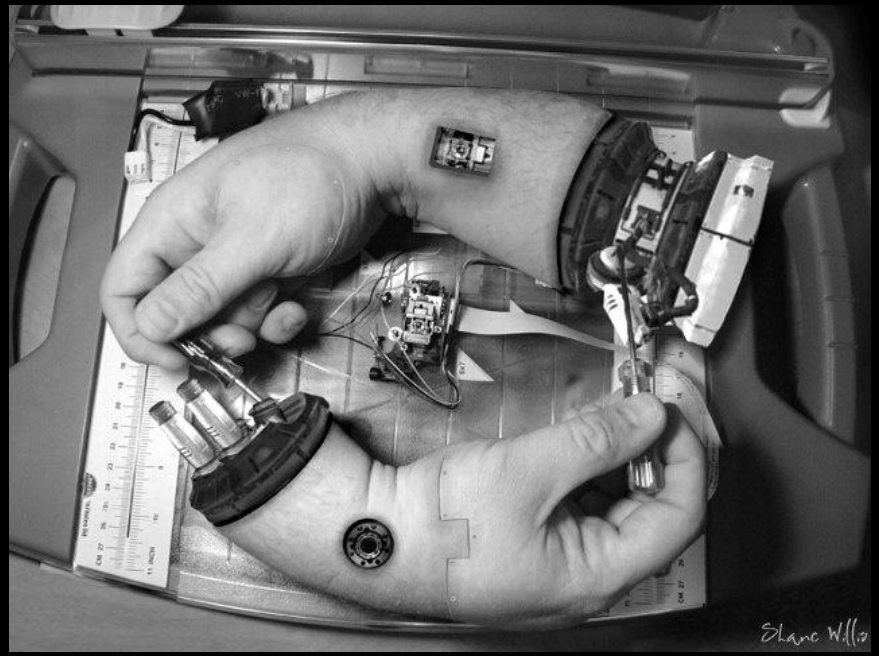
Detector Design Optimization



Cristiano Fanelli

Jefferson Lab

EIC^2 EIC Center at Jefferson Lab

Outline

Tue, Jan 12 (3h) - Thu, Jan 14 (1h)

Detector Design Optimization

Detector Design

- Introduction
  - Why need AI? SOTA.
- Single-objective Optimization:
  - Theory, real-world example
  - BO exercise: intro
- Multi-objective Optimization:
  - Theory, real-world example
  - MOO exercise: intro

Q/A

Hands-on

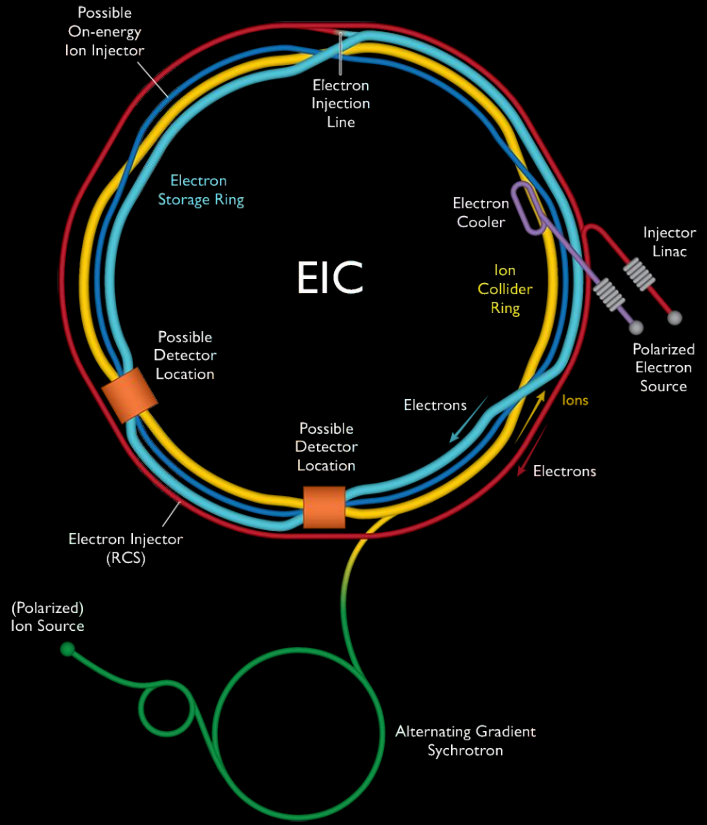
- exercises

Q/A

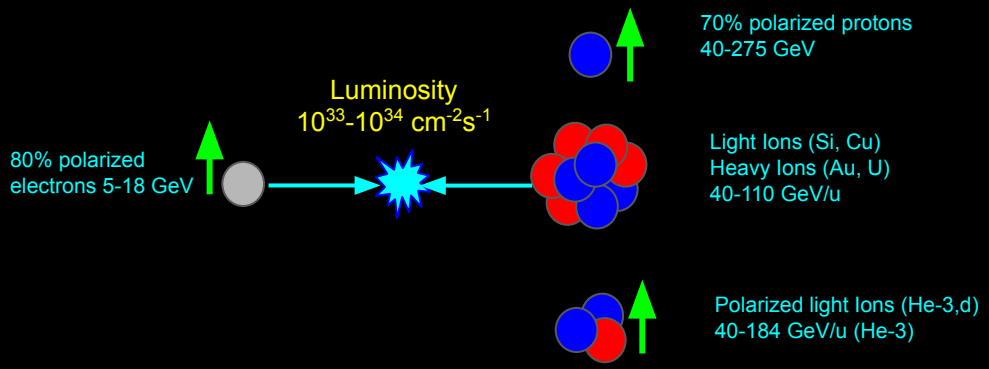
Detector Design: Introduction

- Particle detectors are essential tools to understand the universe and they play a crucial function in our society, from medical imaging to drug development, from dating ancient findings to testing materials properties.
- Fundamental nuclear and particle physics research often requires realizing expensive large-scale experiments combining multiple sub-detectors to investigate the building blocks of nature. According to the DOE, "*AI techniques that can optimize the design of complex, large-scale experiments have the potential to revolutionize the way experimental nuclear and particle physics is currently done*".
- Surprisingly at present few AI-based approaches (and generally procedural methods) for designing particle detectors have been explored/developed.
- More than 50 years have passed since Charpak (nobel prize in 1992) revolutionised particle detectors with the construction of a MWPC. Nowadays we have the unique opportunity to design complex detection systems with the support of AI.
- Using AI will allow to optimize large detectors in NP experiments like the Electron Ion Collider. EIC will be a flagship nuclear physics facility in the US that will be constructed over the next 10 years and it is currently at its design phase. Its R&D program can be one of the first to systematically leverage on AI.

The Electron Ion Collider example



Will be constructed over ten years at an estimated cost between $1.6 and $2.6 billion



- Two intersecting accelerators, one producing a beam of electrons, the other a high-energy beam of protons or heavier atomic nuclei
- Wide coverage of CoM energy $\sqrt{s}_{e\text{-}p}$ ~ (20-140) GeV
- Two large acceptance detectors

A machine for delving deeper than ever before into the building blocks of matter

The EIC scientific program in a nutshell

Emergence of Mass

- Nucleons: 99% of the mass of the visible universe
- How does the proton mass emerge from QCD, and why is it so heavy?
- What is the mechanical structure of the proton?

Nucleon Spin and Imaging

- Full map of nucleon spin structure and dynamics in momentum and position space
- Towards a comprehensive 5D picture of the quantum structure of the proton

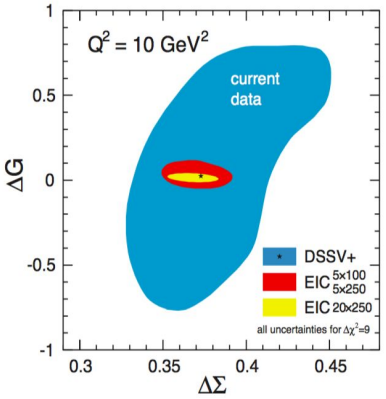
Physics with high-energy nuclear beams

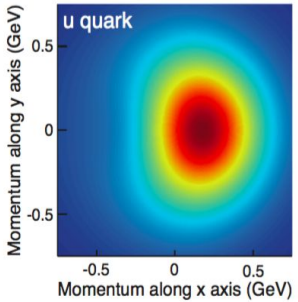
- Saturation: Is the high-energy/low-x limit governed by a universal dense saturated gluon matter?

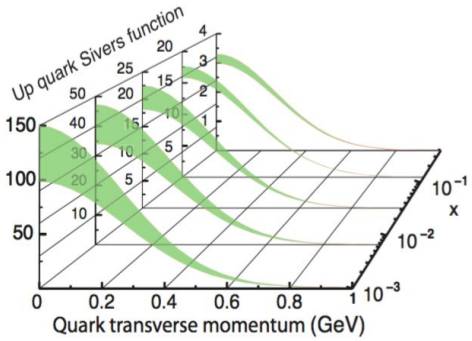
Other Topics

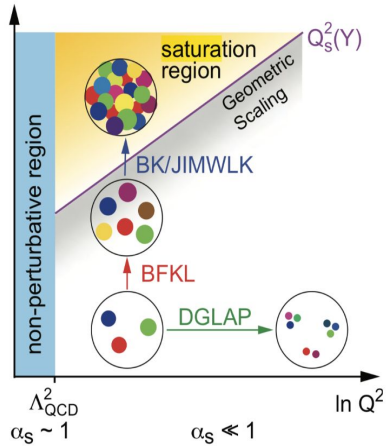
- Confinement, Hadronization
- Passage of color charge through cold QCD matter
- Jet Physics in ep/eA collisions
- BSM
- etc

Accardi, A. *et al* - BNL-98815-2012- JAJLAB-PHY-12-1652arXiv:1212.1701



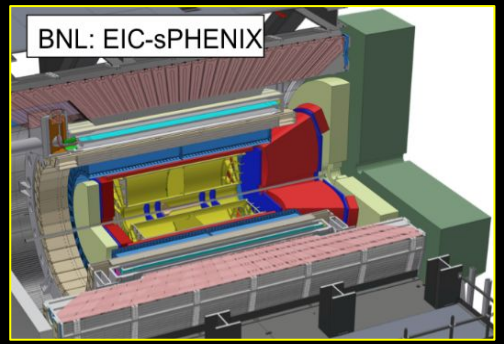


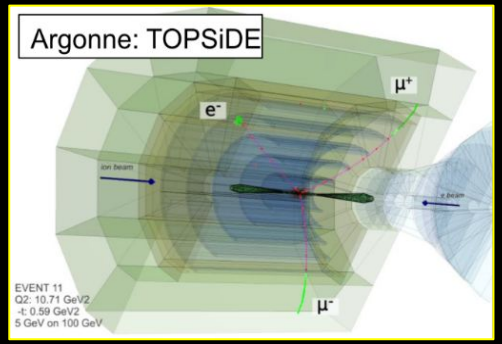


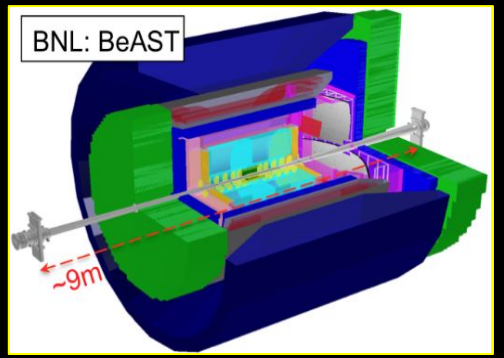


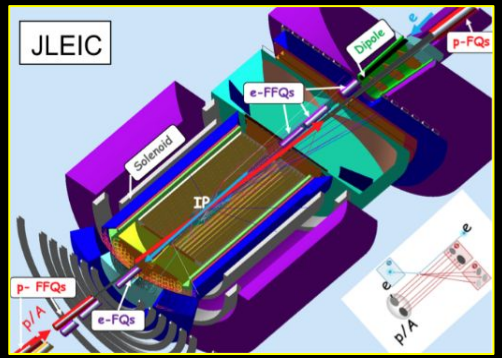
Design Phase: a “Handbook Detector” for EIC

At the beginning of the 2020 we had different concepts...

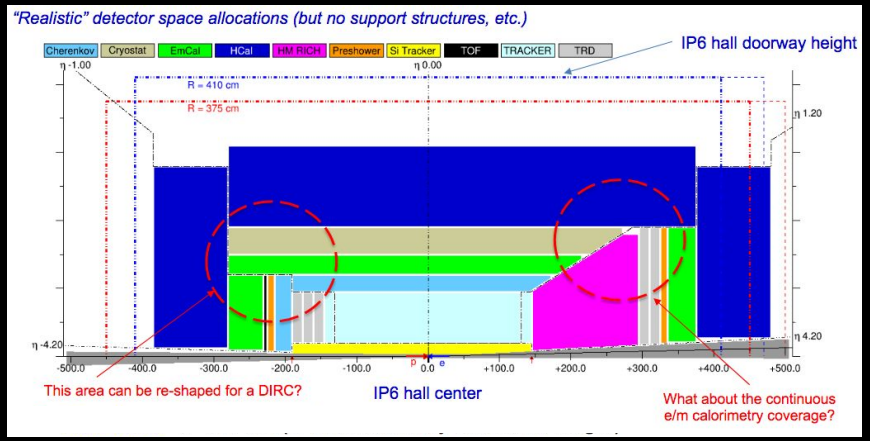








- After CD-0, the project rapidly evolved as the EIC community is working on the EIC Yellow Book to have a general purpose "Reference Detector” for BNL.
- See recent talks by A. Kiselev at the 4th Yellow Report workshop



How do we design and optimize detectors?

- Typically full detector designs are studied once the subsystem prototypes are ready, and in the subsystem design phase constraints from the full detector or outer layers are taken into consideration.

- Actually many parameters (mechanics, geometry, optics) characterize the design of each sub-detector, hence the full design represents a large combinatorial problem. A well known phenomenon observed in optimization problems with high-dimensional spaces is the so-called “curse of dimensionality'” [1], introduced for the first time by Bellman when considering problems in dynamic programming.

- In addition to that, more objective functions often need to be considered at the same time in the design of each sub-detector (e.g., dresolution, efficiency, cost, distinguishing power, etc).

- In this context, AI offers state-of-the-art (SOTA) solutions to solve complex optimization problems in an efficient way.

[1] Bellman, Richard. *Dynamic programming*. Vol. 295. RAND CORP SANTA MONICA CA, 1956.

Detector Design with AI

[1] A. Mosavi, T. Rabczuk, and A. R. Varkonyi-Koczy, “Reviewing the novel machine learning tools for materials design,” in Int. Conference on Global Research and Education, pp. 50–58, Springer, 2017
[2] Z. Zhou, S. Kearnes, L. Li, R. N. Zare, and P. Riley, “Optimization of molecules via deep reinforcement learning,” Scientific Reports, vol. 9, no. 1, pp. 1–10, 2019
[3] CF et al. "AI-optimized detector design for the future Electron-Ion Collider: the dual-radiator RICH case." *JINST* 15.05 (2020): P05009.
[4] Wolpert, D.H., Macready, W.G., 1997. No free lunch theorems for optimization. Trans. Evol. Comp 1, 67–82

- Designing detectors “with” AI is a new area of research at its infancy that can have a tremendous impact across many fields (NP, HEP, Astro-Phys). E.g., among the societal implications, the design of new detectors for Med-Phys with potentially improved performance at lower costs.
- It includes a broad range of approaches, from “optimizing” an existing expert-drawn baseline detector concept, to in principle letting AI design completely “new” and unseen configurations.
- New field, not many examples... we will cover the most significant in the present literature. Many applications in other fields in recent years, e.g., industrial material, molecular and drug design [1, 2].
- AI-driven design does not imply “interfacing” AI with existing advanced simulation platforms used in our community (Geant), but it also (and principally) entails establishing a procedural body of instructions to encode efficiently the optimal design requirements and validate the results in a self-consistent way [3].
- As far as optimization is concerned, the choice of a suitable algorithm is a challenge itself (no free lunch theorem [4]) and the full potential of certain algorithms always requires some degree of customization. First thing to do is to study and characterize the properties of the problem. In these lectures we will cover only some specific class of problems and algorithms.

Characterizing the Detector Design Problem

The detector design problem in NP physics (either collider or fixed target) experiments is typically characterized by:

1. A number of sub-detectors layers starting from the interaction point;

2. A relational 'hierarchy' (or coupling) among layers (e.g., the presence of material in front of a sub-detector can impact its performance);

3. Symmetry (e.g., hermetic detectors with large acceptance like EIC or SoLID have a 'cylindrical' geometry);

4. Modularity (e.g., repeated sub-elements within a sub-detector);

5. Constraints (e.g., volumes cannot overlap);

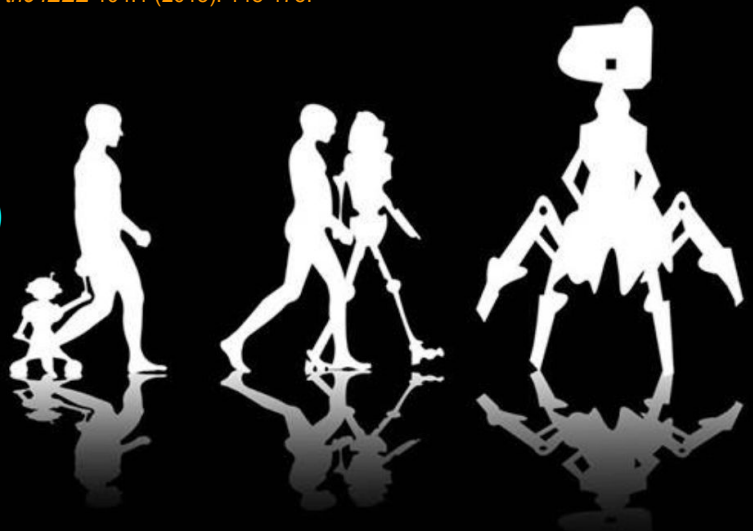
6. Heterogeneous simulations/evaluations (e.g., certain processes like developing showers in calorimeters take longer than others to simulate; point 6 actually encompasses different aspects in the pipeline, see later).

7. The detector response is typically noisy and detailed simulations can be compute expensive.

“Taking the Human out of the Loop”…?

B. Shahriari, et al. *Proceedings of the IEEE* 104.1 (2015): 148-175.

Human-assisted (experts knowledge)



Fully automated

Bayesian Optimization

DL-enhanced

Multi-objective

Evolutionary autoML

Meta-learning

Reinforcement Learning

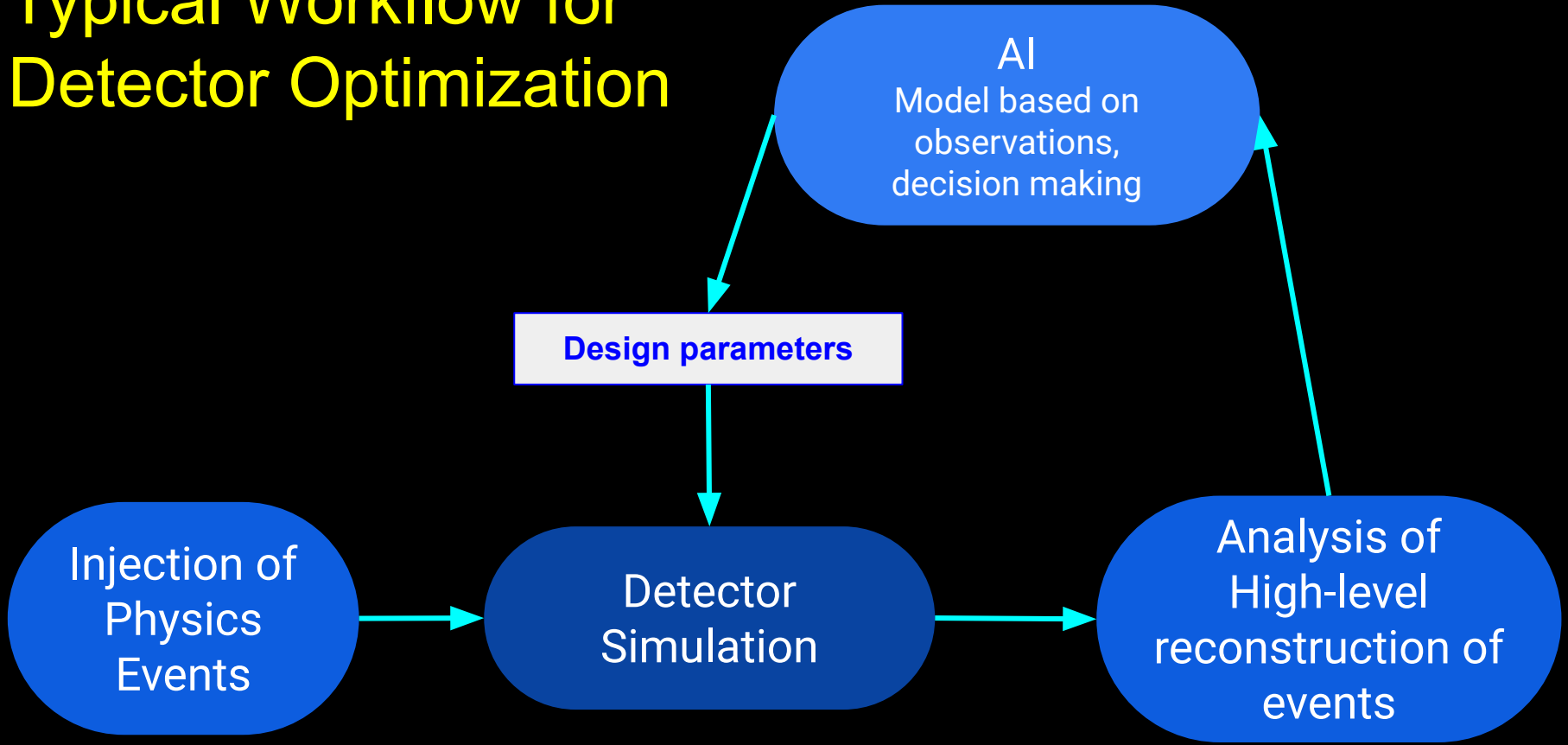
Accelerated Discovery

- *AI for Detector Design*
- *AI for Intelligent Detectors*

- Optimal Design
- Inverse Design
- Self-Design
- Calibration/ Alignment
- Self-Calibration
- etc.

In the following we will cover only some specific algorithm for detector design

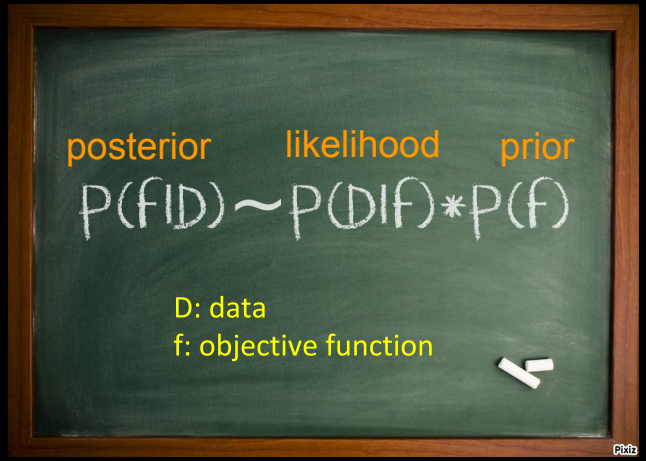
Typical Workflow for Detector Optimization



Bayesian Optimization

- Objective *f* is a black-box function and can be noisy.
- Evaluations are expensive making grid or exhaustive search impractical.
- *f* lacks of special structure (e.g. convex), and it has no gradient information.

If you don't have the above constraints,
do not use Bayesian Optimization

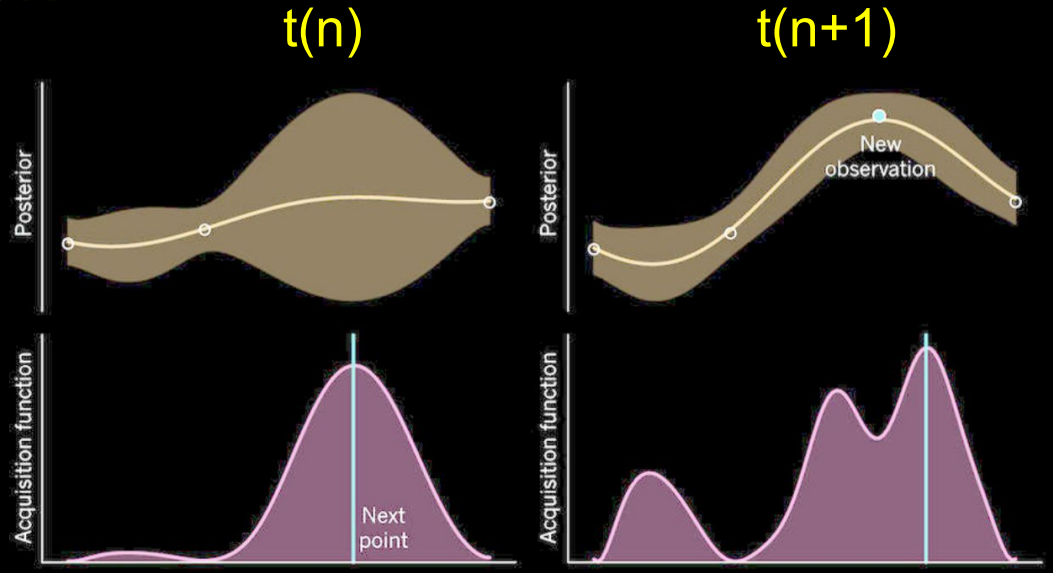


- We want to determine the optimum of *f*, no need to improve estimates of regions where *f* is not optimal. The idea is to build a surrogate function:
  - With a Prior over the space of objective functions, to model our black-box function.
  - Likelihood ~ probability of observing the data given the function *f*.
  - The Posterior probability is the surrogate objective function. It captures the updated beliefs about the unknown objective.

https://machinelearningmastery.com/what-is-bayesian-optimization/
http://krasserm.github.io/2018/03/21/bayesian-optimization/
http://krasserm.github.io/2018/03/19/gaussian-processes/

Bayesian Optimization

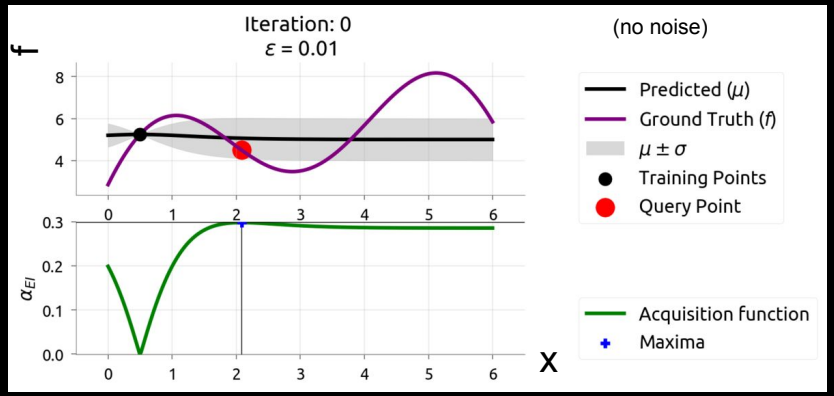
- BO is a sequential strategy developed for global optimization.
- After gathering evaluations we builds a posterior distribution used to construct an **acquisition function**.
- This cheap function determines what is **next query point**.



1. Select a Sample by Optimizing the Acquisition Function.
2. Evaluate the Sample With the Objective Function.
3. Update the Data and, in turn, the Surrogate Function.
4. Go To 1.

http://krasserm.github.io/2018/03/21/bayesian-optimization/
http://krasserm.github.io/2018/03/19/gaussian-processes/

Acquisition Functions



Exploitation **Exploration**

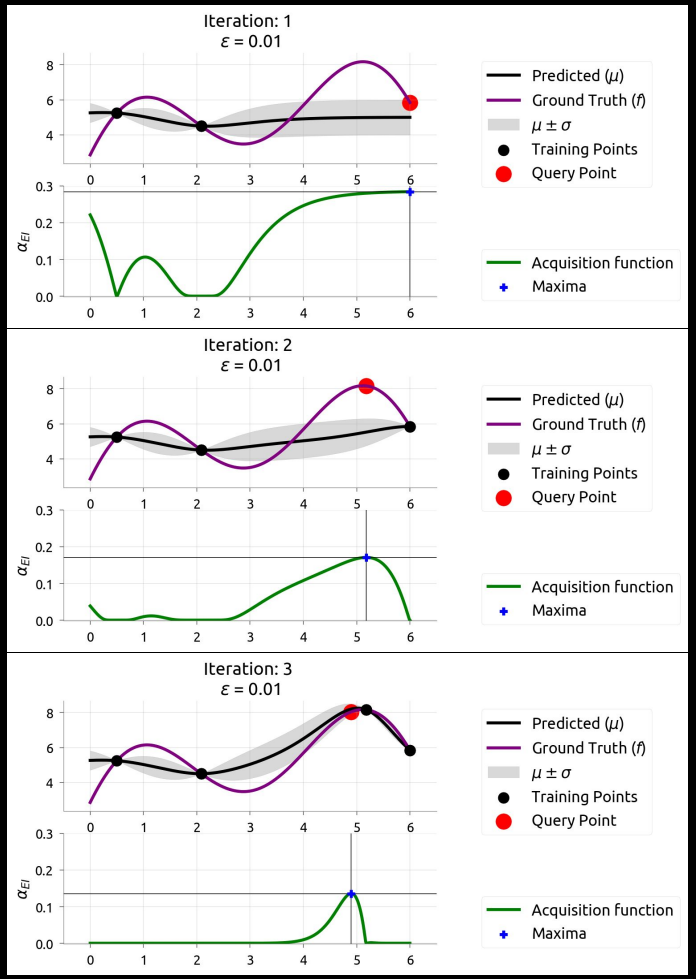
$$EI(x) = \begin{cases} (\mu_t(x) - f(x^+) - \epsilon)\Phi(Z) + \sigma_t(x)\phi(Z), & \text{if } \sigma_t(x) > 0 \\ 0, & \text{if } \sigma_t(x) = 0 \end{cases}$$

Best found so far

$$Z = \frac{\mu_t(x) - f(x^+) - \epsilon}{\sigma_t(x)}$$

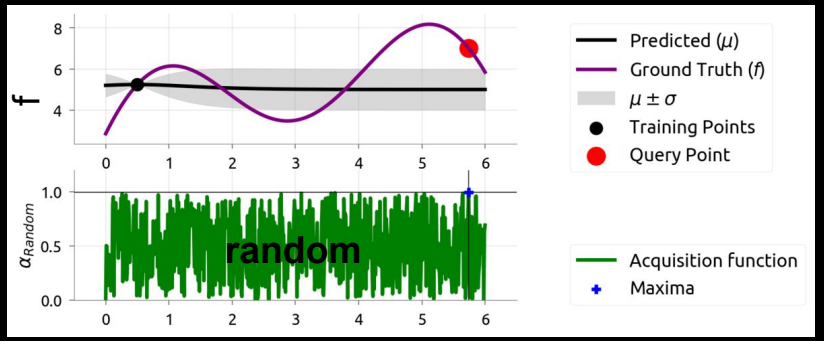
We are sampling x

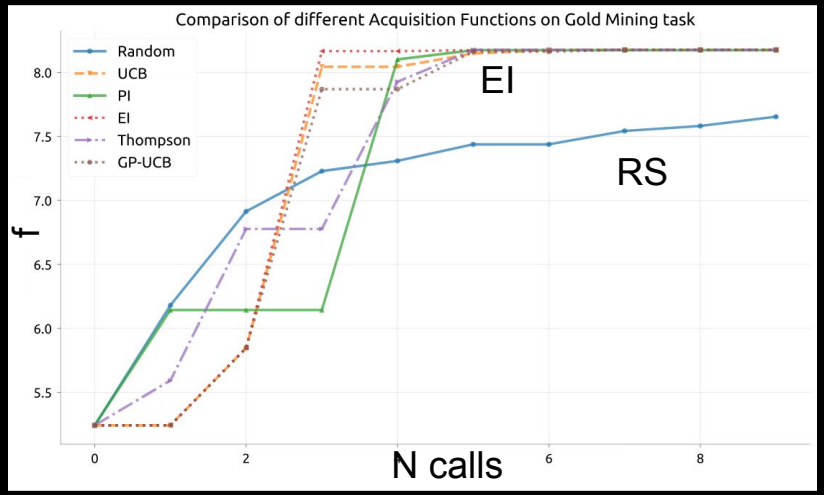
- **Exploitation**: search where μ is high
- **Exploration**: search where σ is high



Acquisition Functions

- Many acquisition functions, e.g., Probability of Improvement, Expected Improvement, Upper (Lower) confidence bound, etc
- In most cases, acquisition functions provide knobs for controlling the exploration-exploitation tradeoff
- When optimization is more complex (more dimensions), then a random acquisition might perform poorly



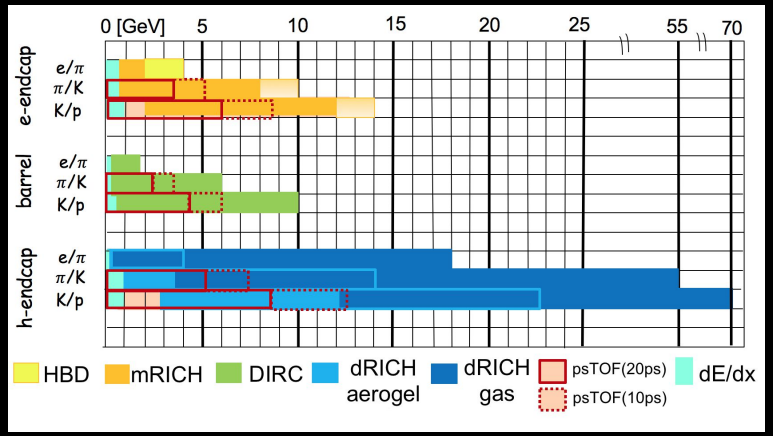


E. Brochu, Eric, V. M. Cora, and N. De Freitas. "A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning." *arXiv:1012.2599* (2010).

- https://distill.pub/2020/bayesian-optimization/
- https://distill.pub/2019/visual-exploration-gaussian-processes/

Dual RICH: case study

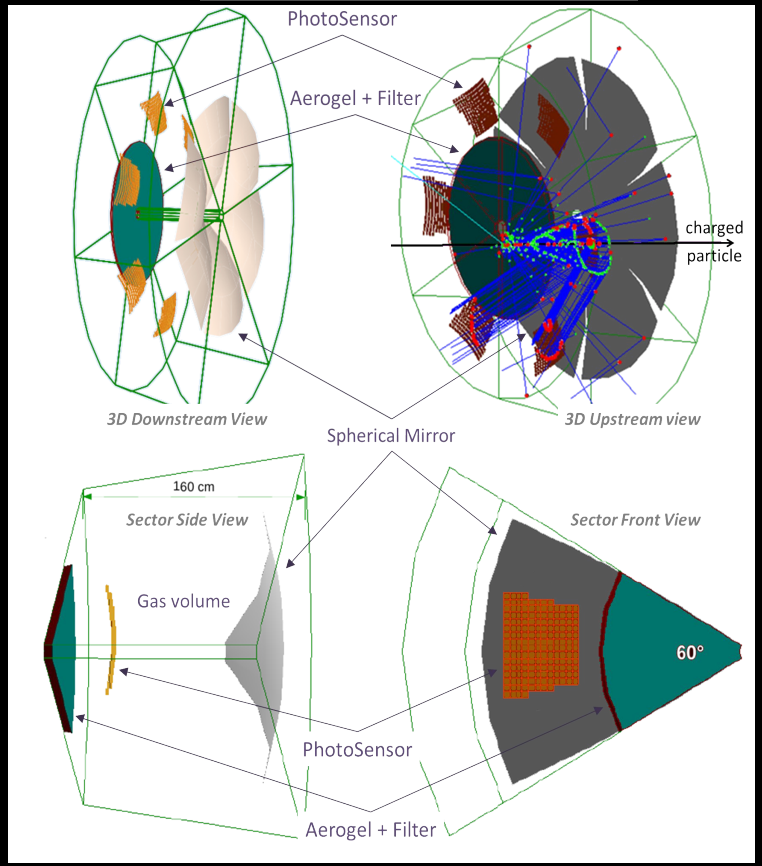
E. Cisbani, A. Del Dotto, CF*, M. Williams et al.
"AI-optimized detector design for the future Electron-Ion Collider: the dual-radiator RICH case."
Journal of Instrumentation 15.05 (2020): P05009.



- Continuous momentum coverage.
- Simple geometry and optics, cost effective.
- Legacy design from INFN, see EICUG2017

  - 6 Identical open sectors (petals)
  - Optical sensor elements:
    8500 cm^2/sector, 3 mm pixel
  - Large focusing mirror

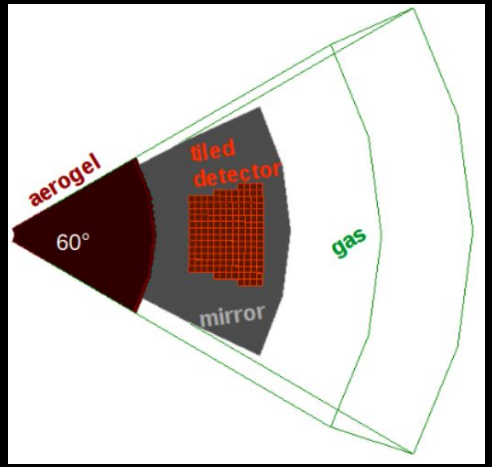
aerogel (4 cm, n(400 nm): 1.02)
+ 3 mm acrylic filter
+ gas (1.6 m, n(C_2F_6): 1.0008)



Construction Constraints on Design Parameters

The idea is that we have a bunch of parameters to optimize that characterize the detector design.
We know from previous studies their ranges and the construction tolerances.

parameter	**description**	**range** [units]	**tolerance** [units]
R	mirror radius	[290,300] [cm]	100 [μm]
pos r	radial position of mirror center	[125,140] [cm]	100 [μm]
pos l	longitudinal position of mirror center	[-305,-295] [cm]	100 [μm]
tiles x	shift along x of tiles center	[-5,5] [cm]	100 [μm]
tiles y	shift along y of tiles center	[-5,5] [cm]	100 [μm]
tiles z	shift along z of tiles center	[-105,-95] [cm]	100 [μm]
$n_{aerogel}$	aerogel refractive index	[1.015,1.030]	0.2%
$t_{aerogel}$	aerogel thickness	[3.0,6.0] [cm]	1 [mm]



Ranges depend mainly on mechanical constraints and optics requirements.
These requirements can change in the next future based on inputs from prototyping.

Choice of Figure of Merit

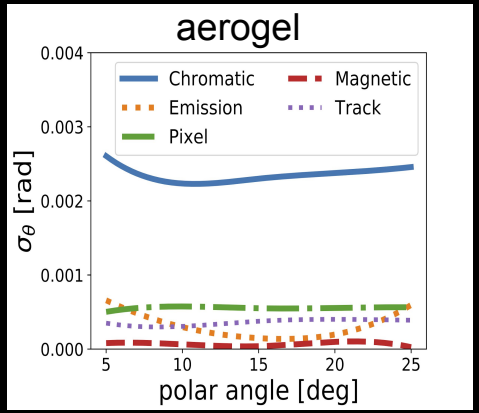
Goal is improve the distinguishing power of pions/kaons, hence:

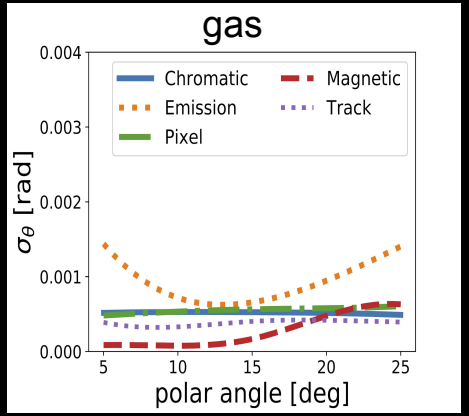
$$N\sigma = \frac{||\langle\theta_K\rangle - \langle\theta_\pi\rangle||\sqrt{N_\gamma}}{\sigma_\theta^{1p.e.}}$$

$$N_\gamma = (N_\gamma^\pi + N_\gamma^K)/2$$

$$h = 2 \cdot \left[\frac{1}{(N\sigma)|_1} + \frac{1}{(N\sigma)|_2}\right]^{-1}$$

Main contributions to resolution





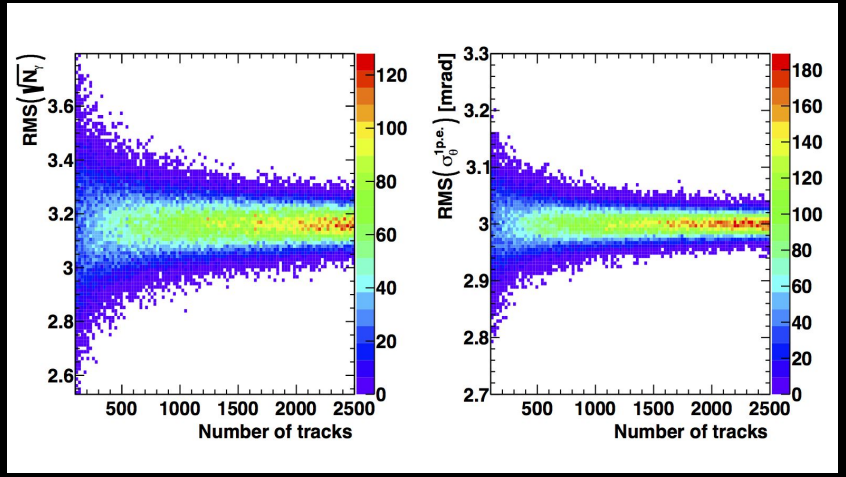
Remember that we do not have an explicit form of the FOM we are trying to optimize as a function of the design parameters

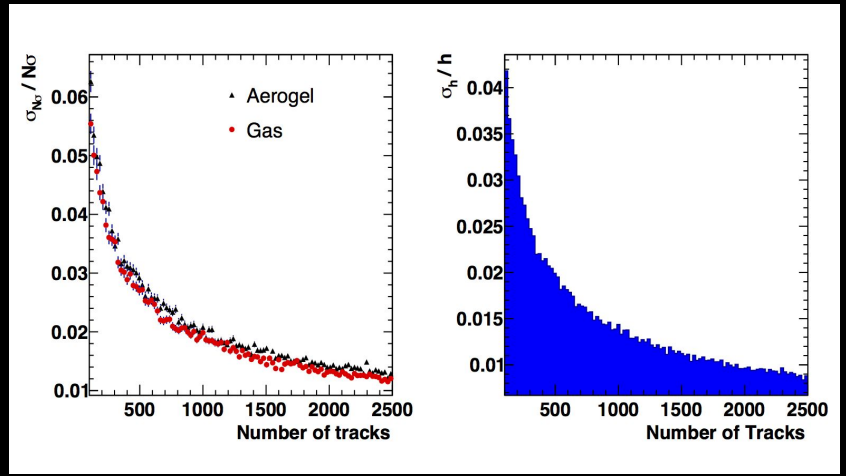
@ p_1 = 14 GeV/c (aerogel) and p_2 = 60 GeV/c (gas) considering the two parts disentangled

Noise Studies

$$N\sigma = \frac{||\langle\theta_K\rangle - \langle\theta_\pi\rangle||\sqrt{N_\gamma}}{\sigma_\theta^{1p.e.}}$$

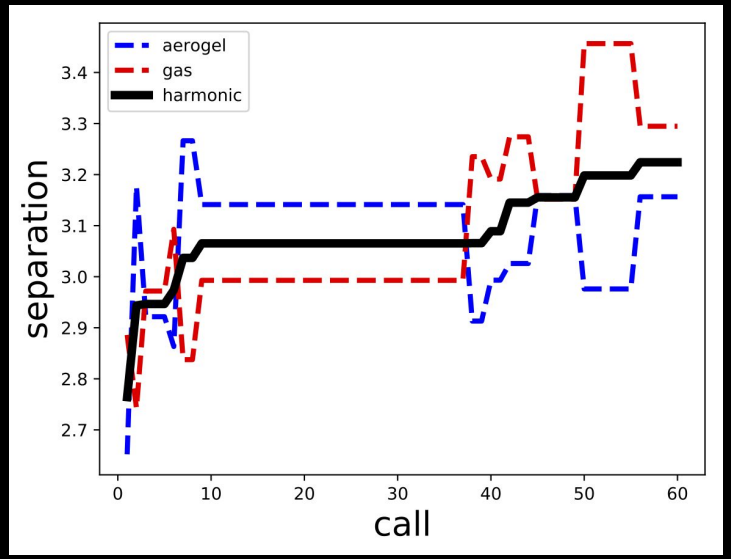
- Dedicated studies to characterize the noise as this is an optimization of a noisy function
- We choose N tracks = 400 based on the studies on noise to minimize as much as possible computing time during simulation.

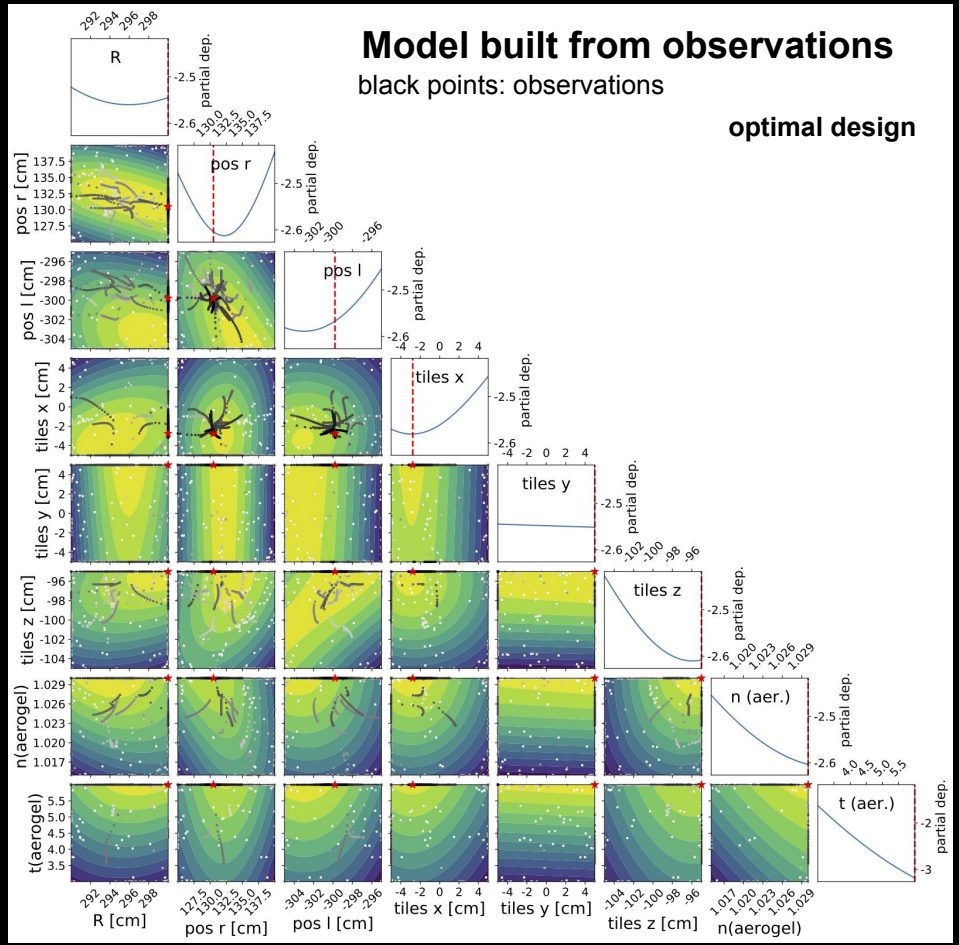




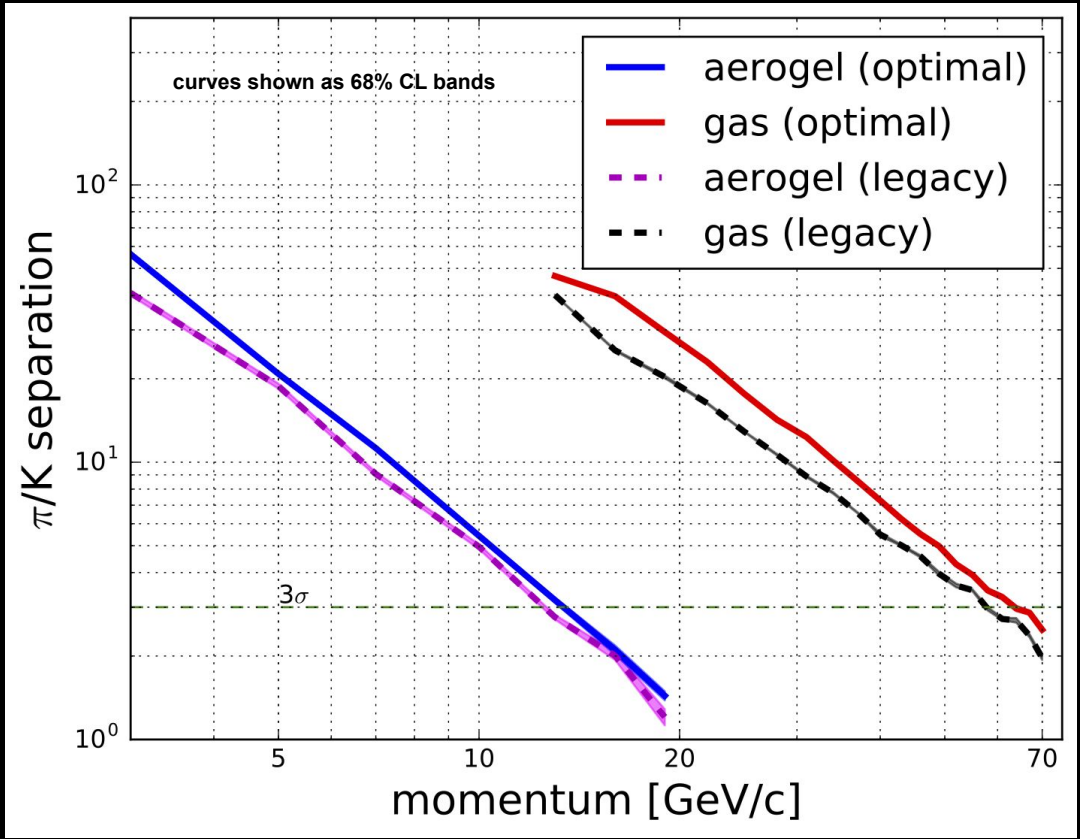
The Model and the Optimized FoM

$$N\sigma = \frac{||\langle\theta_K\rangle - \langle\theta_\pi\rangle||\sqrt{N_\gamma}}{\sigma_\theta^{1p.e.}}$$





dRICH Performance @ the optimal design point

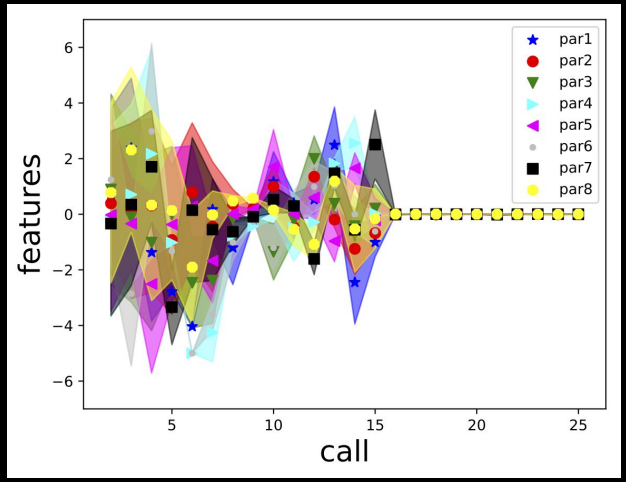


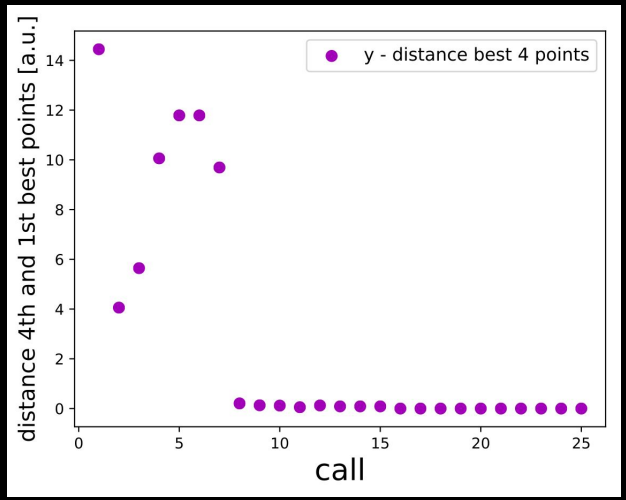
- Statistically significant Improvement in both parts.
- In particular in the gas region where the 5σ threshold shifted from 43 to 50 GeV/c and the 3σ one extended up to
- Notice that before this study we did not know “how well” the legacy design was performing.

E. Cisbani, A. Del Dotto, CF*, M. Williams et al.
JINST 15.05 (2020): P05009.

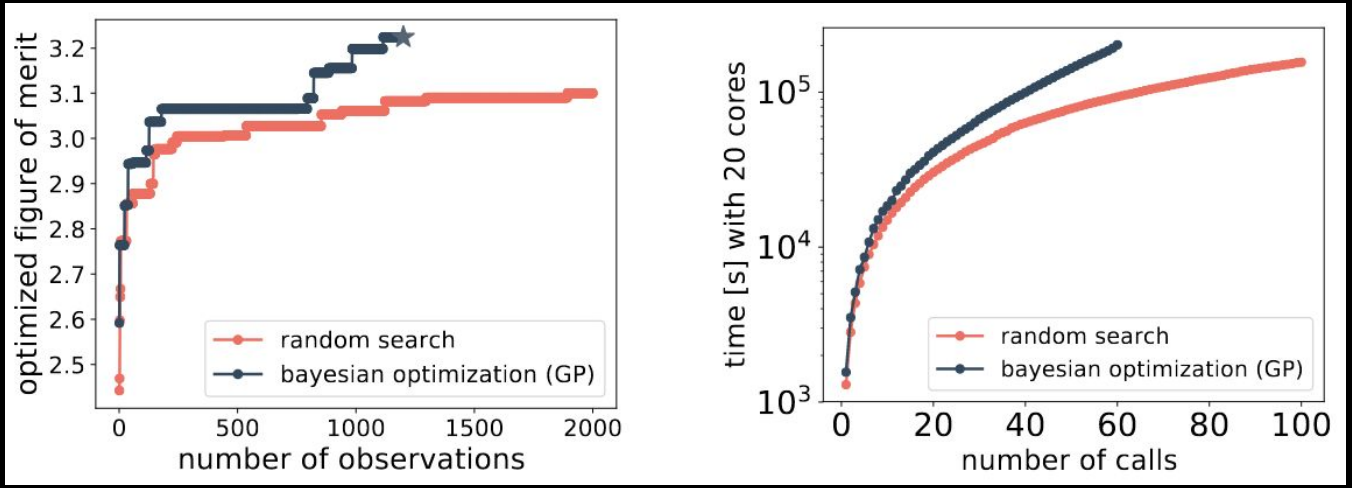
Convergence Criteria

- Can in general be applied in the design space, in the objective space, or looking at the behavior of the acquisition function.
- We defined a set of conditions to ensure convergence:
  - These correspond to the logic AND of booleans on each feature and on the variation of the figure of merit.
  - They are built on standardized Z and Fisher statistics.
- Pre-processing of data required to remove outliers.





Comparison with Random Search



Each call:
400 tracks generated/core
20 cores

1 design point ~ 10 mins/CPU

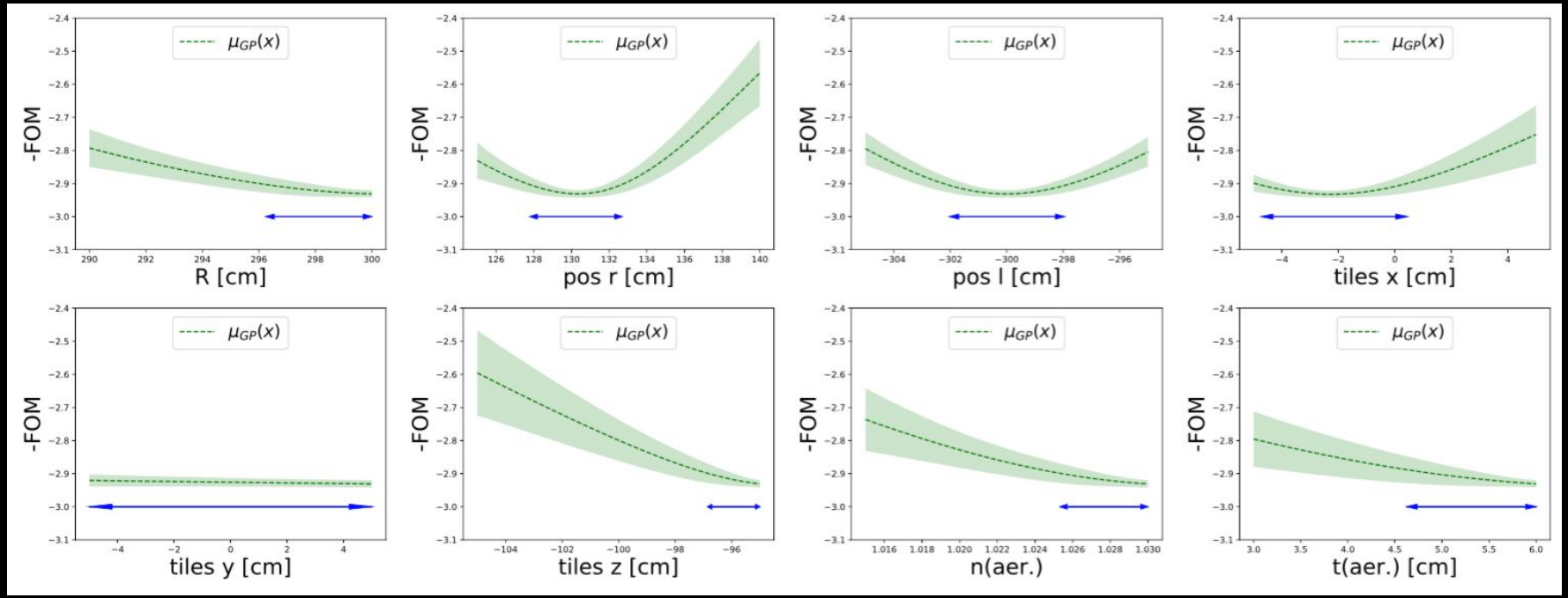
Budget: 100 calls

- BO with GP scales cubically with number of observations.
- Bayesian optimization methods are more promising because they offer principled approaches to weighting the importance of each dimension.
- For this 8D problem - even with 50 cores, RS looks unfeasible due to the curse of dimensionality.
  - Recall that the probability of finding the target with RS is $1-(1-v/V)^T$, where T is trials, v/V is the volume of target relative to the unit hypercube

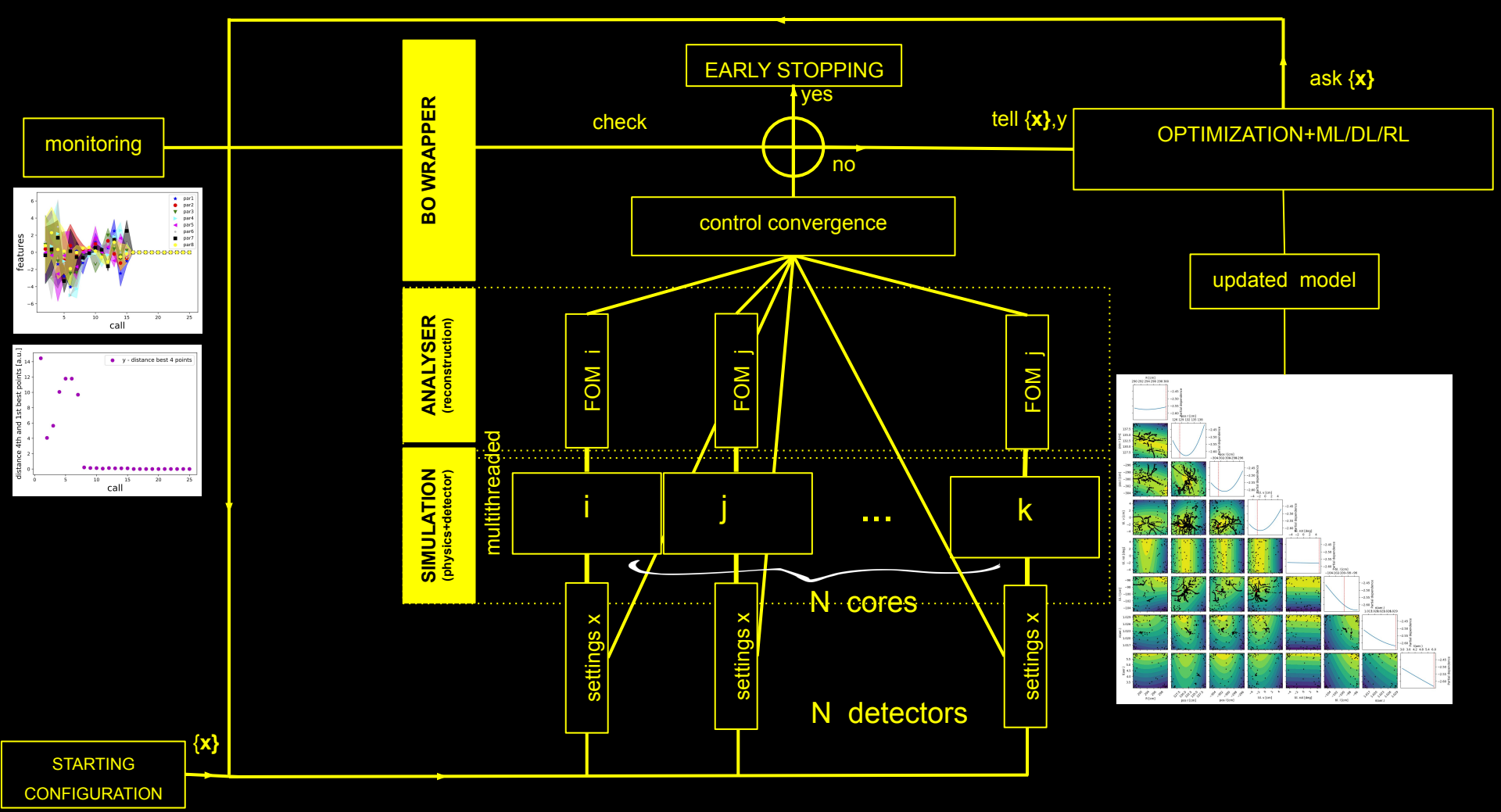
Bergstra, Bengio, “Random search for hyper-parameter optimization”, J. Mach. Learn. Res.13 (Feb) (2012) 281–305.

Tolerance Regions

- BO provides a model of how the FoM depends on the parameters, hence it is possible to use the posterior to define a tolerance on the parameters (regions ensuring improved PID, see previous slide).



- Larger than the construction tolerances on each parameter.
  Notice a small lateral shift of the tiles has negligible impact on the PID capability.


monitoring
BO WRAPPER
EARLY STOPPING
yes
check
tell {x},y
ask {x}
OPTIMIZATION+ML/DL/RL
no
control convergence
updated model
ANALYSER
(reconstruction)
SIMULATION
(physics+detector)
multithreaded
FOM i
FOM j
FOM j
i
j
...
k
N cores
settings x
settings x
settings x
N detectors
{x}
STARTING
CONFIGURATION
features
call
distance 4th and 1st best points [a.u.]
y - distance best 4 points
call

Frameworks and Deployment in the Industry

- scikit-optimize
- sigopt
- hyperopt
- spearmint
- MOE
- BOTorch
- GPFlowOpt
- GPyOpt
- DragonFly
- Hyperband
- Smac
- etc

- Bayesian Optimization has been applied to Optimal Sensor Set selection for predictive accuracy.
- Uber uses Bayesian Optimization for tuning algorithms via backtesting.
- Facebook uses Bayesian Optimization for A/B testing.
- Netflix and Yelp use Metrics Optimization software like Metrics Optimization Engine (MOE) which take advantage of Parallel Bayesian Optimization.

Detector Toy Model

repl.it

cfanelli2 / driver

Stop ■ Share Upgrade +

Files

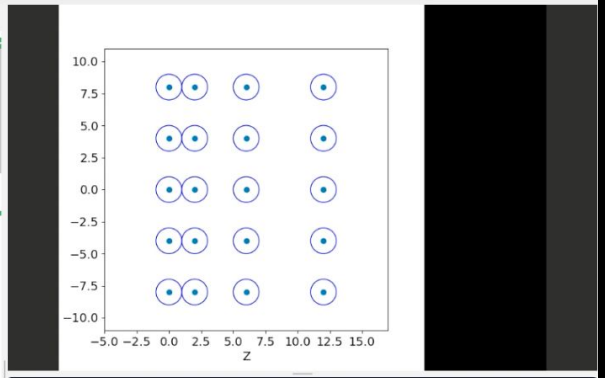
- main.py
- pic
- detector.py

Packager files

- poetry.lock
- pyproject.toml

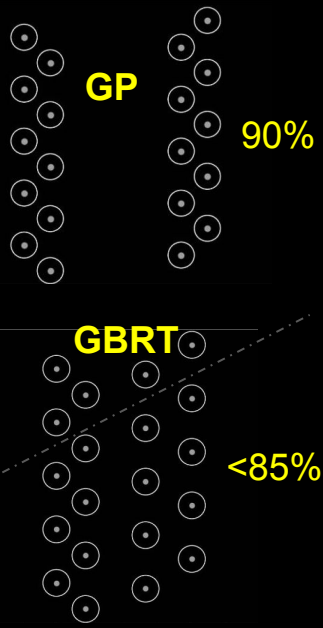
main.py

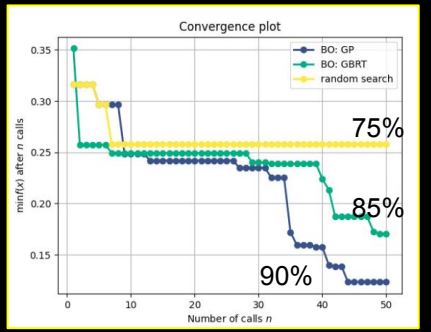
```
import detector

rand_st = np.random.randint(1,10000)
rand_st = 1317 #for reproducibility

# CONSTANT PARAMETERS
R = 1.  # cm
pitch = 4.0  #cm
ncalls = 10

# ADJUSTABLE PARAMETERS
y1 = 0.0
y2 = 0.0
y3 = 0.0
z1 = 2.0
z2 = 4.0
z3 = 6.0

#-------------- GEOMETRY ----------------#
print(".....INITIAL GEOMETRY")
tr = detector.Tracker(R, pitch, y1, y2, y3, z1, z2, z3)
Z, Y = tr.create_geometry()

detector.geometry_display(Z, Y, R, y_min=-10, y_max=10,block=False,pause=5)

N_tracks = 1500
t = detector.Tracks(b_min=-100, b_max=100, alpha_mean=0, alpha_std=0.2)
tracks = t.generate(N_tracks)

detector.geometry_display(Z, Y, R, y_min=-10, y_max=10,block=False, pause=-1)
detector.tracks_display(tracks, Z,block=False,pause=5)

score = detector.get_score(Z, Y, tracks, R)
print("fraction of tracks detected: ",score)


#-------------- OPTIMIZATION OF GEOMETRY ----------------#
print(".....OPTIMIZATION OF GEOMETRY")


def objective(x):
```



https://driver.cfanelli2.repl.run

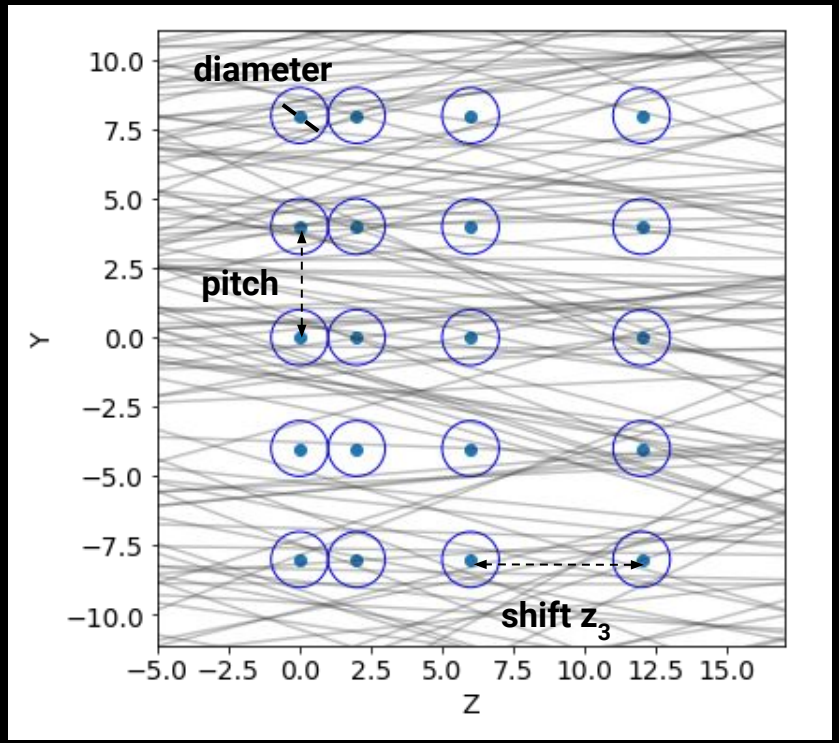
```
.....INITIAL GEOMETRY
```



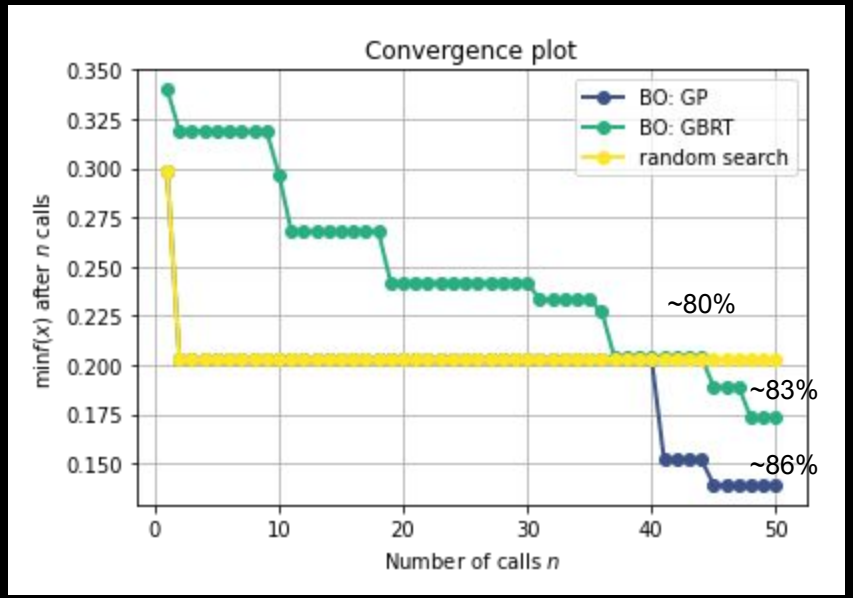


The Toy Model and Hands-on Sessions

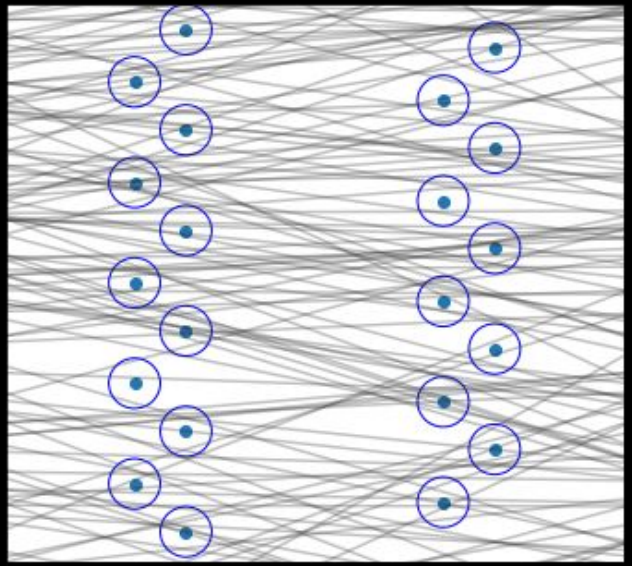
- A simple toy model is used in the hands-on session with the goal to introduce some optimization technique and useful frameworks in python.
- The toy detector consists in a 2D tracking system with 4 layers of wires.
- A total of 8 parameters can be tuned. The adjustable parameters are the radius of each wire, the pitch (along the y axis), and the shift along y and z of a plane with respect to the previous one. We will start with tuning the spatial shifts first (6 parameters).
- Straight tracks are generated at different angles and random origin. The tracker geometry and the track generation is already defined in the imported module *detector.py*.

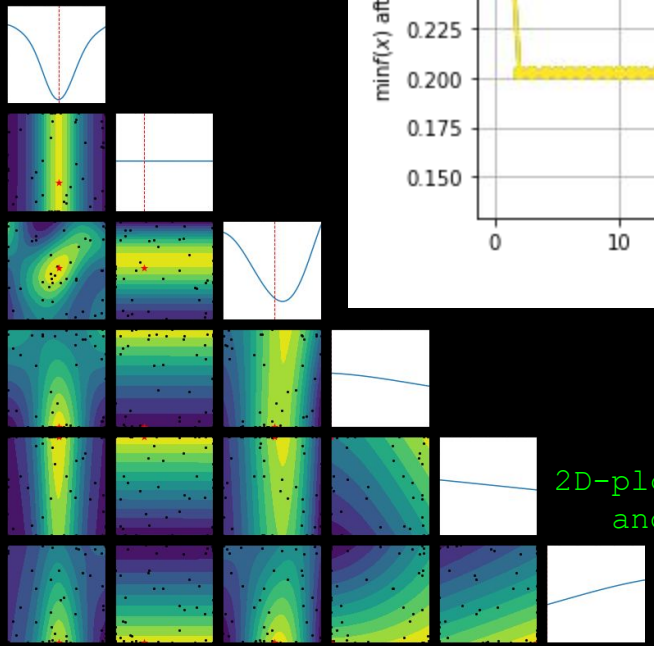


Toy Model: BO



"Optimal" configuration

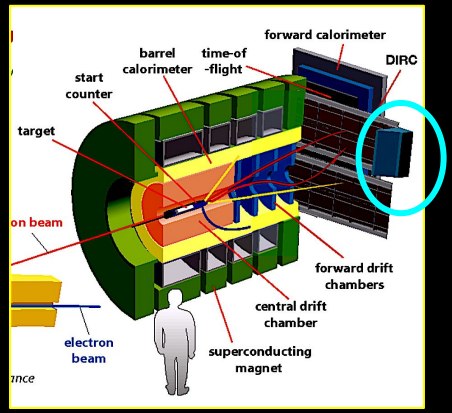




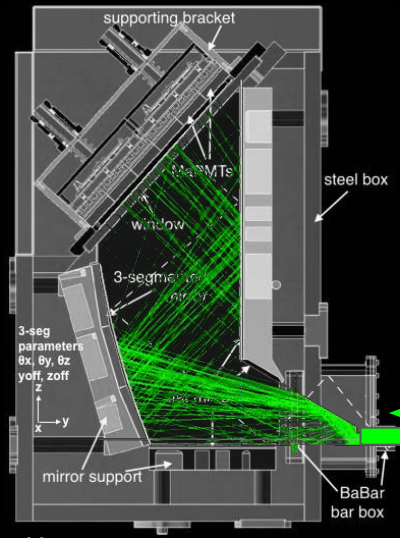
2D-plots of objective function and partial dependencies

Objective: Efficiency is defined as at least two wires are hit

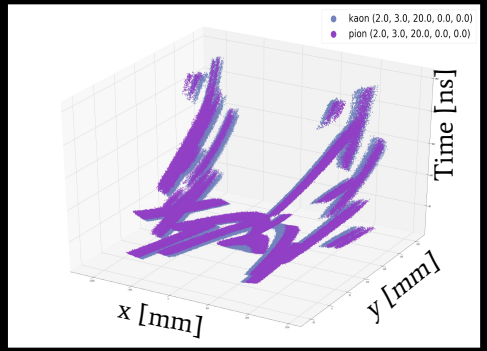
GlueX DIRC Alignment



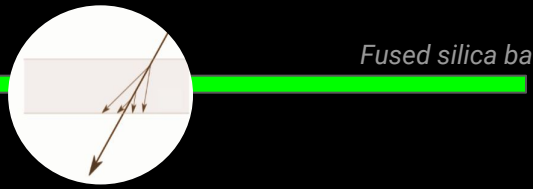
GlueX View

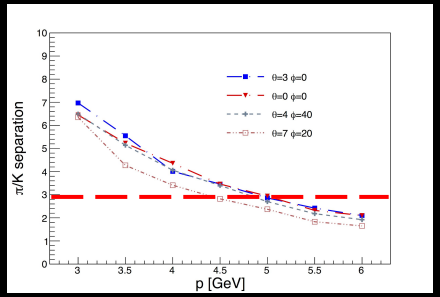


Optical box



3D Readout





π/K separation with DIRC

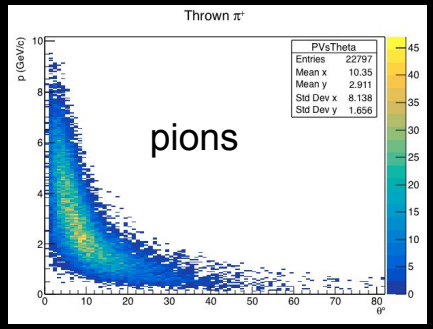
3D (x,y,t) readout allows to separate spatial overlaps.

Patterns take up significant fractions of the PMT in x,y and are read out over 50-100 ns due to propagation time in bars.

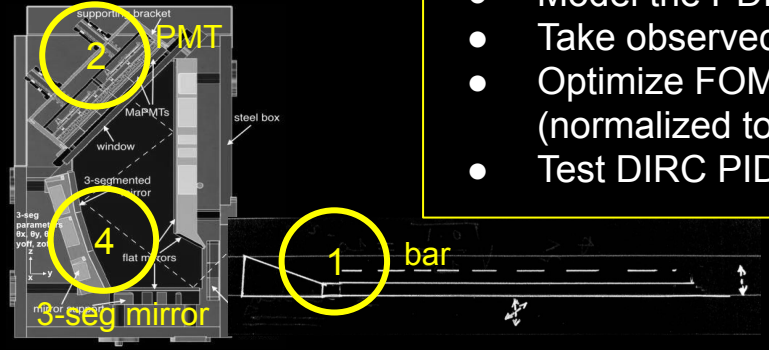
H12700 PMTs have a time resolution of O(200 ps) and read-out electronics giving time information in 1 ns buckets.

Approach

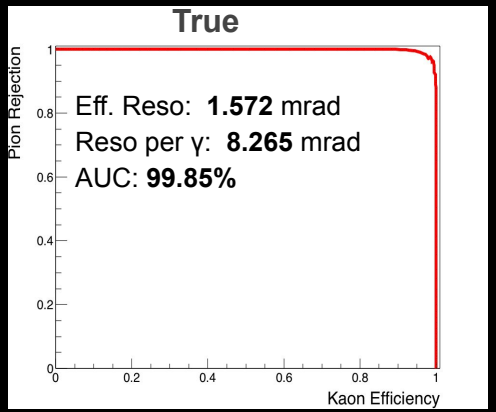
- Select high purity sample of particles at low P (well identified by GlueX PID w/o DIRC)
- Model the PDF as a function of the offsets
- Take observed hits to build Likelihood
- Optimize FOM = logL (normalized to a default alignment)
- Test DIRC PID on larger momentum P

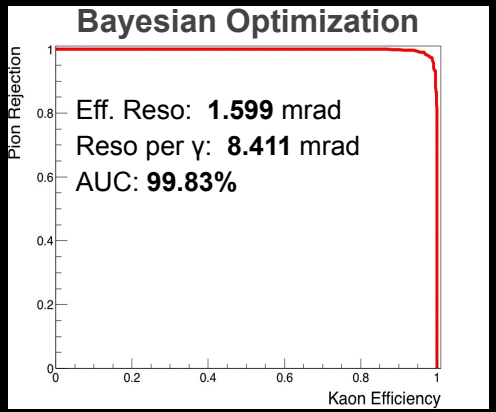


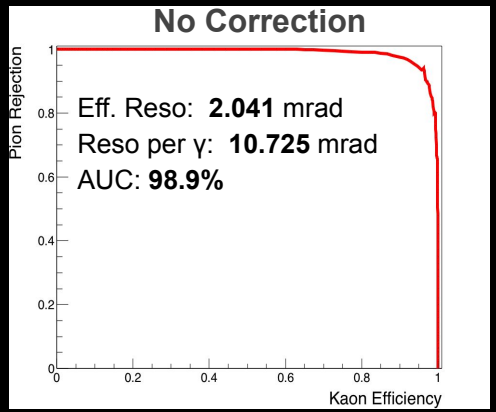
Main alignment parameters



Pion rejection vs Kaon efficiency at large P







True:

3-seg mirror:
θx,θy,θz=(0.25,0.50,0.15) deg,
y = 0.50 mm;
bar: z = 2.00 mm;
PMT: (r,θ)=(1.50 mm,1.00 deg)

BO-reversed engineered:

3-seg mirror:
θx,θy,θz=(0.25, 0.58, 0.12) deg,
y = 0.59 mm;
bar: z = 2.08 mm;
PMT: (r,θ)=(1.87 mm, 1.35 deg)

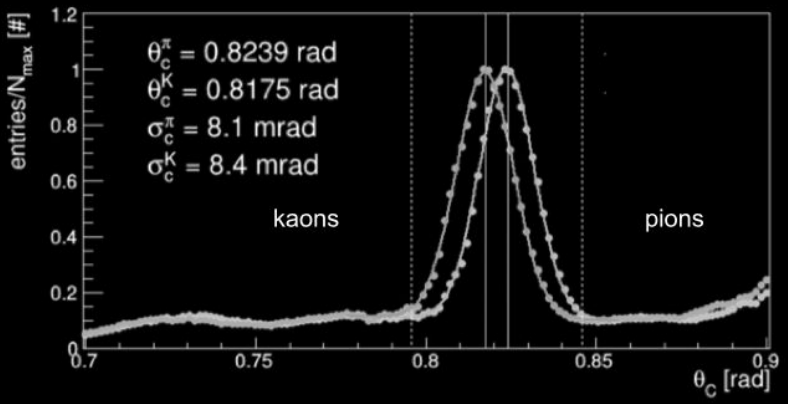
Reconstruction Algorithms and PID

LUT-based geometrical

R. Dzhygadlo et al. Nucl. Instr. And Meth. A, 766:263 (2014)

1. Creation of the LUT: store directions at the end of the radiator for each hit pixel

2. Direction from the LUT for the hit pixels are combined with the track directions (from tracking)

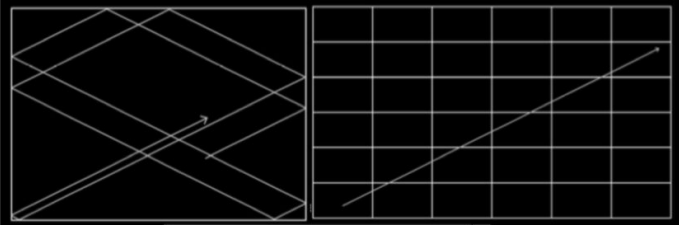


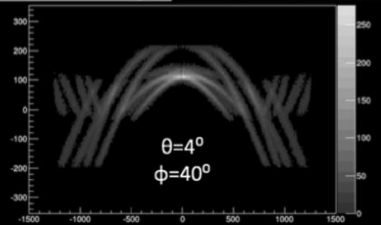
KDE-based

J. Hardin and M. Williams, JINST 11.10 (2016)

Fast tracing mapping straight lines through a tiled plane

1. Generation - 2. Traces through bars - 3. Traces through expansion volume





$$P(x) \approx \sum_{i}^{n} K(x - s_i)$$

https://github.com/jmhardin/FasDIRC

basically a trade-off memory/CPU usage

faster reconstruction/hit pattern

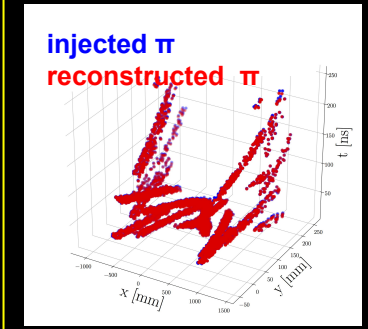
better resolution in regions with high overlap

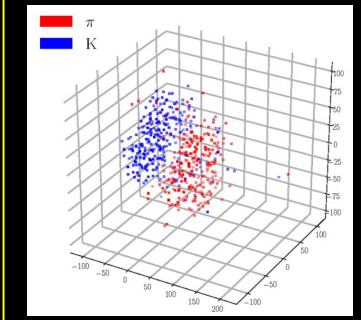
Hyperparameters tuning: DeepRICH

CF and J. Pomponi
Machine Learning: Science and Technology 1.1 (2020): 015010

DeepRICH

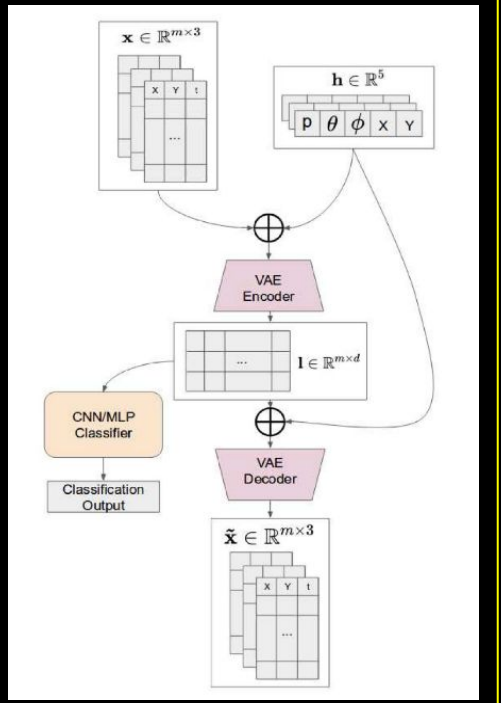
injected π
reconstructed π





latent space
t-SNE used for 3D visualization

injected



reconstructed

Hyperparameters

Table 2. List of hyperparameters tuned by the BO. The tuned values are shown in the outermost right column. The optimized test score is about 92%.

symbol	description	range	optimal value
NLL	λ_r	$[10^{-1},10^{2}]$	0.784
CE	λ_c	$[10^{-1},10]$	1.403
MMD	λ_v	$[1,10^{3}]$	1.009
LATENT_DIM	latent variables dimension	[10,200]	16
var_MMD	σ in $\mathcal{N}(0,\sigma)$	[0.01,2]	0.646
Learning Rate	learning rate	[0.000 1, 1]	$6.6{\cdot}10^{-4}$

DeepRICH Performance

Table 3. The area under curve (%), the signal efficiency to detect pions ε_S and the background rejection of kaons ε_B corresponding to the point of the ROC that maximizes the product $\varepsilon_S{*}\varepsilon_B$. The corresponding momenta at which these values have been calculated are also reported. This table is obtained by integrating over all the other kinematic parameters (i.e. a total of ∼6k points with different θ, ϕ, X, Y for each momentum).

	DeepRICH			FastDIRC		
Kinematics	AUC	ε_S	ε_B	AUC	ε_S	ε_B
4 GeV/c	99.74	98.18	98.16	99.88	98.98	98.85
4.5 GeV/c	98.78	95.21	95.21	99.22	96.33	96.32
5 GeV/c	96.64	91.13	91.23	97.41	92.40	92.47

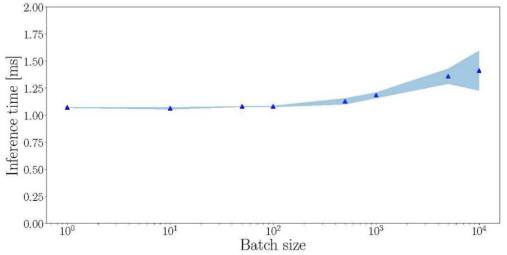


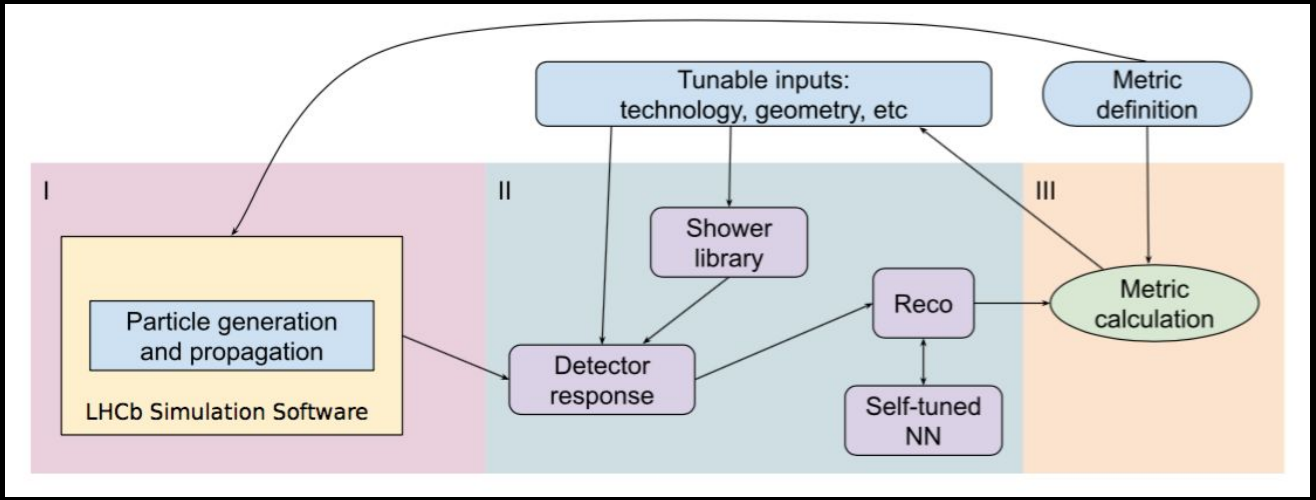
Figure 9. After training, the inference time is almost constant as a function of the batch size, meaning that the effective inference time—*i.e.*, the reconstruction time per particle—can be lower than a μs, the architecture being able to handle 10^4 particles in about 1.4 *ms* in the inference phase. Notice that the corresponding memory size in the inference phase is approximately equal to the value reported in table 4.

Other Applications

ML-assisted approach to calorimeter R&D

- Advanced detector R&D for both new and ongoing experiments in HEP requires per- forming computationally intensive and detailed simulations as part of the detector-design optimisation process.
- ML can substitute the most computationally intensive steps while retaining the GEANT4 accuracy to details.
- In [1] they focus on the Phase II Upgrade of the LHCb Calorimeter under the requirements on operation at high luminosity. The optimization pipeline looks like the following:

[1] A. Boldyrev et al (Yandex), arXiv:2005.07700v1 [physics.ins-det] 2020

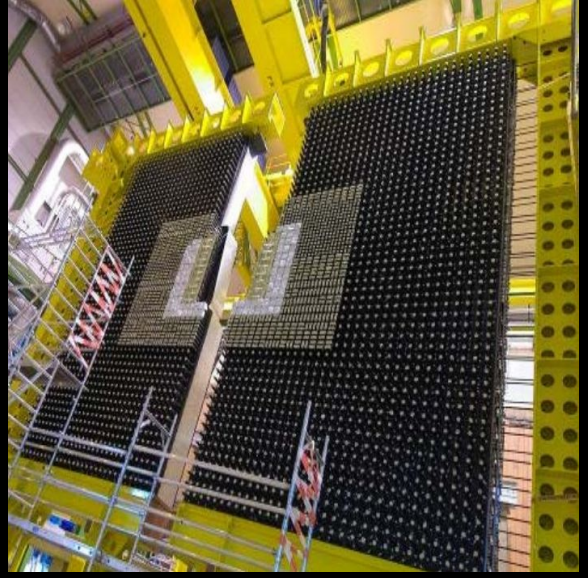


- Notice similarities with workflows discussed before.
- BO can be used to self-tune parameters.
- ML (decision trees) intervenes in tuning the reconstruction.

ECAL LHCb

$$\frac{\sigma_E}{E} = \frac{(8 \div 10)\%}{\sqrt{E(GeV)}} \oplus 0.9\%$$

- 4 mm thick scintillator tiles and 2 mm thick lead plates (Shashlik technology) ~25 X_0 (1.1 λ_I); Moliere radius ~ 36 mm;
- Modules 121.2 x 121.2 mm^2, 66 Pb +67 scintillator tiles;
- Segmentation: 3 zones 3 module types, Inner (9 cells per module), Middle (4), Outer (1). Total of 3312 modules, 6016 cells, (7.7 x 6.3) m^2, ~100 tons.



- Take advantage of segmentation / modularity (see discussion on characterization of the detector design problem) and create a Geant4 standalone simulation for 30x30 cells of size 20.2x20.2 mm^2 which can be rearranged in the inner, middle and outer ECAL modules.
- Used a signal sample $B^0_s \rightarrow J/\psi(\mu^+\mu^-)\pi^0(\gamma\gamma)$ and the LHCb minimum bias sample as background.
- Studies as a function of pile-up (PU) and number of primary vertices (nPV).
- Calibration (spatial and energy) optimized using XGBoost and BO for fine-tuning of parameters at the simulation and reconstruction steps.