International
MUON Collider
Collaboration

SAPIENZA
UNIVERSITÀ DI ROMA

Machine Learning approach to shield optimization at Muon Collider

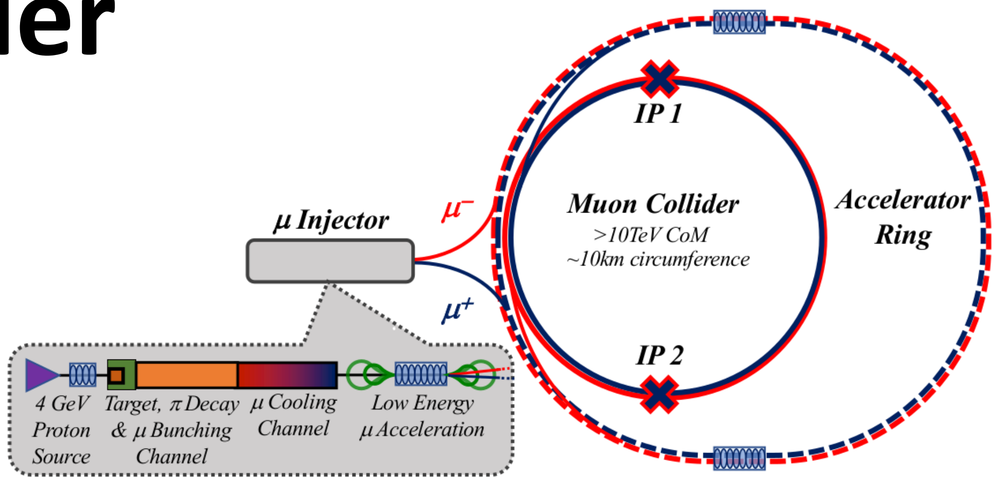
D. Calzolari , **L. Castelli***, F. Collamati, A. Lechner, D. Lucchesi

**speaker*

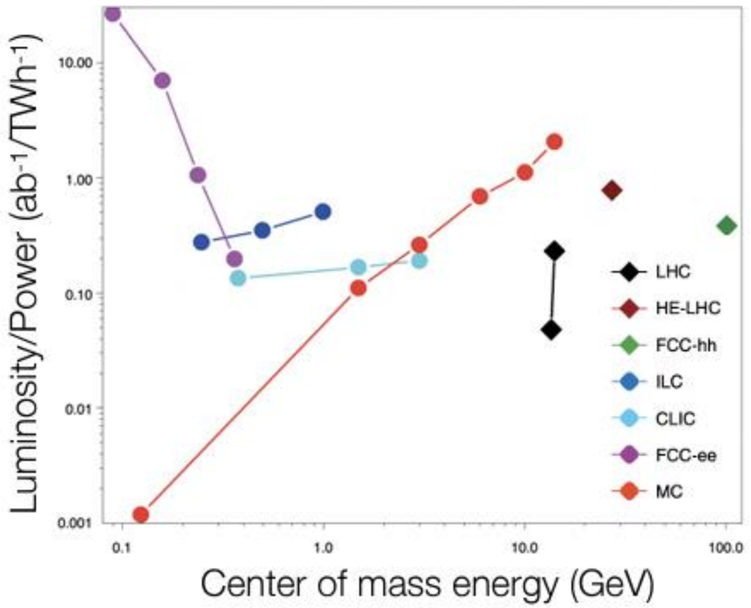
Muon Collider

- **Motivation**
  - Both a precision and high energy collider
  - Luminosity scales with energy
  - Cost effective and sustainable machine
- **Challenges**
  - Muon beam production, cooling and acceleration
  - **Beam Induced Background**
  - Neutrino flux

[3]



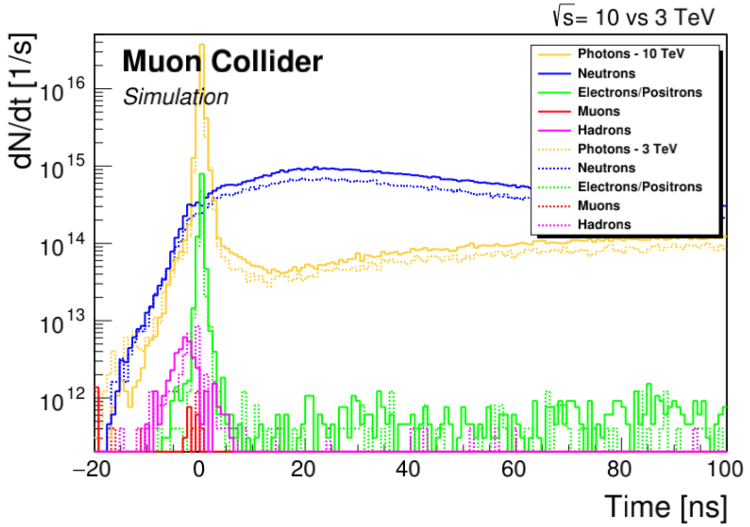
[1]

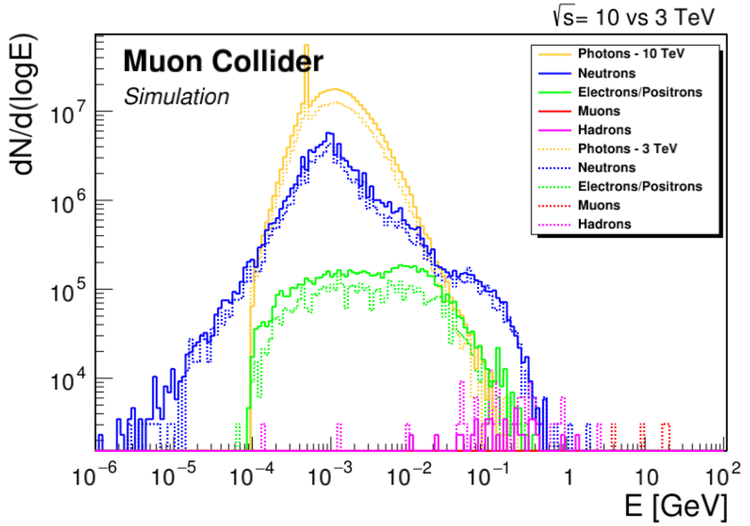


[2]

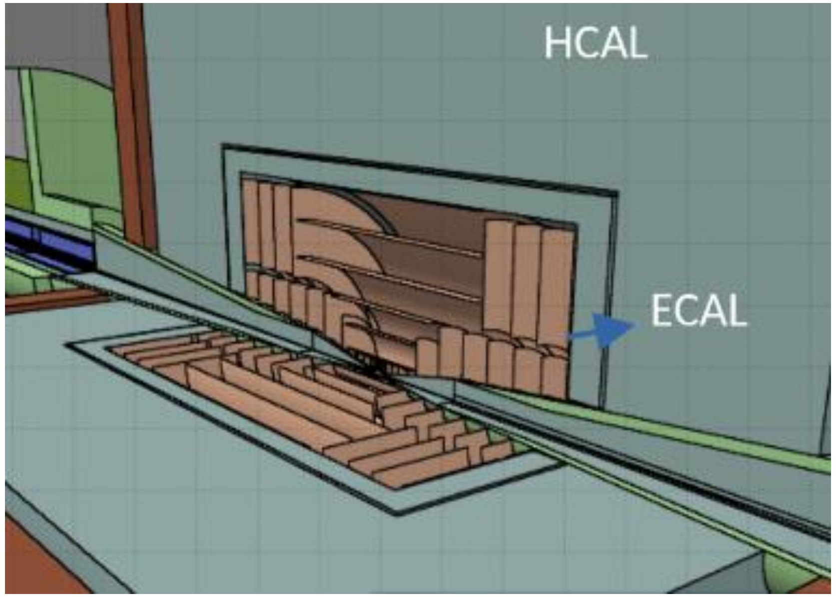
Beam Induced Background at $\sqrt{s} = 3\ TeV$

- Muon decay products interact with the machine
  - $2.34 \cdot 10^7$ *expected decays/meter* each bunch crossing
- Intense flux of particle reaches the detector each bunch crossing
  - $3.45 \cdot 10^7$ expected photons
  - $7.18 \cdot 10^6$ expected neutrons
  - $5.75 \cdot 10^5$ expected e^+/e^-
- Mitigation strategies required:
  - **Tungsten shields** (nozzles) reduce the number of particle arriving to the detector
  - Readout window of $[-1, +15]ns$ with respect to bunch crossing removes off-time Beam Induced Background

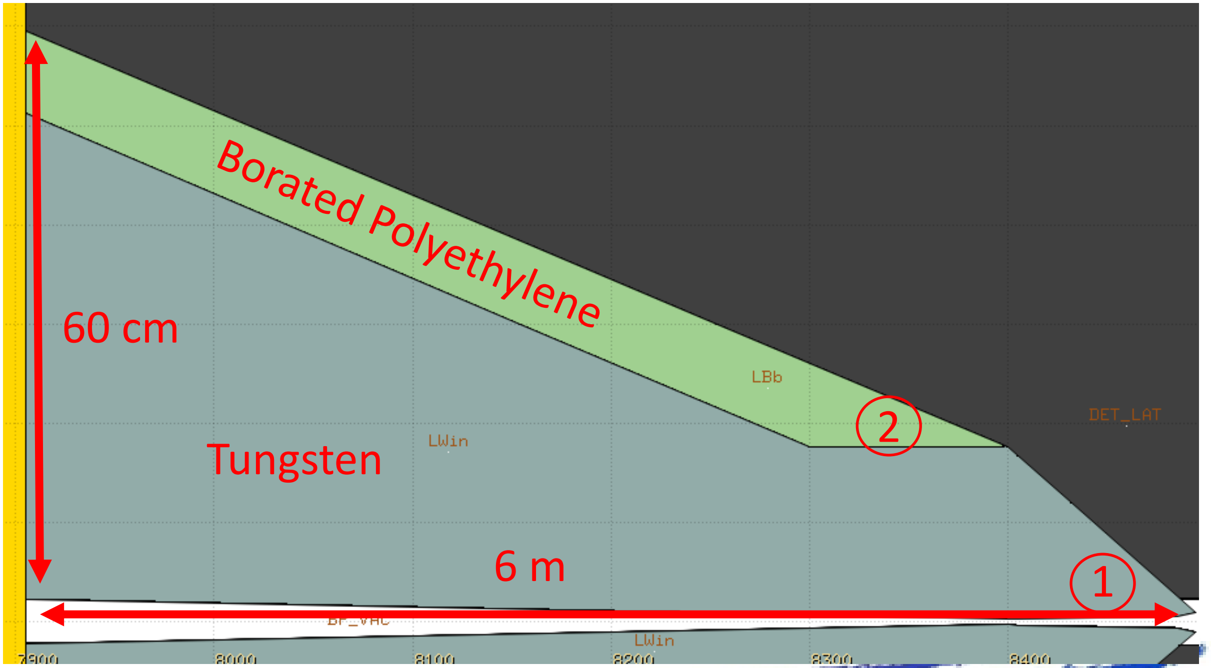




Original Design

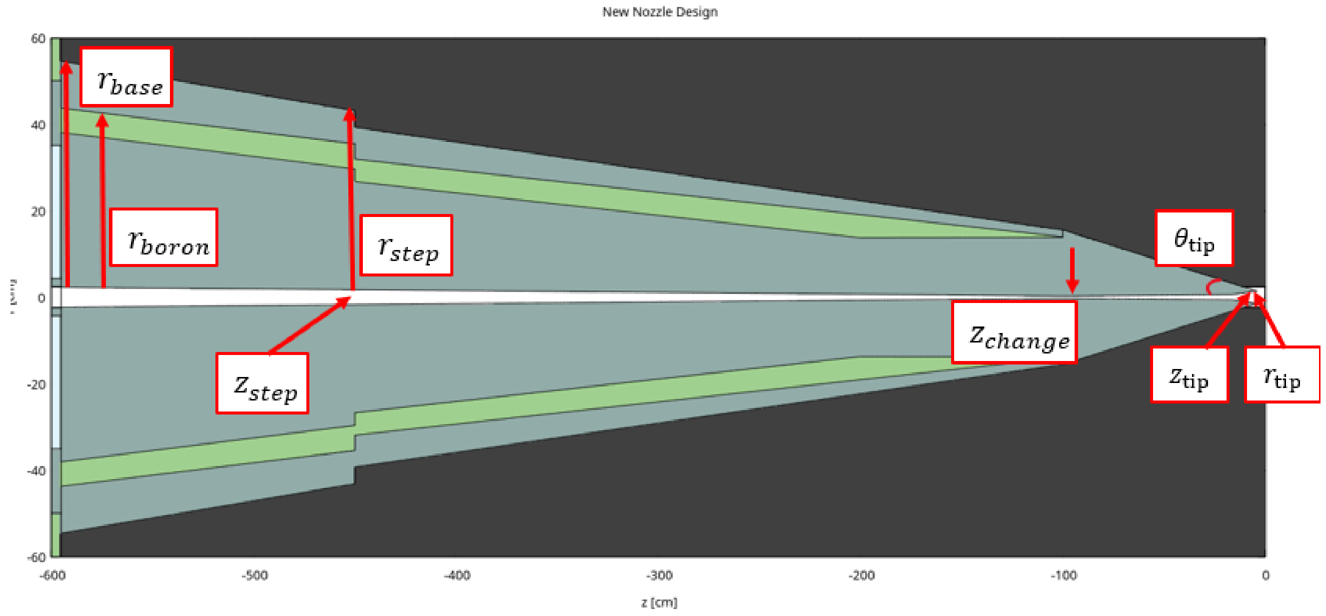


- MAP nozzle design:
  1. 10° closest to the IP
  2. 5° starting from $z = 100\ cm$
- Optimized for $\sqrt{s} = 1.5\ TeV$
- Optimization of shape and material required for **3** and $10\ TeV$
- Goal is to reduce BIB effects on detector and maximize detector acceptance
- BIB simulated with FLUKA
- The time needed to simulate depends on the computer characteristics. In my case $\sim 4\ days$ working with 8 cpus



Shape Optimization Approaches

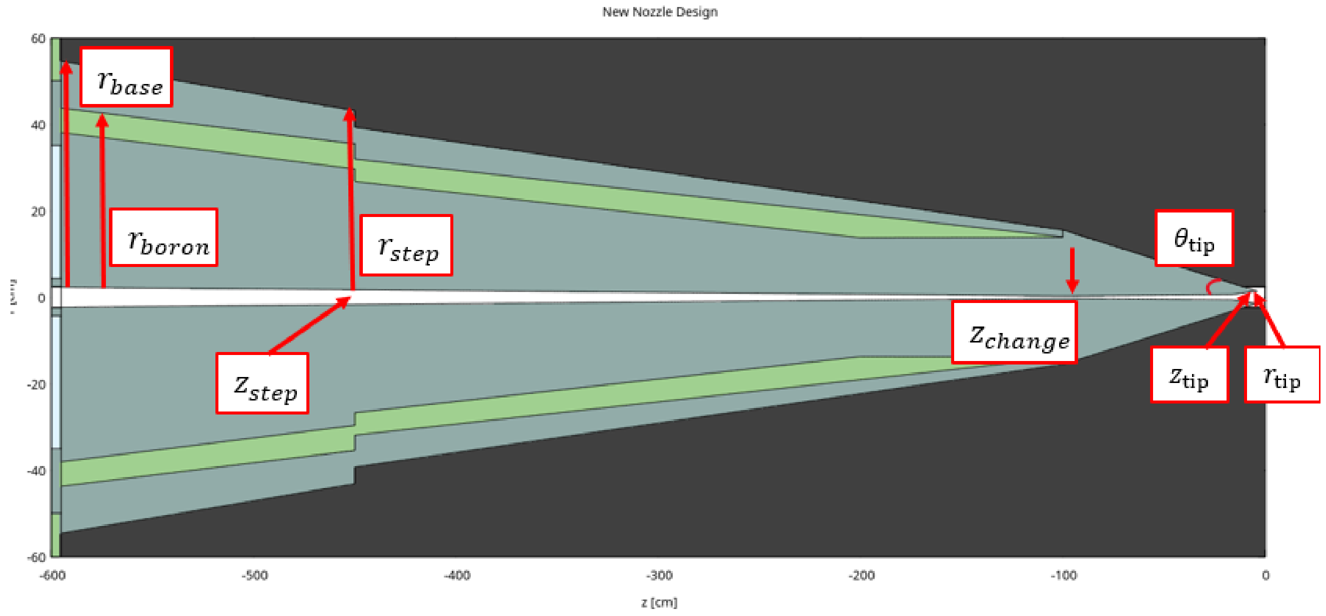
- **Hard ML approach**
  - 8 parameters, 13121 "low" statistics simulations
  - Simple XGBoost regressor
- **Smart ML approach**
  - Bayesian optimization loop with "medium" statistic
- **By hand optimization**
  - "High" statistics simulation with user-defined parameters



- **Key concepts**
  - Layer of tungsten outside the boron
  - Last $\sim 100\ cm$ geometry most impacting
  - **Small changes impact strongly the BIB**

Optimization Approaches

- **Statistics**
  - Low → 0.02% of bunch crossing
  - Medium → 0.06% of bunch crossing
  - High → 1.6% of bunch crossing
- **Parameters:**
  - $\theta_{tip} \in [0.126, 0.174]$ degree
  - $r_{base} \in [45, 60] cm$
  - $r_{boron} \in [0.8, 0.95]$% of r_{base}
  - $z_{step} \in [-450, -250] cm$ from IP
  - $r_{step} \in [0.75, 0.95]$% of r_{base}
  - $z_{change} \in [-130, -80] cm$ from IP
  - $z_{tip} \in [-6, -4] cm$ from IP
  - $r_{tip} \in [0.6, 1.4] cm$

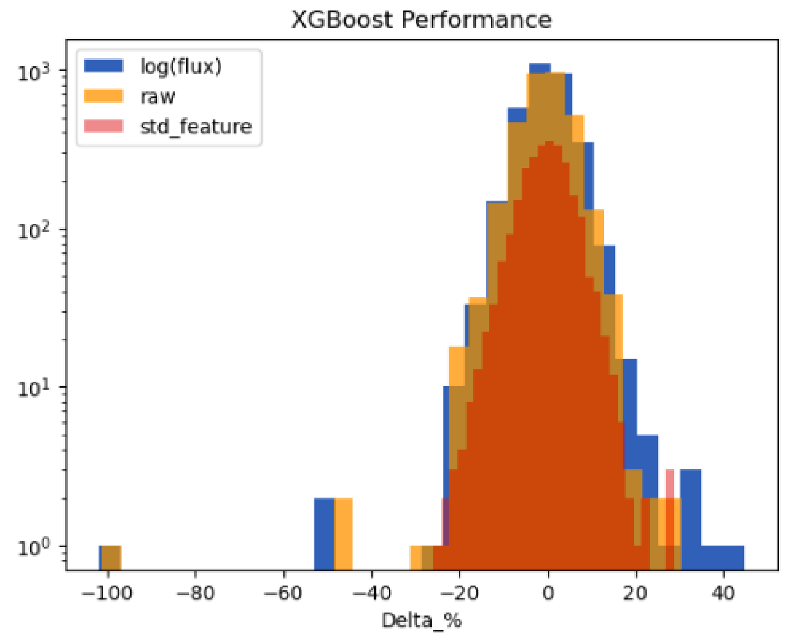


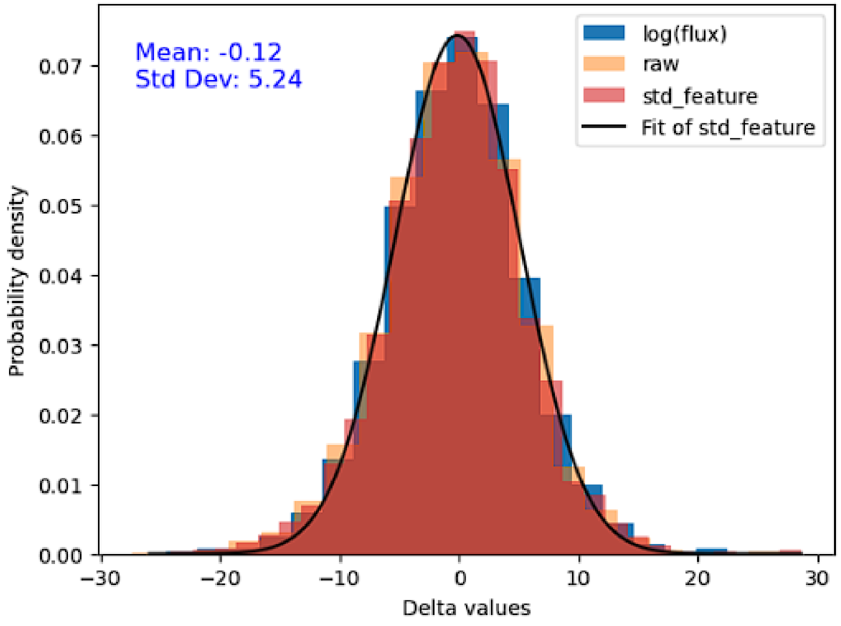
- **Figure of merit**
  - Integrated flux of particles entering the Detector area
  - $\text{Flux} \in [1.3 \cdot 10^3, 1.1 \cdot 10^5]$, for low statistic simulation

International
UON Collider
Collaboration

Hard ML results

- XGBoost regressor to predict the flux from the parameters to:
  - Perform in short time large amount of pseudo-simulation
  - Do a Bayesian optimization without running FLUKA
- Flux in 2 o.o.m range, so log applied → no effect
- Applied scalers (std, min-max) → removed outliers
- $Delta[\%] = \frac{Flux_{true} - Flux_{predicted}}{Flux_{true}} * 100$

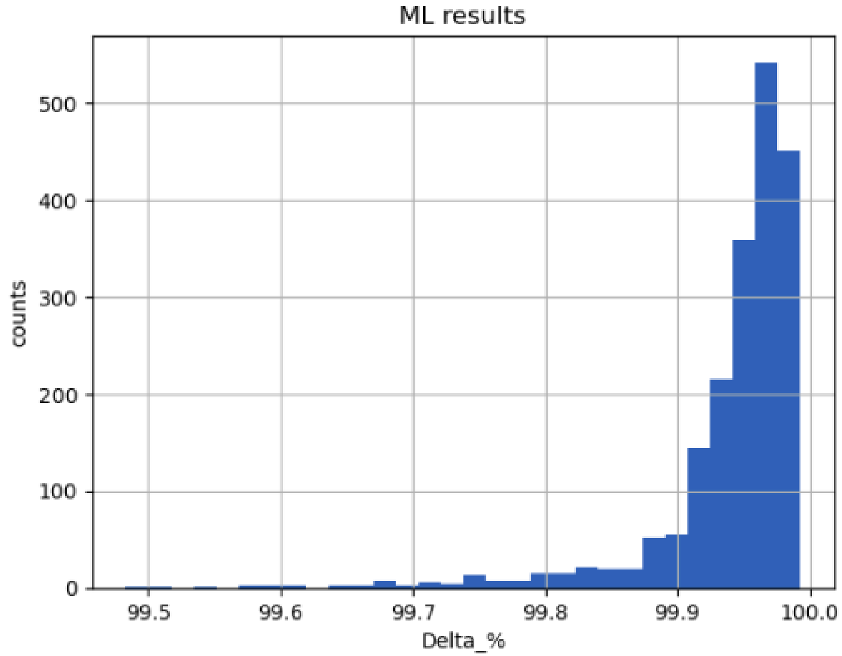




Hard ML results

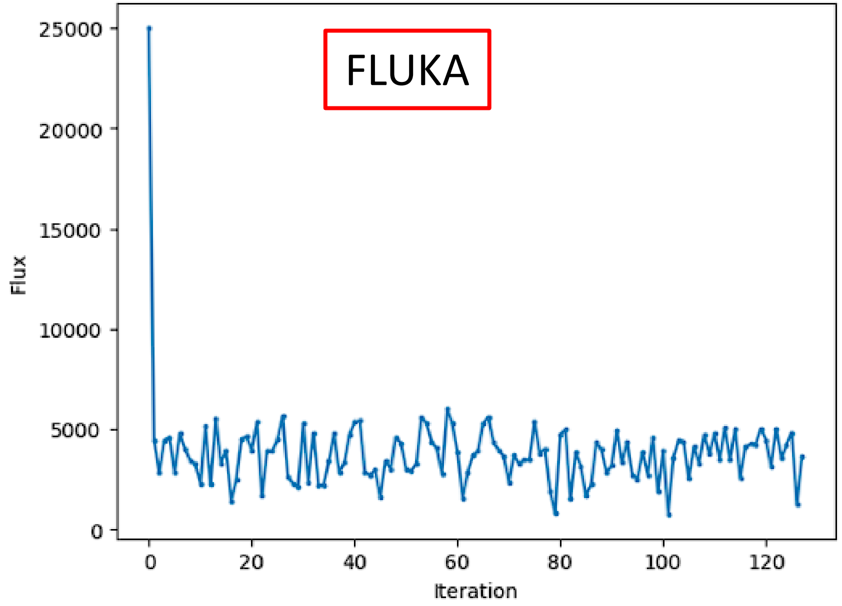
- XGBoost regressor to predict the flux from the parameters to:
  - Perform in short time large amount of pseudo-simulation
  - Do a Bayesian optimization without running FLUKA
- Flux in 2 o.o.m range, so log applied → no effect
- Applied scalers (std, min-max) → removed outliers
- $Delta[\%] = \frac{Flux_{true} - Flux_{predicted}}{Flux_{true}} * 100$
- Tried a pytorch NN model, but did not achieve any result

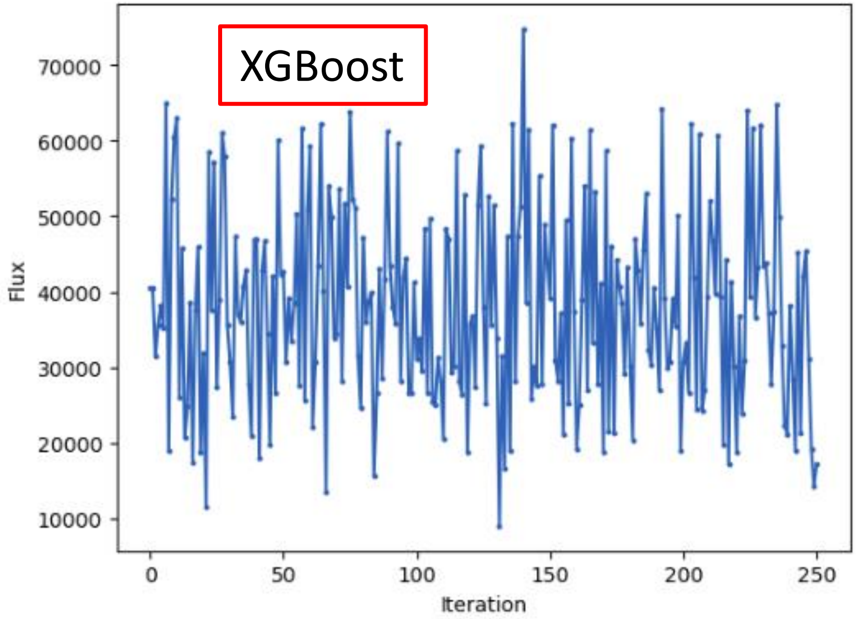
```python
class NeuralNetwork(nn.Module):
    def __init__(self, ninputs, device=torch.device("cpu")):
        super().__init__()
        #self.flatten = nn.Flatten()
        self.linear_relu_stack = nn.Sequential(
            nn.Linear(ninputs, 512),
            nn.BatchNorm1d(512),
            nn.ReLU(),
            nn.Linear(512,128),
            nn.BatchNorm1d(128),
            nn.ReLU(),
            nn.Linear(128,64),
            nn.BatchNorm1d(64),
            nn.ReLU(),
            nn.Dropout(p=0.2),
            nn.Linear(64, 1)
        )
        self.linear_relu_stack.to(device)
```



Bayesian Optimization

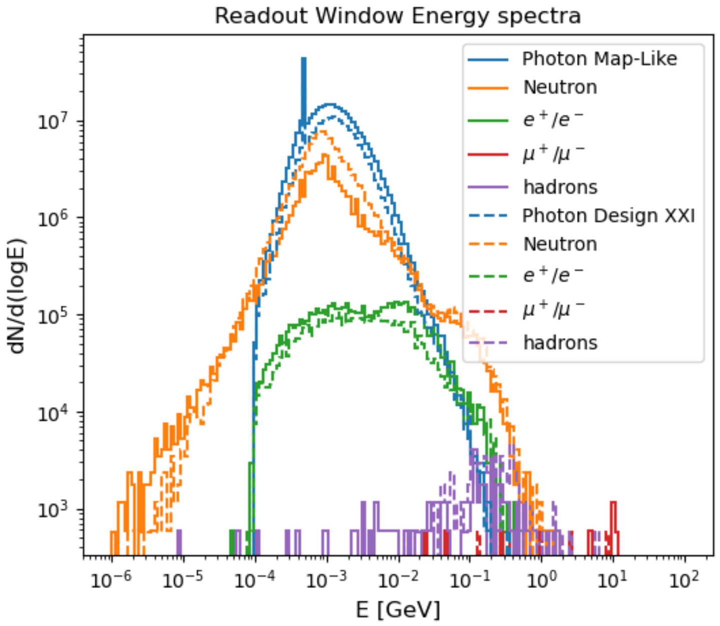
- Code provided by **T. Ramazyan** and **E. Kurbatov**
  - Optimization of black-box function → Nozzle response to incoming BIB
  - Building a probabilistic model based on past evaluation
  - Model makes an educated guess on where the best solution is in the phase-space
- Optimization loop with 126 iteration, running FLUKA simulation with medium statistics
- Run a loop with the best XGBoost model as pseudo-simulation → It did not converge

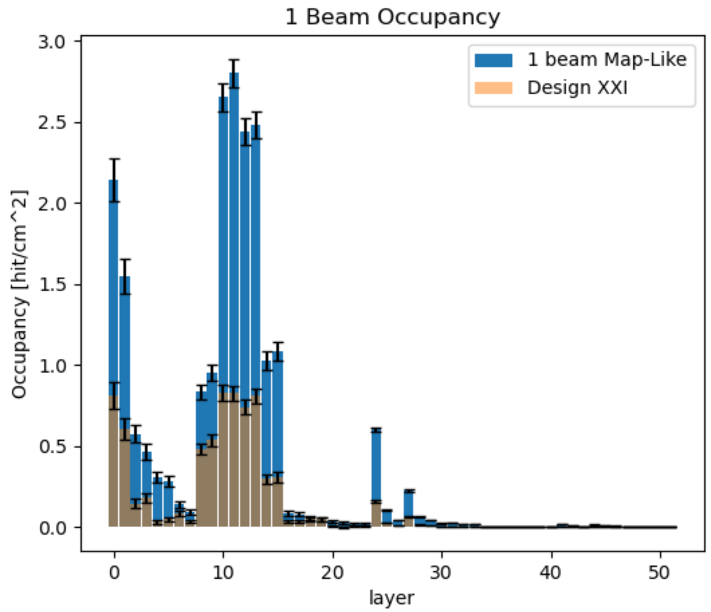




Optimization Results

- Combining XGBoost analysis and by-hand simulations, an improved design has been achieved
- BIB energy spectrum after readout window applied (left)
- Occupancy on the vertex detector (right)
- Plots compare the new geometry with the original one from MAP





Particle	MAP Design	Design XXI
photons	$3.45 \cdot 10^7$	$2.27 \cdot 10^7$
neutrons	$7.18 \cdot 10^6$	$1.34 \cdot 10^7$
e^+/e^-	$5.75 \cdot 10^5$	$4.19 \cdot 10^5$

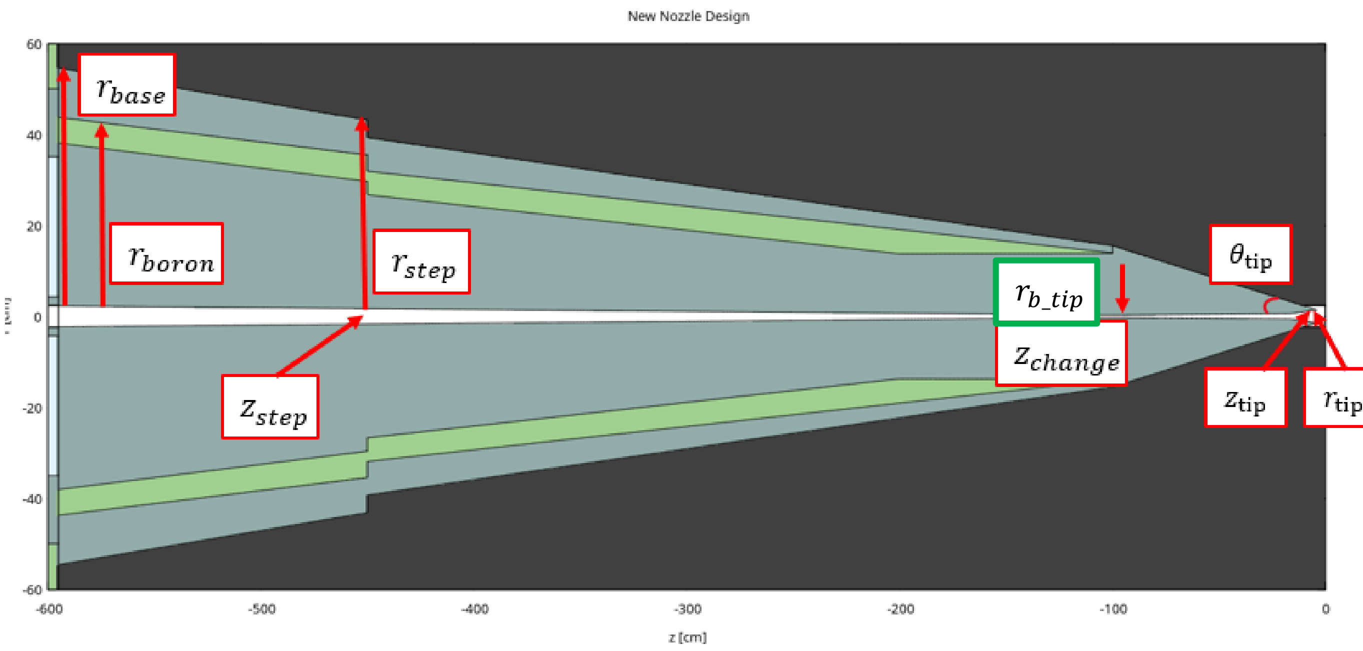
Flux per bunch crossing

	MAP Design	Design XXI
Layer 0	2.14	0.81
Layer 1	1.54	0.61

Occupancy for 1.6% of bunch crossing

Conclusions and next steps

- A first optimization has been achieved
- ML algorithm can be optimized
- A 9th parameters will be considered
- Optimization on a complex observable



$$flux \rightarrow a \cdot \frac{\Delta flux_{\gamma}}{flux_{ref_{\gamma}}} + b \cdot \frac{\Delta flux_{n}}{flux_{ref_{n}}} + c \cdot \frac{\Delta flux_{e}}{flux_{ref_{e}}} + d \cdot \frac{\Delta V}{V_{ref}}$$

Any suggestion?

Reference

- [1] C. Accettura et al., Towards a Muon Collider, arxiv.org
- [2] K. Long, D. Lucchesi, M. Palmer, N. Pastrone, D. Schulte, V. Shiltsev, Muon Colliders: Opening New Horizons for Particle Physics, arxiv.org
- [3] C. Accettura et al., Interim report for the International Muon Collider Collaboration, arxiv.org
- [4] Y. Alexahin, E. Gianfelice-Wendt and V. Kapin, MUON COLLIDER LATTICE CONCEPTS, Iopscience.iop.org

Backup

Muon Collider Parameters

International
MUON Collider
Collaboration

Target integrated luminosities

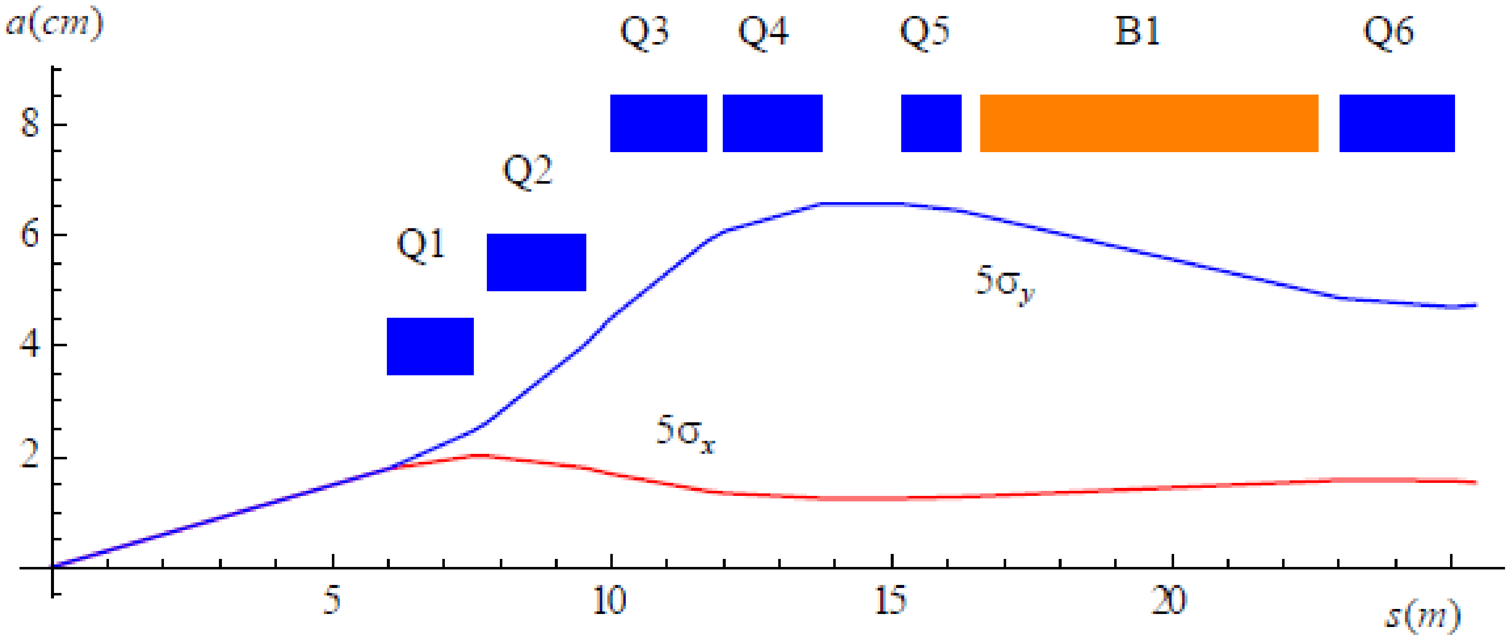
$\sqrt{s}$	$\int \mathcal{L}dt$
3 TeV	$1\ \mathrm{ab}^{-1}$
10 TeV	$10\ \mathrm{ab}^{-1}$
14 TeV	$20\ \mathrm{ab}^{-1}$

Note: currently focus on 10 TeV, also explore 3 TeV

- Tentative parameters based on MAP study, might add margins
- Achieve goal in 5 years
- FCC-hh to operate for 25 years
- Aim to have two detectors

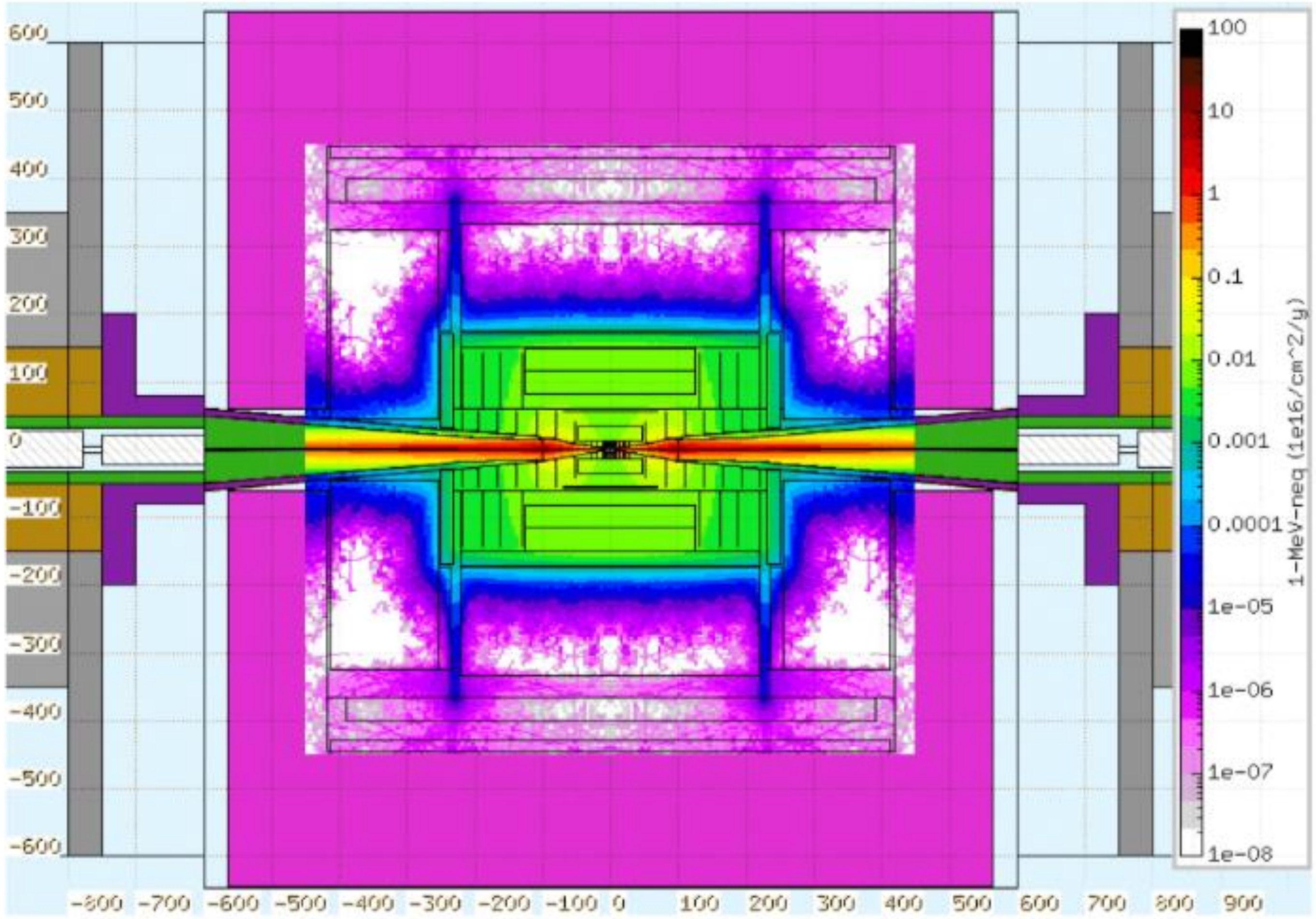
Parameter	Unit	3 TeV	10 TeV	14 TeV	CLIC at 3 TeV
L	$10^{34}\ \mathrm{cm}^{-2}\mathrm{s}^{-1}$	1.8	20	40	2 (6)
N	10^{12}	2.2	1.8	1.8	
f_r	Hz	5	5	5	
P_{beam}	MW	5.3	14.4	20	28
C	km	4.5	10	14	
<B>	T	7	10.5	10.5	
ε_L	MeV m	7.5	7.5	7.5	
σ_E / E	%	0.1	0.1	0.1	
σ_z	mm	5	1.5	1.07	
β	mm	5	1.5	1.07	
ε	µm	25	25	25	
$\sigma_{x,y}$	µm	3.0	0.9	0.63	

$\sqrt{s} = 1.5\ TeV$ Design



Dose

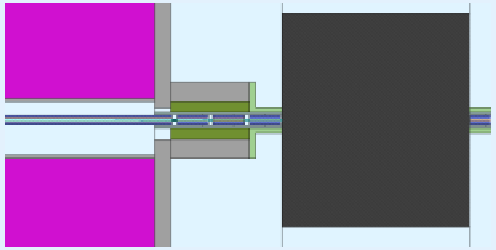
- Considering 200 operational days/year, 1-MeV-neq fluence is expected to be:
  - $\sim 10^{14-15}\ cm^{-2}y^{-1}$ in the tracker
  - $\sim 10^{14}\ cm^{-2}y^{-1}$ in the electromagnetic calorimeter



BIB simulation with FLUKA

1. From muon decay to nozzle area

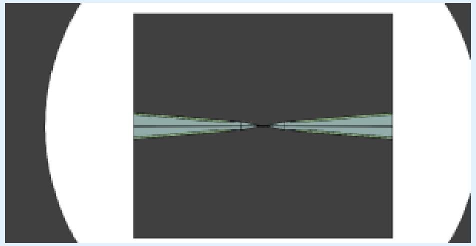
Machine dependent



- Generated one beam of μ^+ decays within $\mathbf{55\ m}$ from the Interaction Point
- **Energy threshold** for particles production fixed at $\mathbf{100\ keV}$
- Particles which arrives to the nozzles are scored

2. Nozzle area to detectors

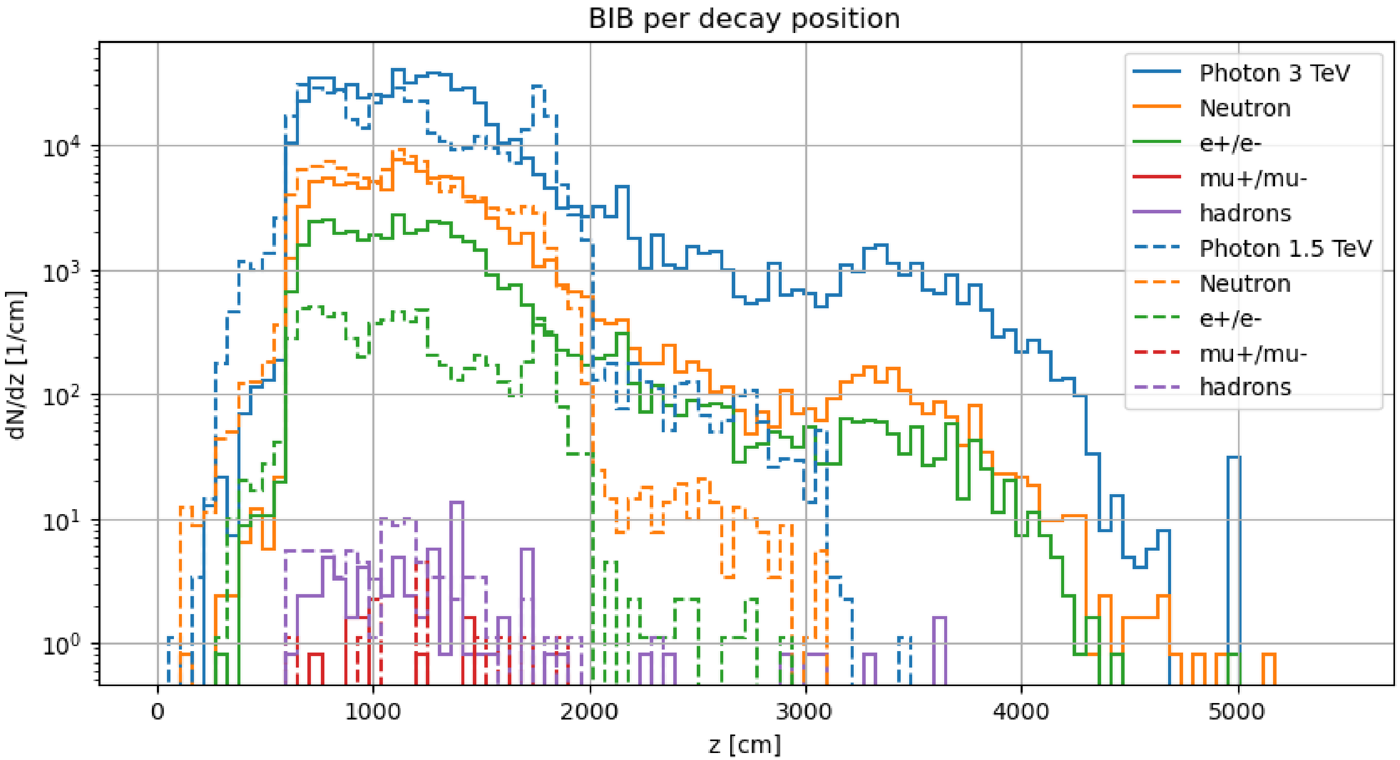
Nozzle dependend



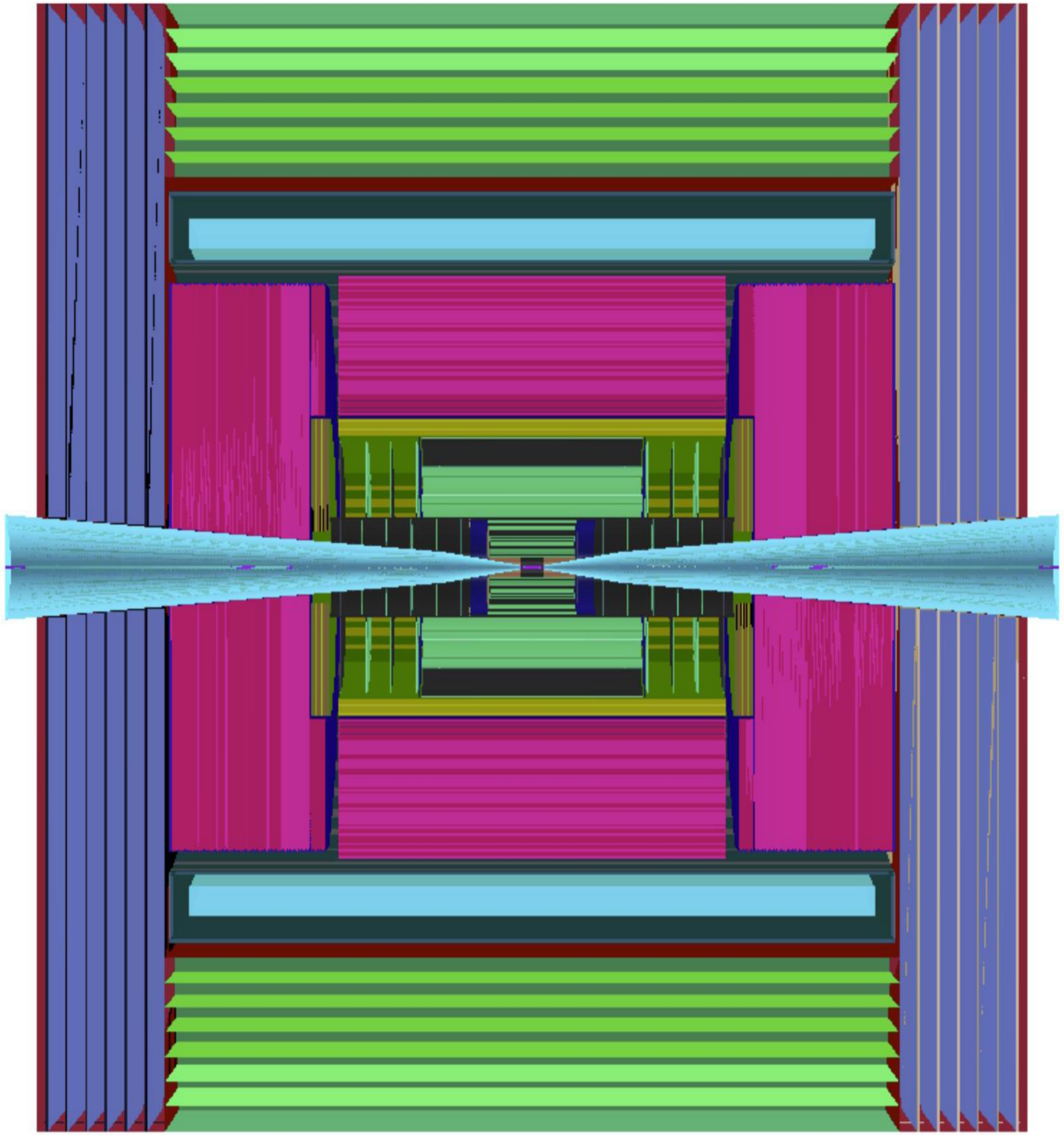
- Propagation through the Nozzles
- Particles who exit the nozzle and enters the detector area are scored
- $\sim$1.6% of one BIB event (i.e. bunch crossing) considering only 1 beam → $\mathbf{4\ days}$ **per simulation**

Muon decay position

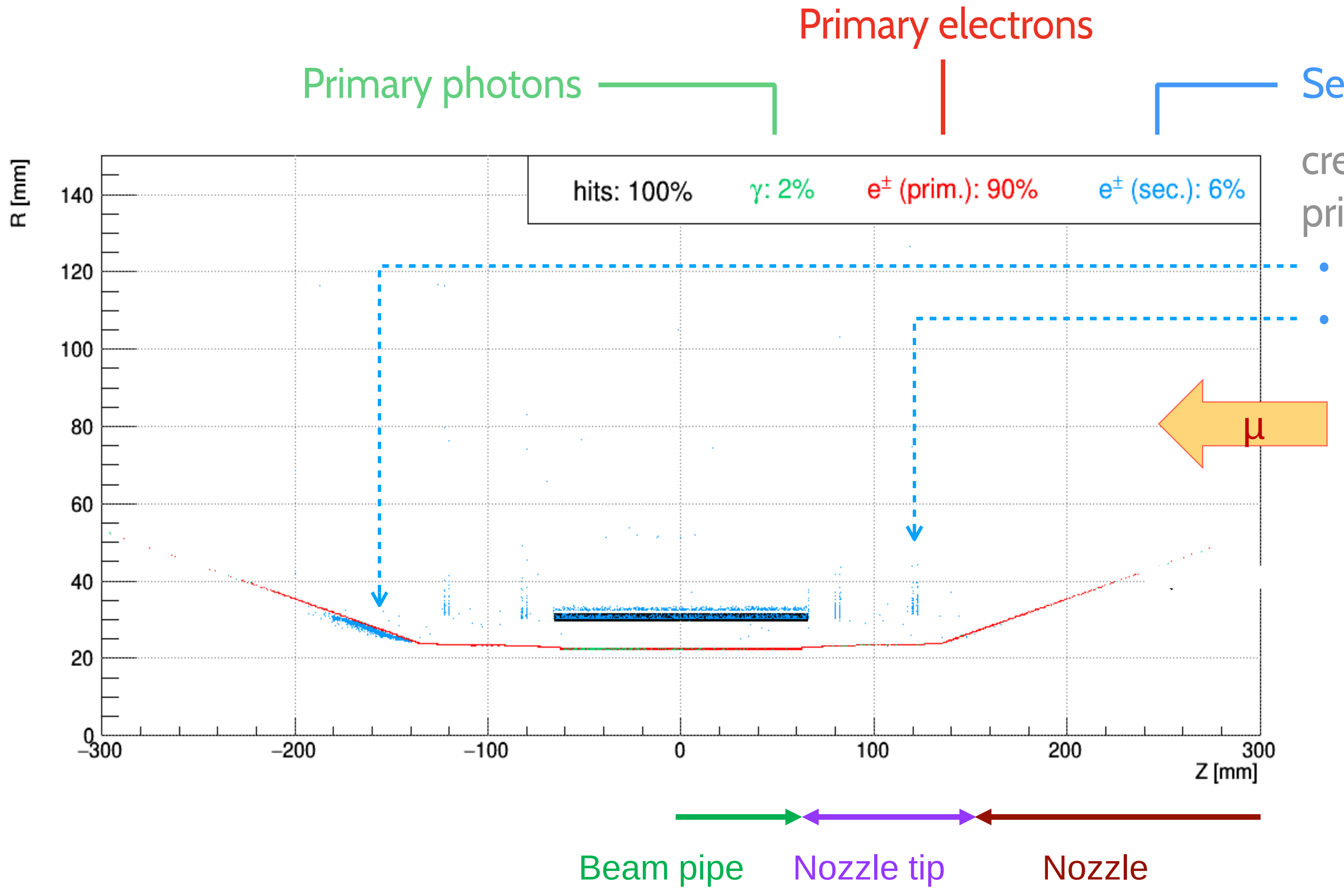
- Muon decays up to 55 m cause BIB in the detector for the $\sqrt{s} = 3\ TeV$ case
- In the $\sqrt{s} = 1.5\ TeV$ it is enough to consider decays from 35 m



Detector



Occupancy in the tracker



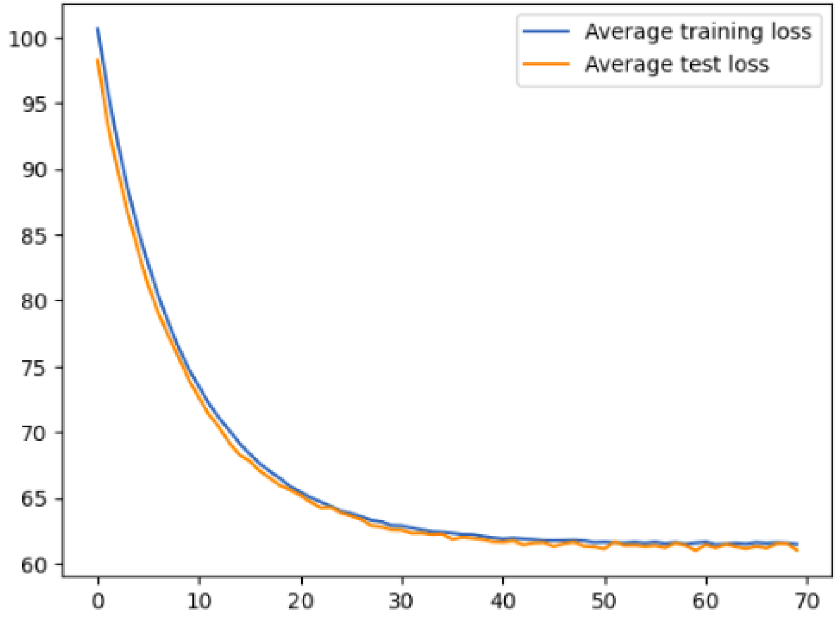
created by interaction of primary photons with:

- MDI material
- detector material

Hard ML results

- XGBoost regressor to predict the flux from the parameters to:
  - Perform in short time large amount of pseudo-simulation
  - Do a Bayesian optimization without running FLUKA
- Flux in 2 o.o.m range, so log applied → no effect
- Applied scalers (std, min-max) → removed outliers
- Prediction correct within 10% with best model
- Tried a pytorch NN model, but did not achieve any result

```python
class NeuralNetwork(nn.Module):
    def __init__(self, ninputs, device=torch.device("cpu")):
        super().__init__()
        #self.flatten = nn.Flatten()
        self.linear_relu_stack = nn.Sequential(
            nn.Linear(ninputs, 512),
            nn.BatchNorm1d(512),
            nn.ReLU(),
            nn.Linear(512,128),
            nn.BatchNorm1d(128),
            nn.ReLU(),
            nn.Linear(128,64),
            nn.BatchNorm1d(64),
            nn.ReLU(),
            nn.Dropout(p=0.2),
            nn.Linear(64, 1)
        )
        self.linear_relu_stack.to(device)
```



Hard ML results

- Feature Importance with XGBoost regressor