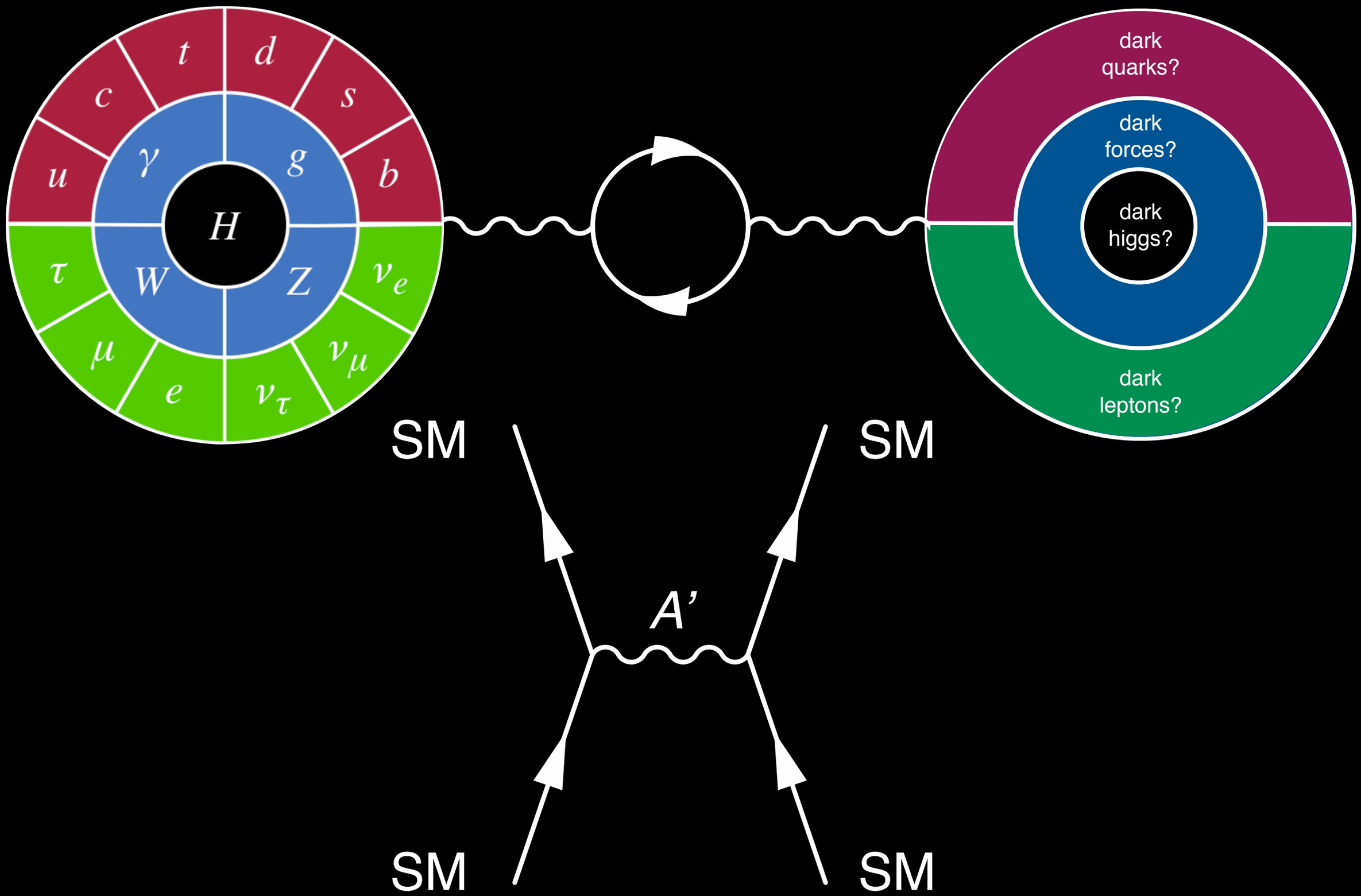
Currently none of these provide very strong constraints on DM-DM interactions, i.e. DM could be charged.

Dark Photons



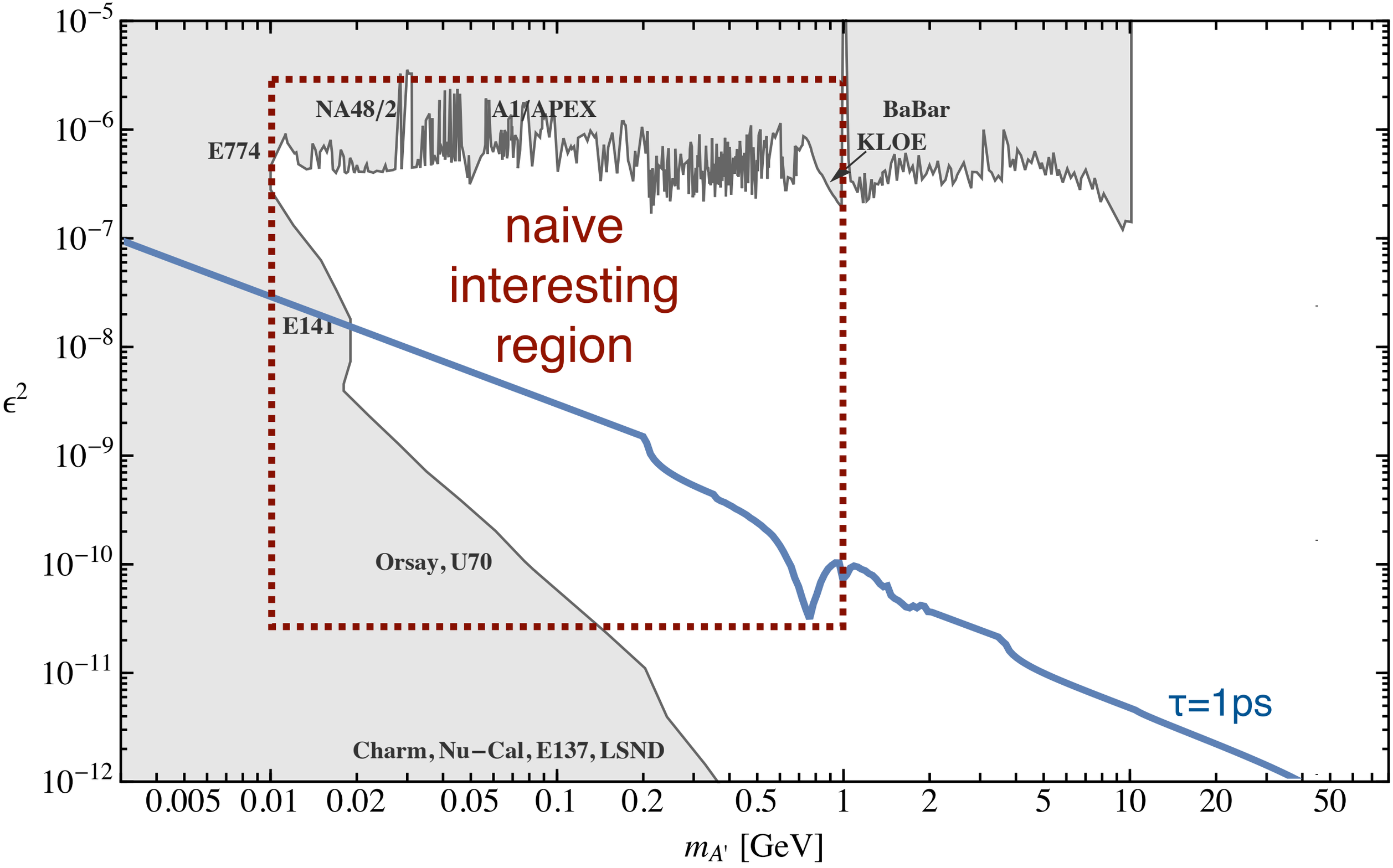


Dark Photons



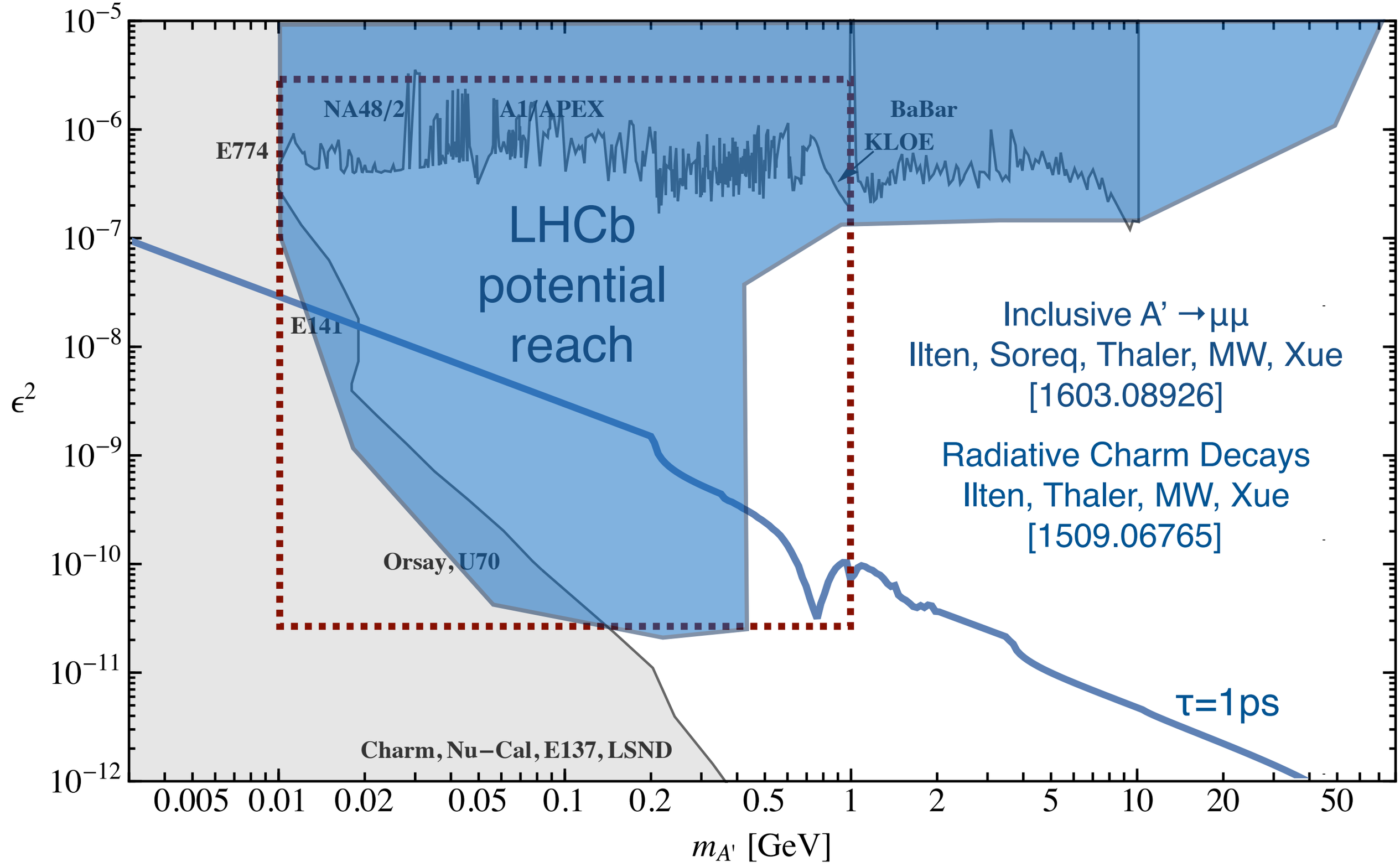
Visible A’ Decays

Existing bounds on visible A’ decays.



Visible A' Decays

LHCb sensitivity assuming we can keep all of these events.



Make It

Fake It

prompt decay

SM

SM

e^+

A’

e^-

(lots of) SM

displaced decay

SM

SM

A’

e^+

e^-

(lots of) SM

γ^*

e^+

e^-

SM (mis-ID)

SM (mis-ID)

b,c-hadron

e^+

(lots of) SM

SM (mis-ID)

b,c-hadron(s)

e^+

e^-

(lots of) SM

γ

material

e^+

e^-

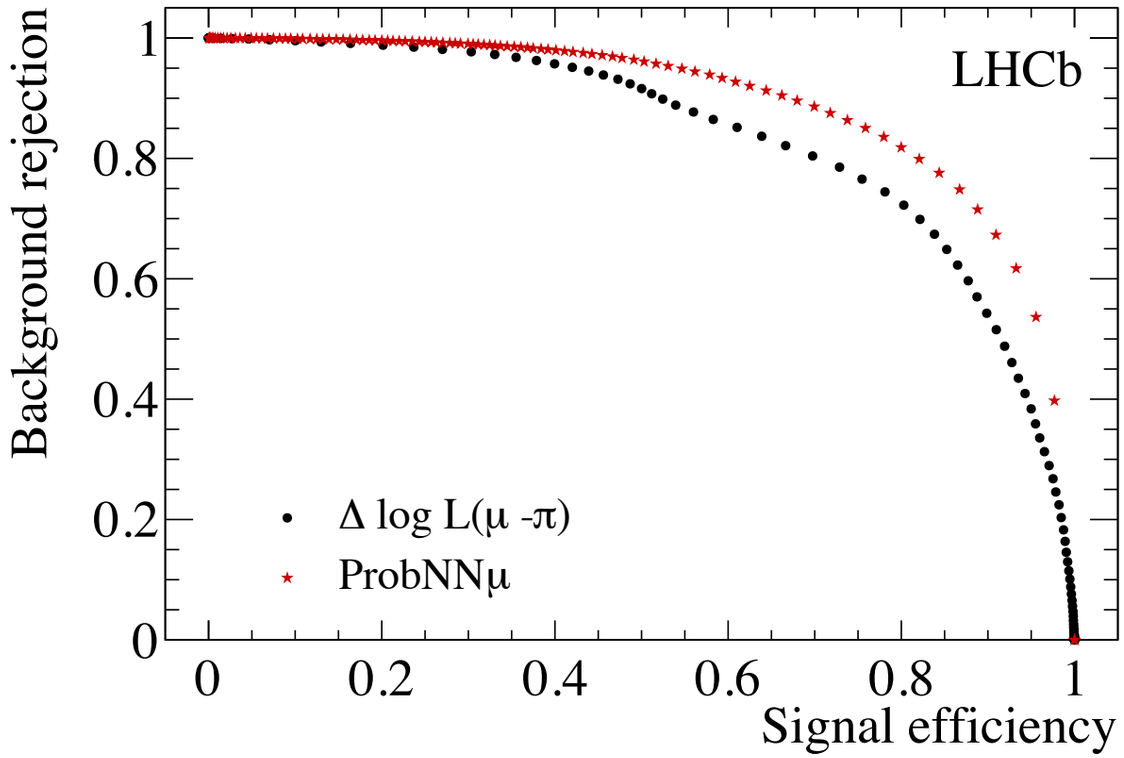
Charged PID

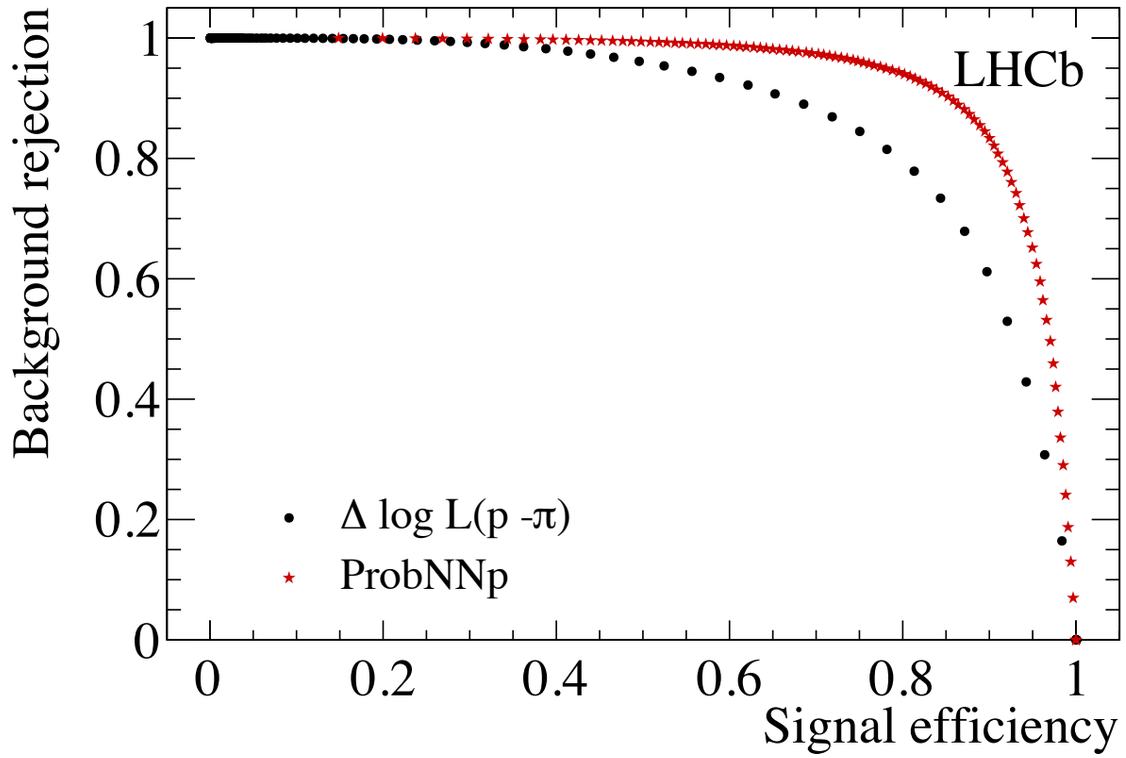
Charged PID: determining whether a track originates from an e, μ, π, K, p, or fake.

Info from the tracking, calorimeter, RICH, and muon systems all play an important role here—and are correlated.

PID NNs

Single-hidden-layer NN trained on 32 features from all subsystems. Each is trained to identify a specific type of particle (or fake track).



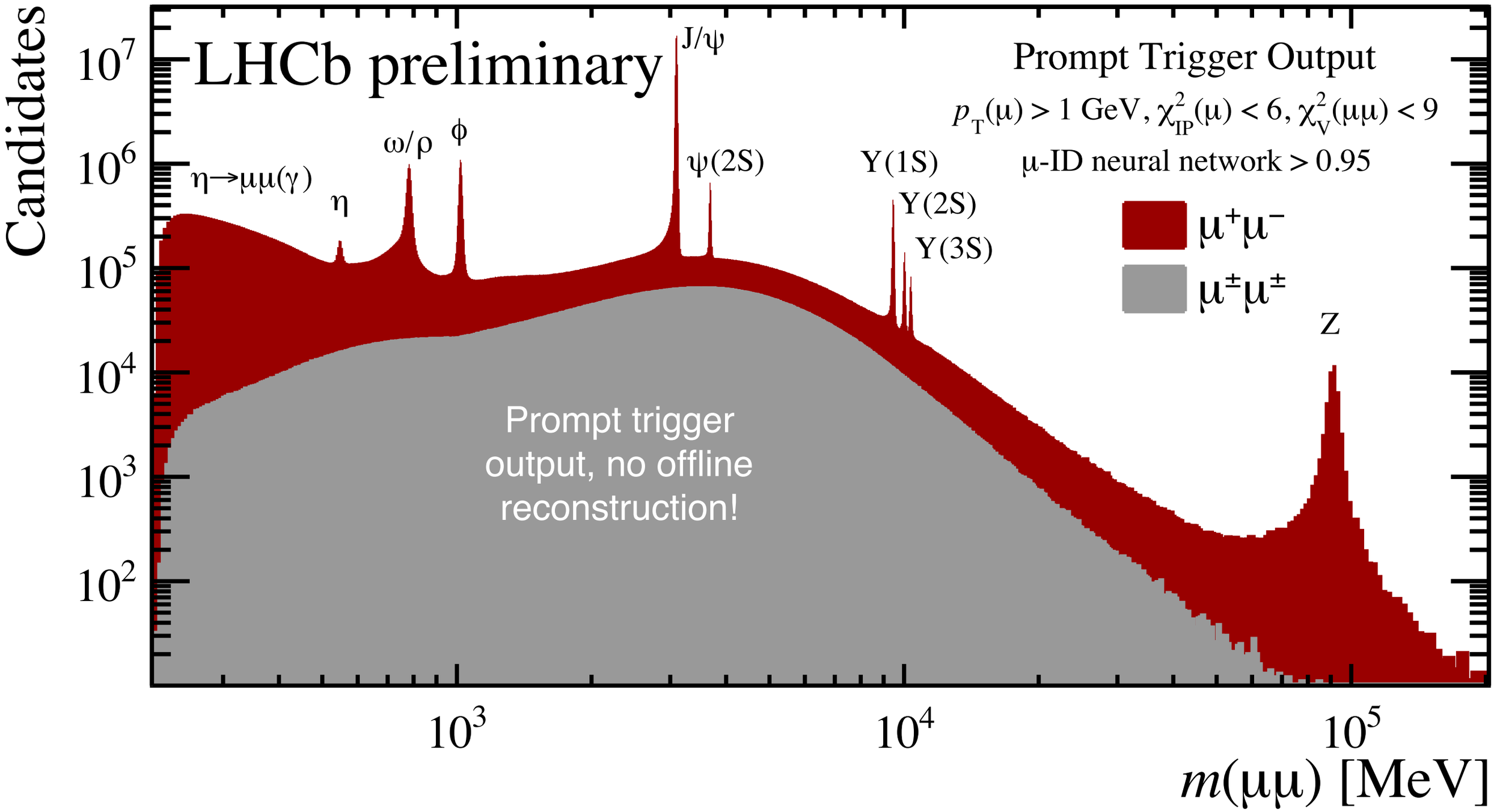


Typically get a factor of 3x less pion contamination in a muon sample than using the CombDLL approach — 10x less in a dimuon sample.

Currently exploring state-of-the-art: XGBoost ~ Deep NN ~ 50% less BKGD than basic BDT or ANN, which again give 2-3x less BKGD than DLLs.

Dark Photons?

New triggers in 2016 for both prompt and displaced dark-photon searches (rely heavily on advances to the LHCb online system in Run 2–including implementing NN-based PID in the trigger).



More ML?

LHCb uses a lot of ML in the trigger, but are there places where we could start using ML to do even better?

- Performing the final calibrations and event reconstruction online means that, in principle, there is no reason to write out the raw event to storage. In practice, we still mostly write out the raw event too, though for about 1/4 of the 5 PB/year we keep, only the reconstructed objects are written. Furthermore, each selection can choose which subset of objects to persist. However, the data rate is still a problem, and will be even more so in Run 3. Can we use ML (e.g. autoencoders) to perform lossy (but physics lossless) compression?
- Performing tracking down to 0.5 GeV in p_T in real time will be a challenge in Run 3. We use ML to kill fake tracks, but can we do this earlier in the tracking to save time? Can we use ML to learn the covariance matrix elements to avoid having to Kalman filter all tracks? Can ML recognize decay signatures without requiring doing serial track finding first?
- Etc.

Details

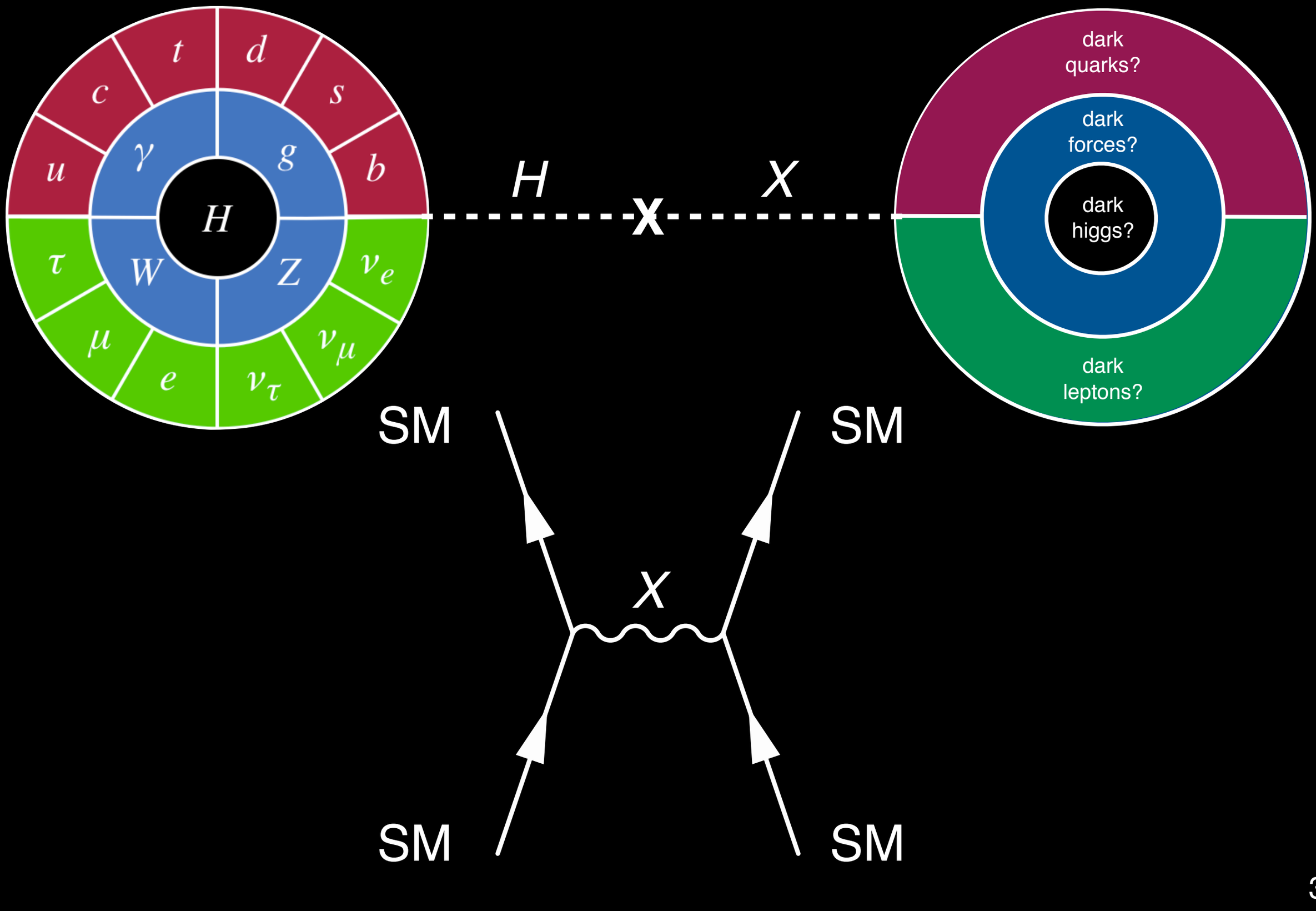
•We typically train our ML algorithms on MC, then characterize their performance using data control samples (same way we characterize our hardware). In principle, data samples could also be used in the training, but then one would need to deal with BKGD in those samples (this is not hard to do using event weights). N.b., make sure to not confuse non-optimal with wrong or the inverse.

•Dimensional reduction achieved by ML makes it possible to maximize performance without complicating data-driven validation. There are many standard candles at the LHC to use for data-driven validation.

•As an aside, systematics tend to scale with inefficiency, so a highly-performant black box often incurs a smaller systematic than a simple, less performant algorithm — and also is easier to deal with than hardware (of course, there are exceptions).

•Bottom line: We use ML because it enables great science. It greatly improves performance in many areas, even converting some measurements from infeasible to simple & precise. The LHCb trigger is even mostly ML-based, and has been since the start of 2011 data taking (see V.Gligorov, MW, JINST 8 (2012) P02013).

Higgs Portal

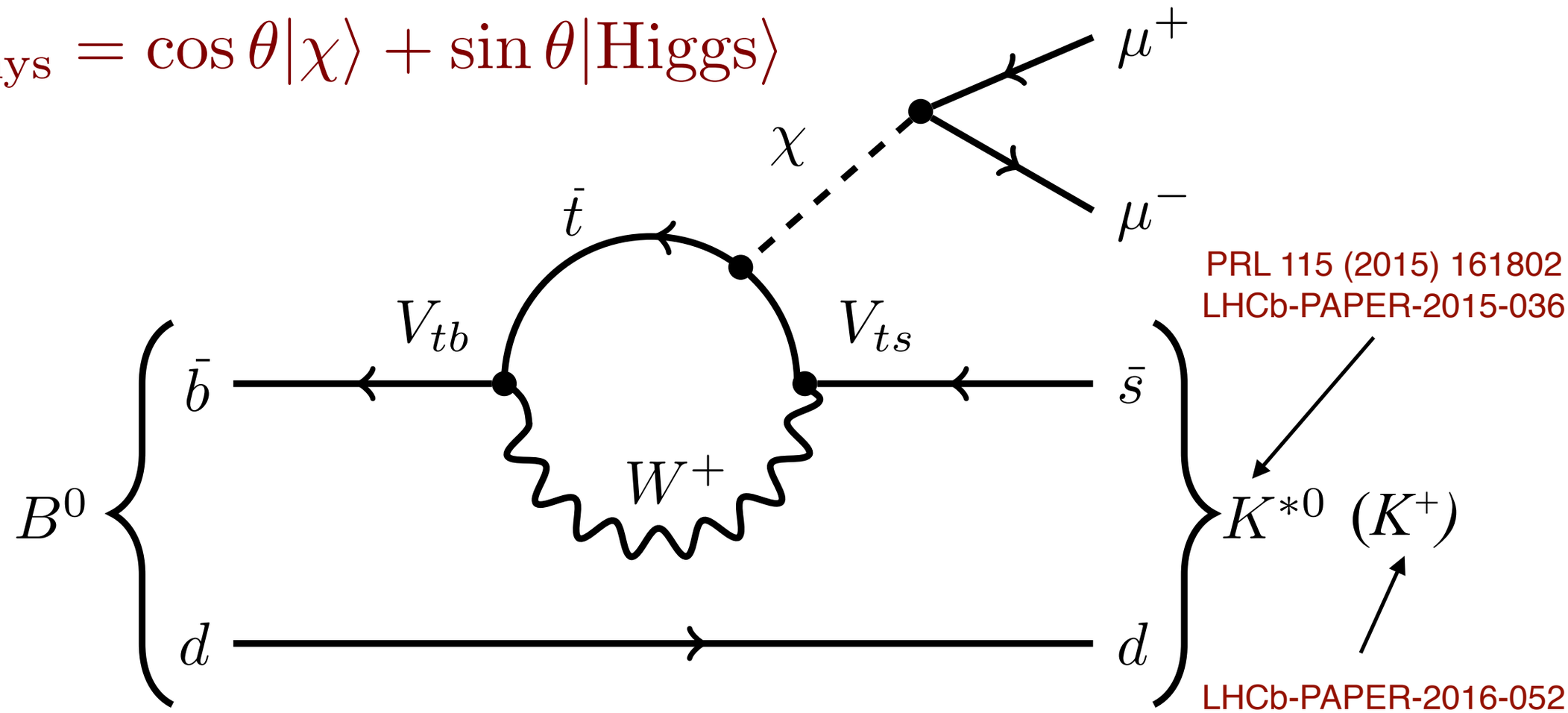


Higgs Portal

b→*s* penguin decays are an excellent place to search for low-mass hidden-sector particles (e.g., anything that mixes with the Higgs sector).

$$|\mathrm{Higgs}\rangle_{\mathrm{phys}} = -\sin\theta|\chi\rangle + \cos\theta|\mathrm{Higgs}\rangle$$

$$|\chi\rangle_{\mathrm{phys}} = \cos\theta|\chi\rangle + \sin\theta|\mathrm{Higgs}\rangle$$



PRL 115 (2015) 161802
LHCb-PAPER-2015-036

LHCb-PAPER-2016-052

Model-independent limits set on $B(B\to K^*X)B(X\to\mu\mu)$ translate into model-dependent constraints at the PeV scale for ALPs and mixing angles down to 0.1 mrad between the X and H fields between 2m(μ)-2m(τ).

Uniform Boosting

The Higgs-portal search covers a factor of 20 in mass, and 4 orders of magnitude in lifetime. Cannot generate a MC sample for every mass and lifetime value, so how do we do the search?

- Generate MC samples at 15 points that roughly span the mass-lifetime plane that we want to explore.
- Use 10 of these for training/validation, and hold back 5 for testing.
- Redefine the loss function to understand that we do not know the true mass and lifetime value of the hidden-sector boson. Technically, we want to de-correlate the ML response from the location in the mass-lifetime plane (see J.Stevens, MW [1305.7248]; A.Rogozhnikova, A.Bukva, V.Gligorov, A.Ustyuzhanin, MW [1410.4140]).
- Find that the uniform BDT performs nearly identically on the 5 samples not used in training as it does on the 10 in the training—and on these 5 samples it does much better than any traditional algorithm tried (essentially, they all learn the masses of the training samples).

Many other use-cases, e.g., avoiding creating peaking backgrounds, etc.

Jet Parton Identification (Tagging)

Heavy Flavor Jets

JINST 10 (2015) P06013
LHCb-PAPER-2015-016

Jets that originate from a b or c quark are of particular interest in HEP, e.g., studying H, top, PDFs via V+jet(s), etc. Look for an SV in the jet from a b or c hadron decay, which occurs about 70%, 25%, 1% of the time for b, c, light jets. Next, use SV features to discriminate:

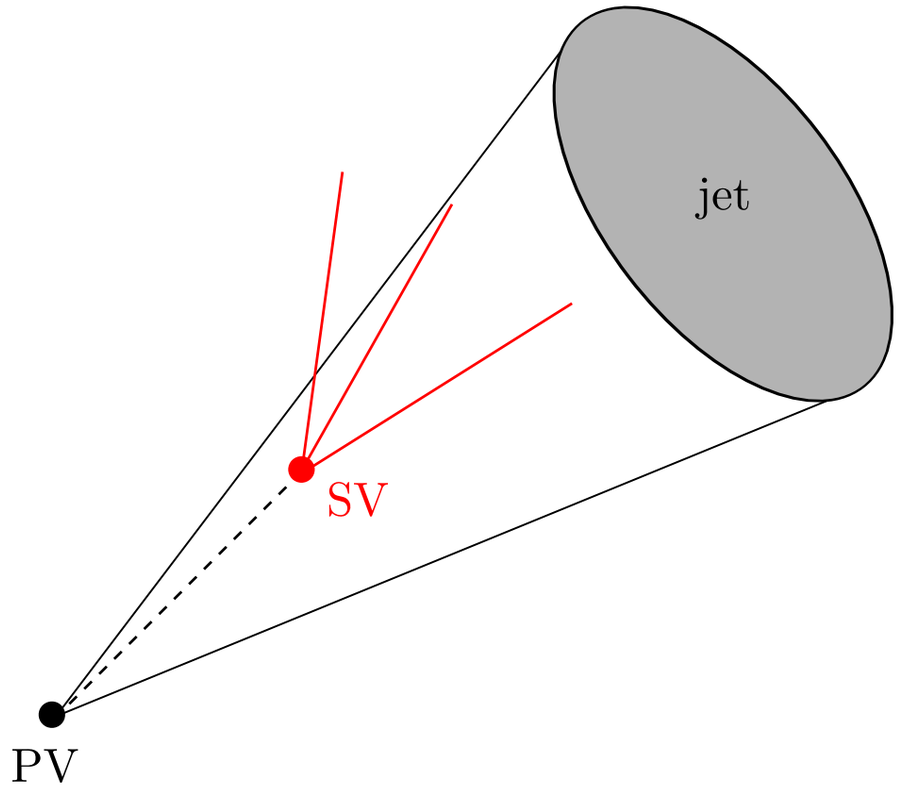
•mass and “corrected” mass;

•transverse distance from and flight distance x^2 of PV to SV;

•p_T(SV)/p_T(jet) and ΔR(SV,jet);

•number of tracks in SV, number not in the jet, and sum of IP x^2 of all SV tracks;

• net charge of the SV.



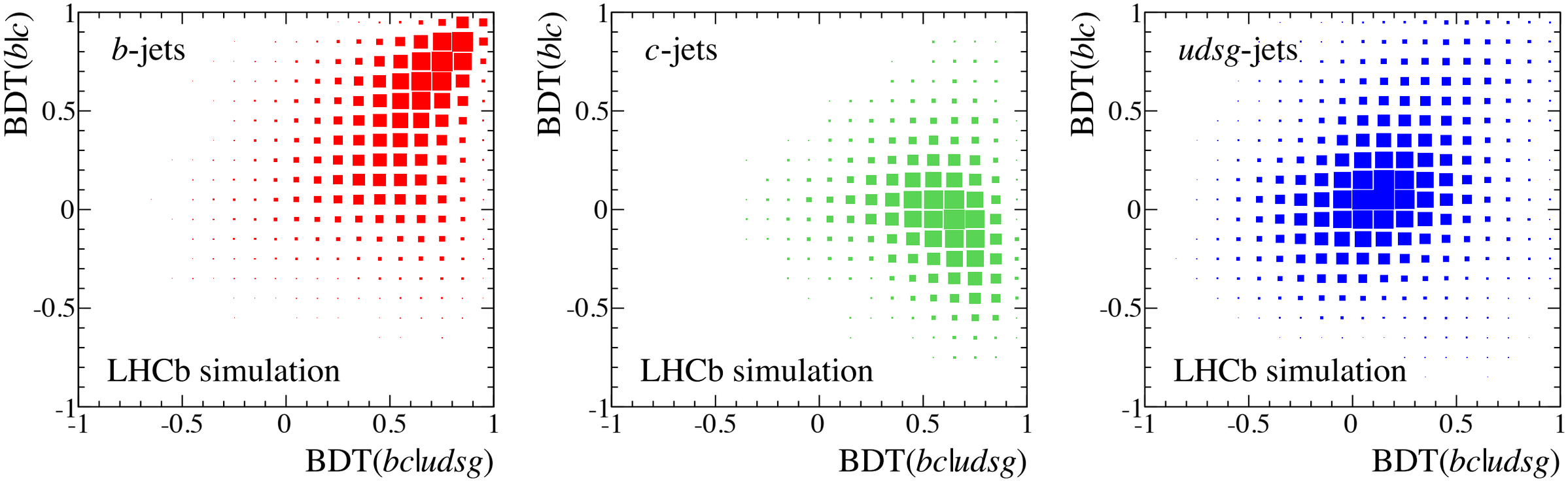
Each feature provides some discrimination power, but typically only between b,c vs light or b vs c—none are powerful enough to fully separate types.

ML Jet Tagging

JINST 10 (2015) P06013
LHCb-PAPER-2015-016

Put 10 features into two BDTs: one for b,c vs light, and another for b vs c. No feature can fully separate types, but their correlations (largely) can.

LHCb simulation: each distribution normalized to one; 70%, 25%, 1% of b, c, light jets are tagged.



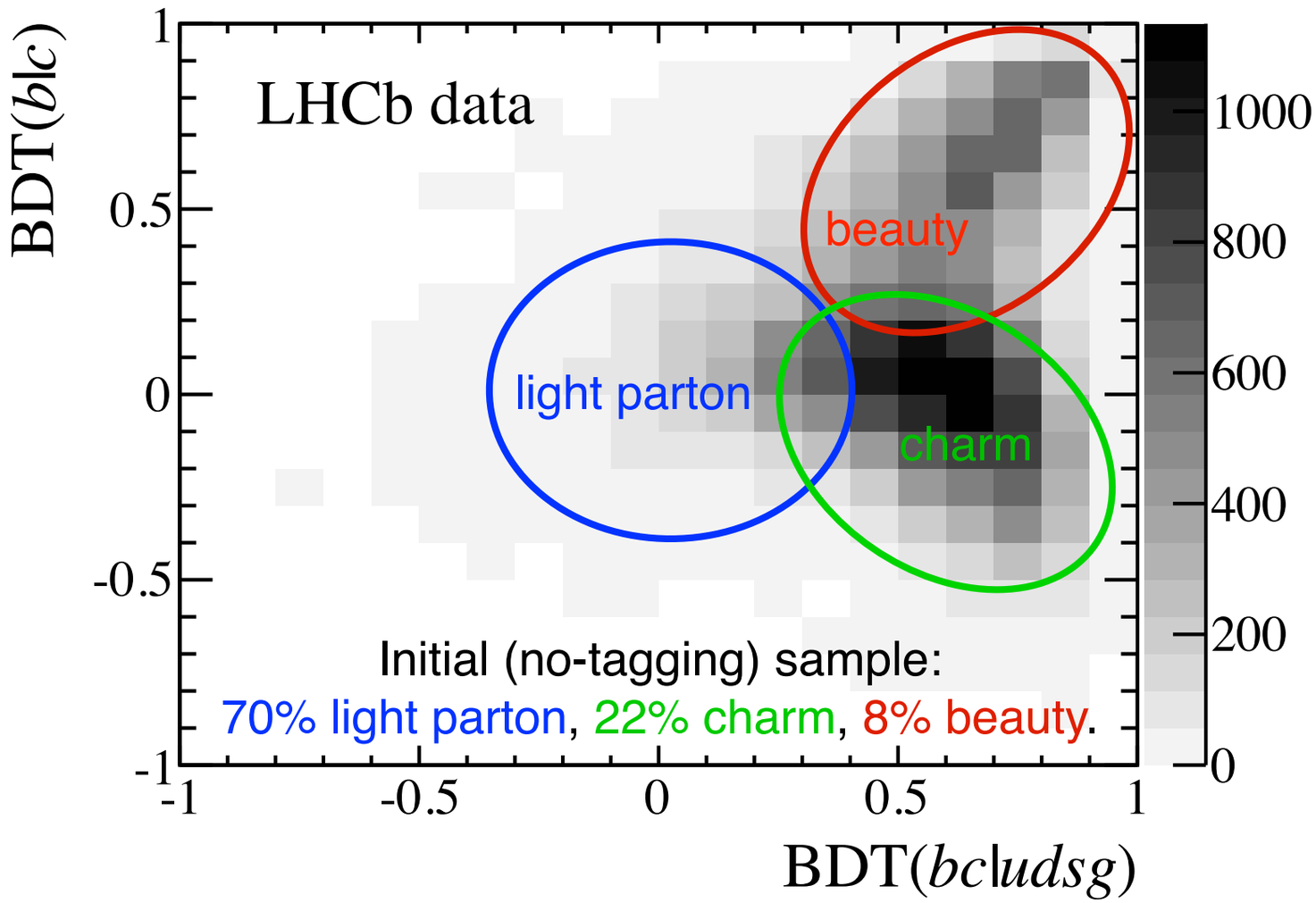
Could cut on BDT responses to obtain high-purity b-jet or c-jet samples. Alternatively, fit 2-D BDT distribution to extract the b-jet and c-jet yields.

Looked at doing a single 3-class algorithm but that doesn't seem to help here (shown to work better in other applications).

ML Jet Tagging

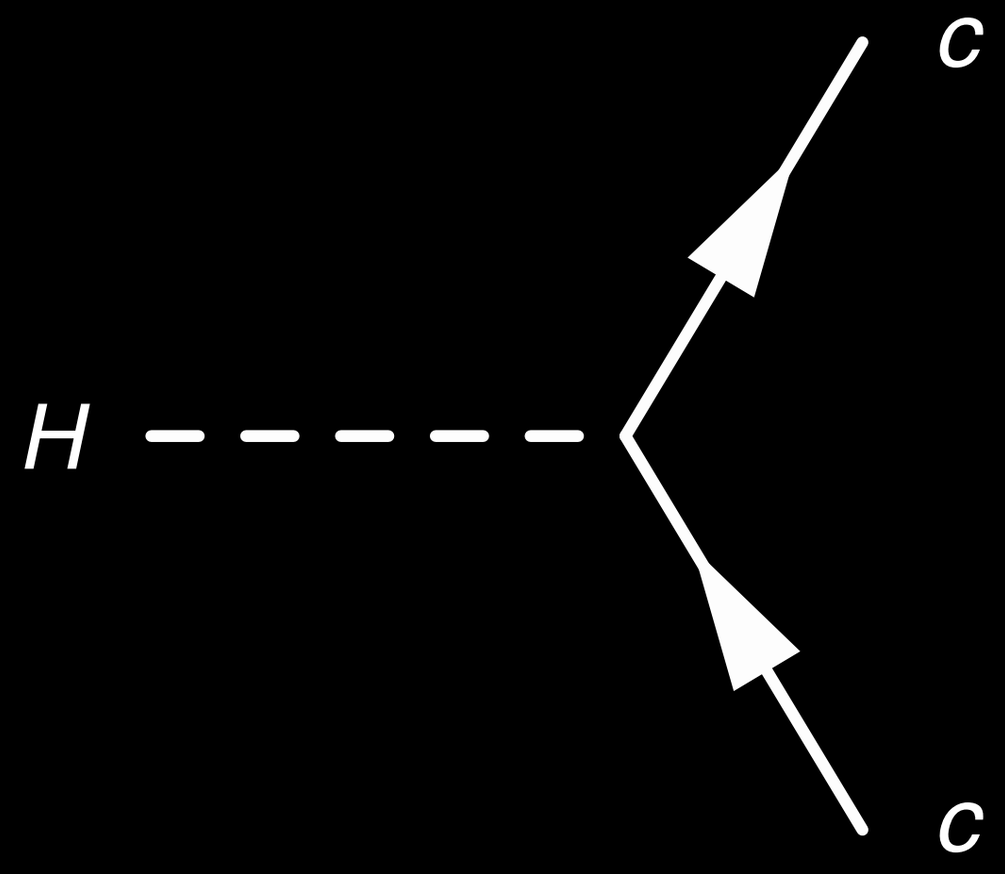
JINST 10 (2015) P06013
LHCb-PAPER-2015-016

2-D BDT plane (nearly) optimally utilizes 10-D info to ID b, c, and light jets.



Performance validated & calibrated using large heavy-flavor-enriched jet data samples (2-D data validation much easier than 10-D!). Some analyses cut on these BDT responses, others fit the 2-D distributions to extract b,c,l yields.

ML-based jet tagging, combined with LHCb's excellent SV resolution, will permit measuring the charm Yukawa coupling down to a value of about 2xSM by the end of the HL-LHC era.



Main limitation becomes luminosity (small SM branching fraction) and "irreducible" SM Wcc background. Can we use ML to reduce it?

Details

•We train our jet-tagging BDTs on MC then calibrate and measure their performance using data control samples.

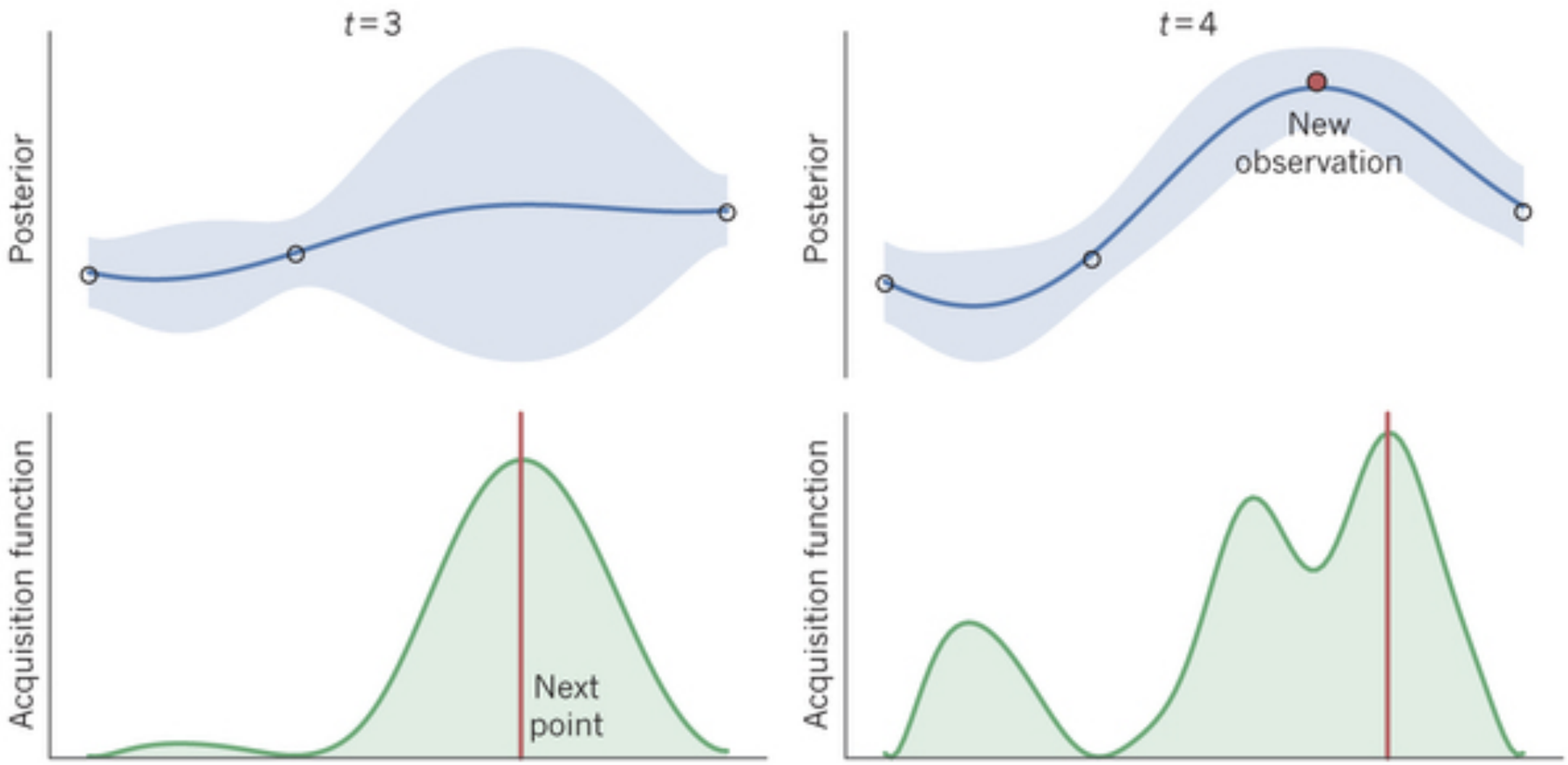
•To reiterate: it's vital that ML algorithms be validated in a data-driven manner! (What this means varies for different applications of course.)

•Dimensional reduction achieved by such algorithms makes it possible to maximize performance without complicating validation.

•A lot of interesting work ongoing now to discriminate between quark and gluon jets using features like multiplicity, energy sharing, etc., and in jet substructure and other jet topics.

Bayesian Optimization

Bayesian optimization refers to a family of methods that do global optimization of black-box functions (no derivatives required).



Start from prior for objective function, treat evaluations as data and produce a posterior used to determine the next point to sample.

See Ilten, MW, Yang, [arxiv:1610.08328], with fully working code on GitHub here: https://github.com/yunjie-yang/TuneMC (could also be used for data calibration, or any black-box problem).

Event generator tuning using Bayesian optimization

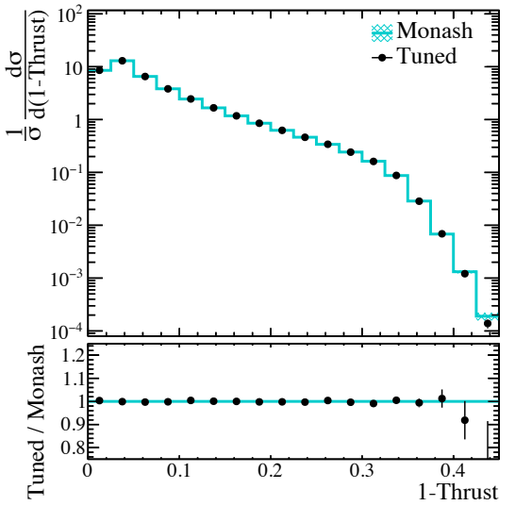
Philip Ilten, Mike Williams, and Yunjie Yang

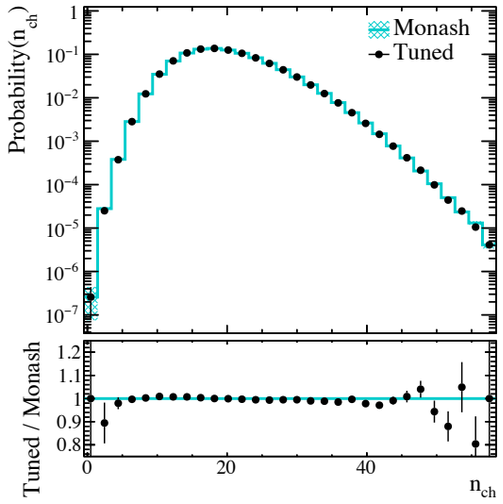
Laboratory for Nuclear Science, Massachusetts Institute of Technology, Cambridge, MA 02139

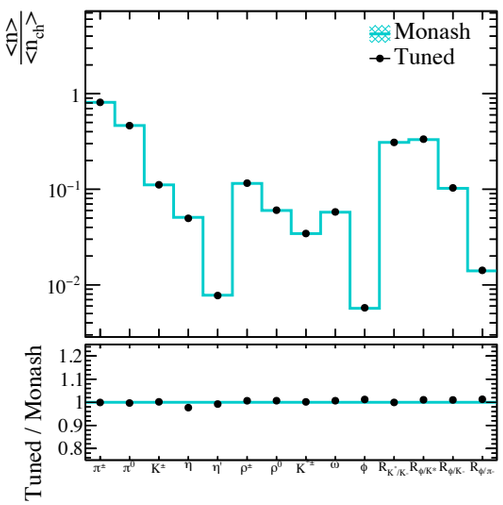
ABSTRACT: Monte Carlo event generators contain a large number of parameters that must be determined by comparing the output of the generator with experimental data. Generating enough events with a fixed set of parameter values to enable making such a comparison is extremely CPU intensive, which prohibits performing a simple brute-force grid-based tuning of the parameters. Bayesian optimization is a powerful method designed for such black-box tuning applications. In this article, we show that Monte Carlo event generator parameters can be accurately obtained using Bayesian optimization and minimal expert-level physics knowledge. A tune of the PYTHIA 8 event generator using e^+e^- events, where 20 parameters are optimized, can be run on a modern laptop in just two days. Combining the Bayesian optimization approach with expert knowledge should enable producing better tunes in the future, by making it faster and easier to study discrepancies between Monte Carlo and experimental data.

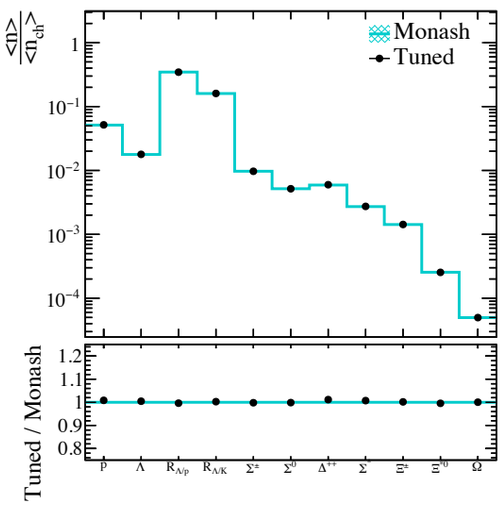
Pythia Tune

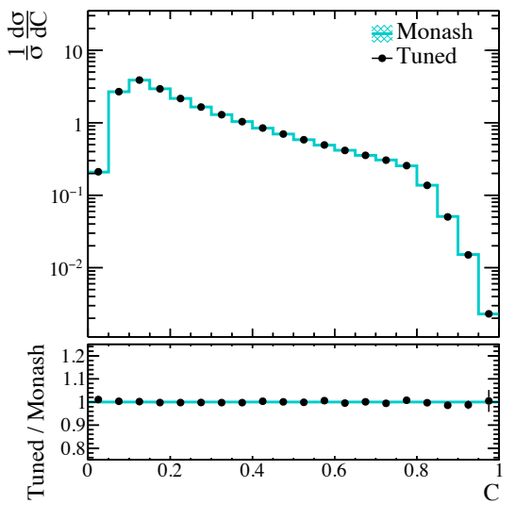
Some example results.

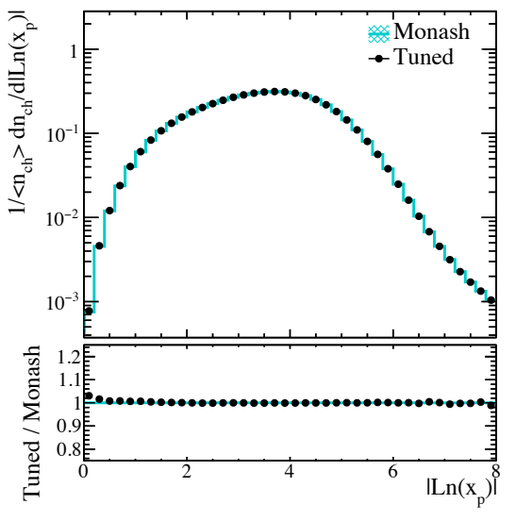


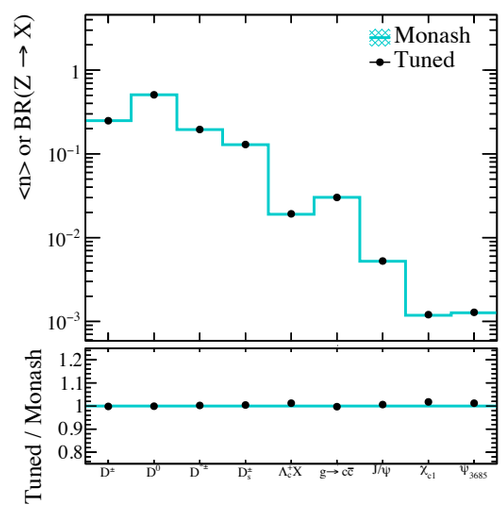


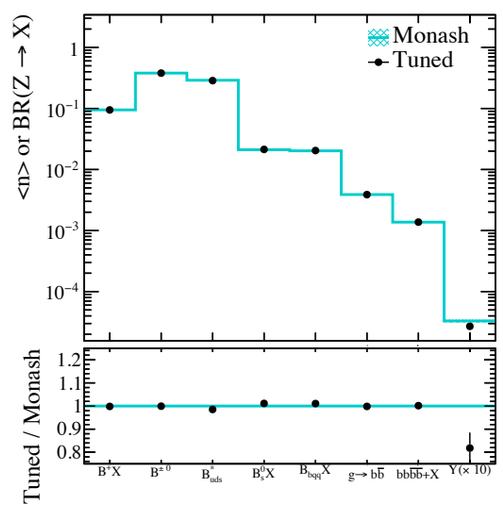


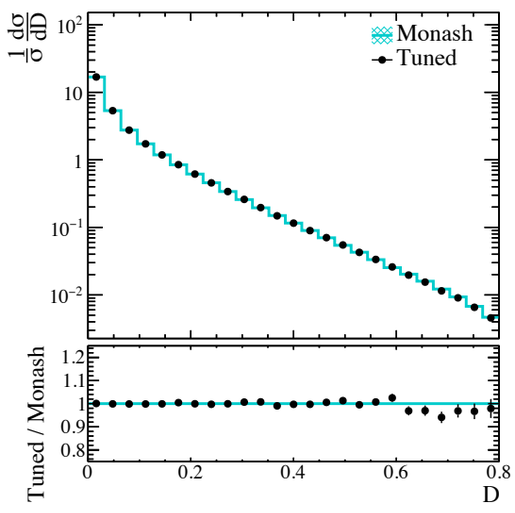


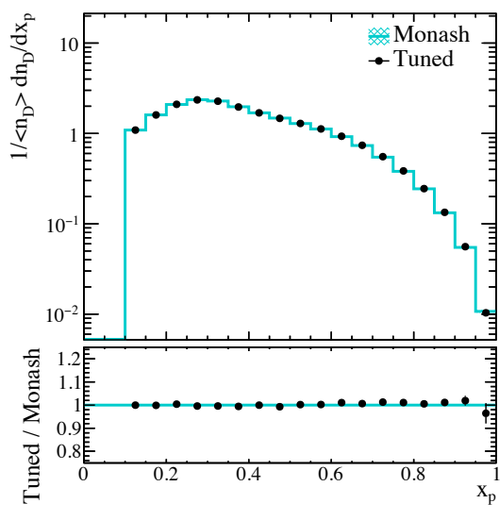


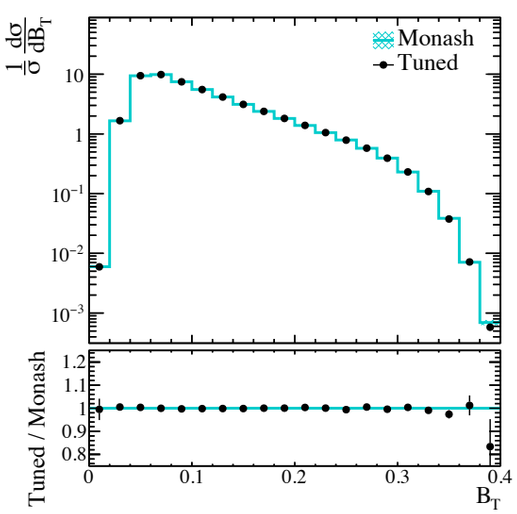


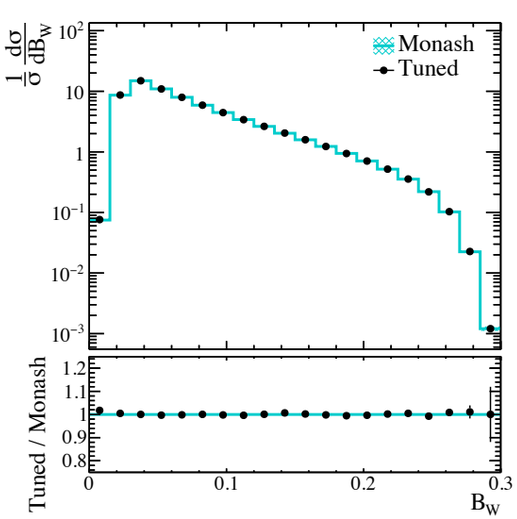












Summary

Real-time calibration works, moving to a triggerless readout will provide even bigger gains, ML usage is ubiquitous — all of these enable great science!

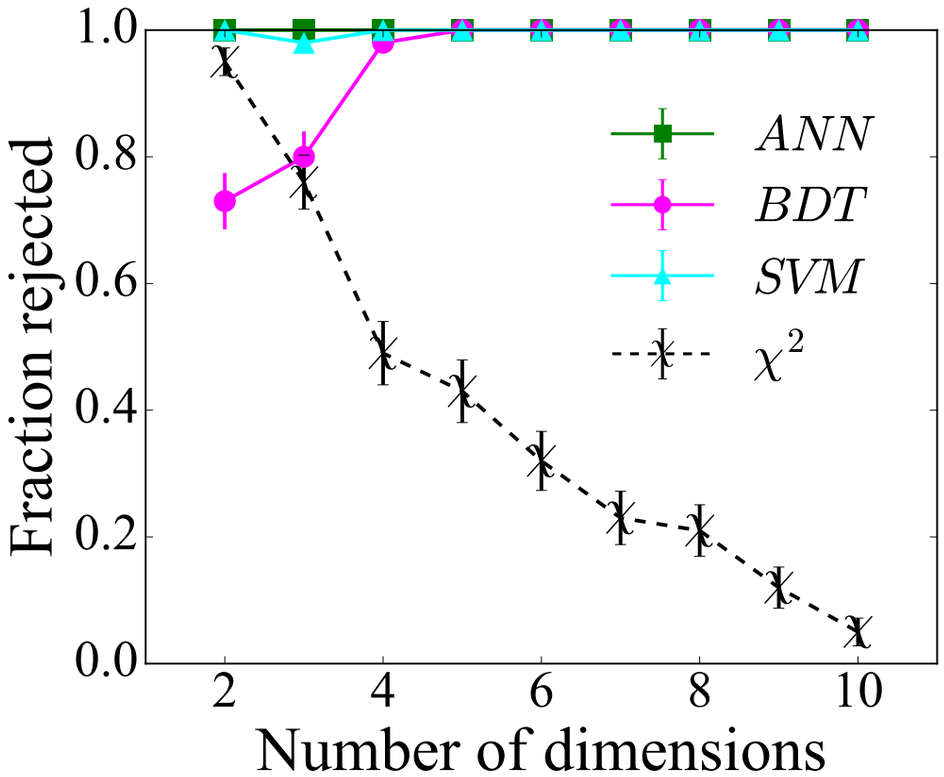
Many useful tools provided in the HEP-ML package pypi.python.org/pypi/hep_ml/0.2.0 and in REP https://github.com/yandex/rep (both produced by our colleagues at Yandex).

ML & GoF

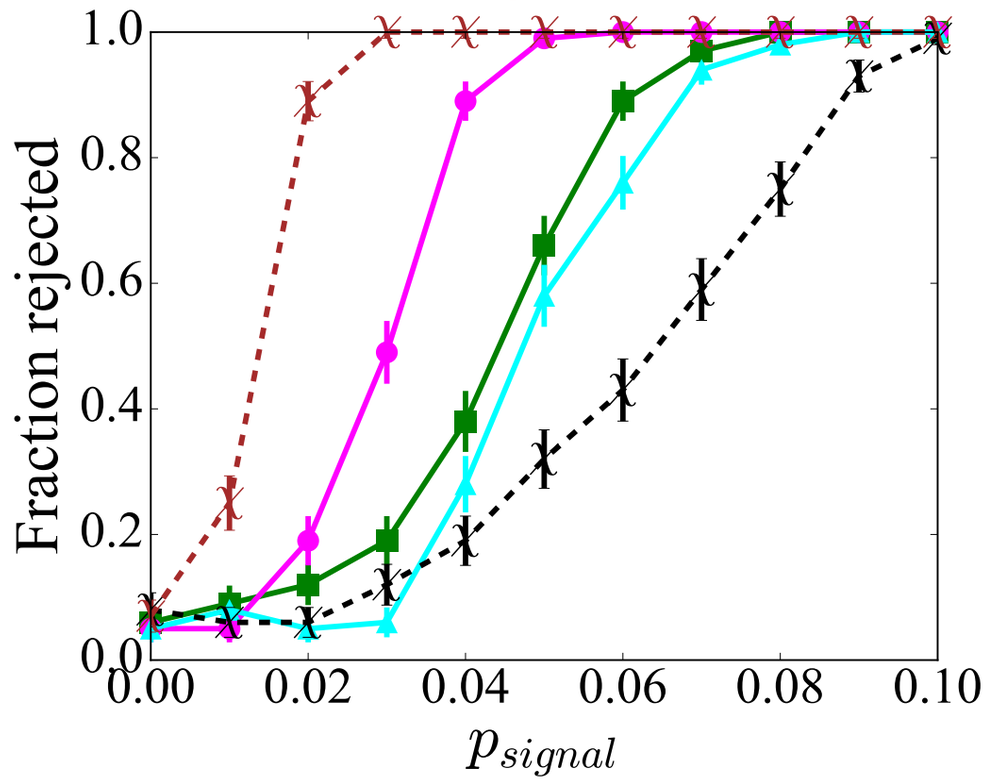
Since ML algorithms learn dimensional reduction, they can also be used to do goodness of fit in high dimensions. This is simple: train a ML algorithm using the data and an MC sample generated from the fit PDF, produce an unbiased 1-D ML response distribution for each data type, then do a 1-D GoF test (e.g. chisquare) on these 1-D distributions (simple).

Weisser, MW [1612.07186]

N-D Gaussian (5% width change)



2-body decay using 4-vectors (not mass)



The ML learns an approximation of the mass here. In this toy example, the mass (which is a weird 8-D manifold) is optimal. Knowing that—and using it—we can beat the machine (Whiteson et al showed Deep Learning can really learn things like the mass and other human-designed features).