ML for the RICH alignment

Armen Gyurjinyan
INFN

LNF Workshop 13/12/2022 – 16/12/2022

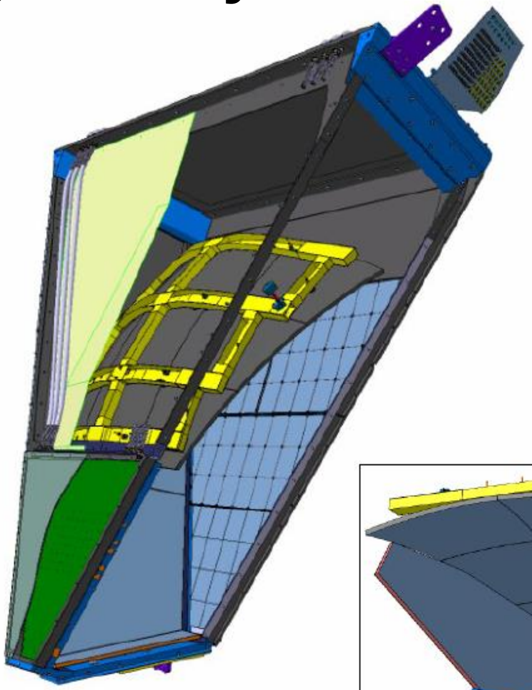
Outline

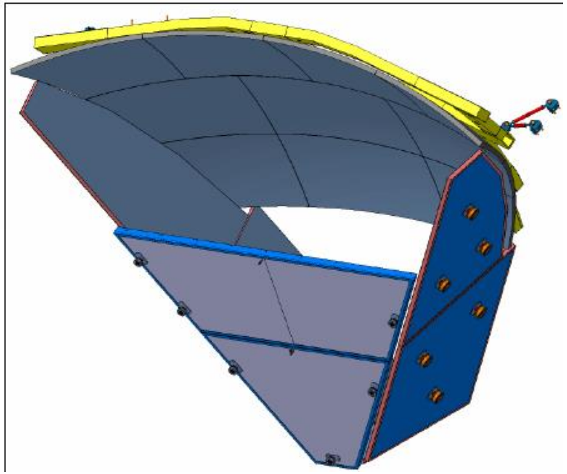
- Introduction to Fast Monte Carlo simulations
  - RICH Geometry for the simulation
  - Generation and reconstruction
  - Quantify the quality of alignment
  - Geometry and output file structure
- Best parameters for alignment with NN model
  - Minima finding with FCN and SGD
  - Fully connected network
  - Stochastic gradient descent with momentum
- ML results

RICH geometry

Few simplifications in the geometry definitions

- **1 lateral mirror per side instead of 2**
- **1 spherical mirror instead of 10**
- **each aerogel layer is made by only 1 large tile; tile segmentation is done in the data output based on the emission point coordinates**
- **the MAPMT array is segmented in a regular matrix of 6.5x6.5 mm pixels; PMT segmentation is done into output data based on the hit point coordinates**





Event generation

1. **Generate a charged particle (momentum and production vertex) and propagate to the closest aerogel layer**
- no magnetic field, only straight tracks
- only electrons
2. **Propagate the track to the MAPMT plane and calculate the pixel number (cluster position)**
3. **Calculate the number of Cherenkov photons (Poisson distribution with given mean)**
4. **For each photon**
- calculate the emission point: randomly along the particle track in the aerogel
- calculate the Cherenkov angle based on β and n: some smearing is applied (σC=4.5 mrad)
- propagate the photon through the RICH based on the given geometry until it reaches the MAPMT
- calculate the pixel number
- store information on the generated photon: hit position, path length and time, number of reflections, mirror hit position, etc

Event reconstruction

Track hit

- Take the generated pixel number (if any), calculate the position with reconstruction (modified) geometry

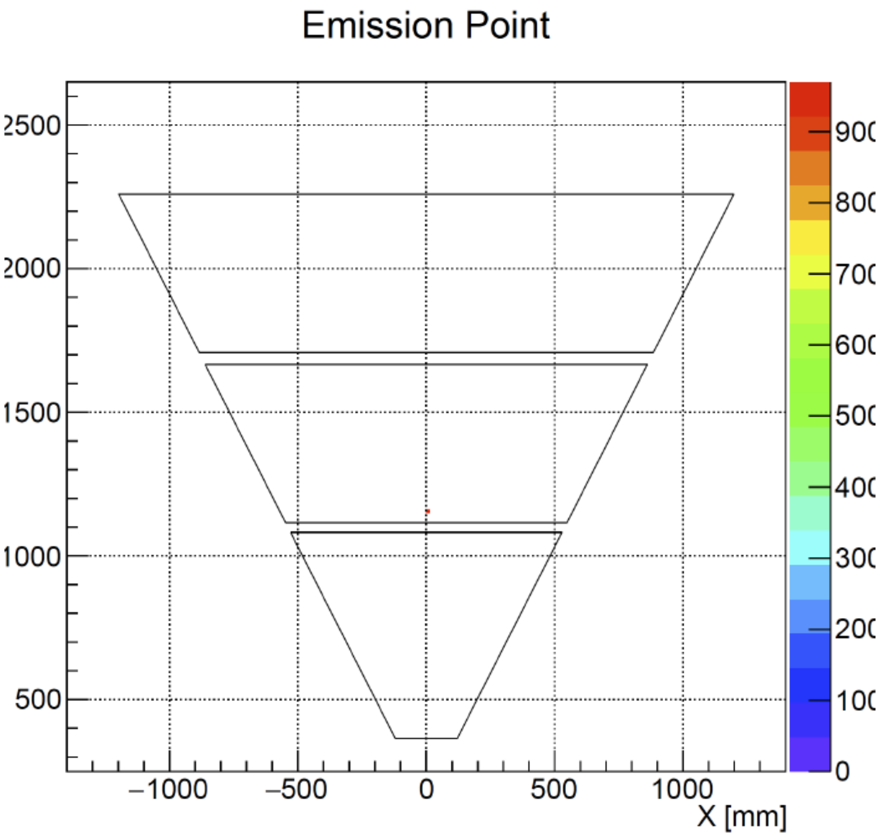
Photon hits

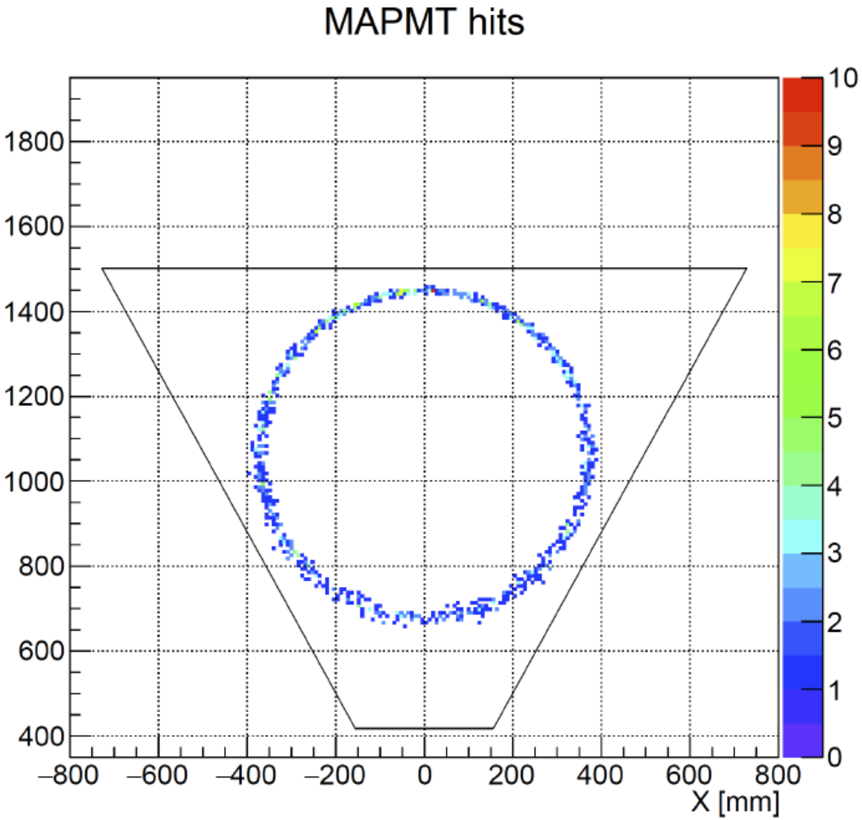
- take the generated pixel hit, calculate the position with reconstruction (modified) geometry: detected hit
- calculate the emission point: mid point in the aerogel
- first try Cherenkov angles: θ from the nominal refractive index, φ from the generated hit (to have fast convergence)
- propagate the photon to the MAPMT with the current (modified) geometry
- use real data reconstruction method

Store the results of the reconstructed event

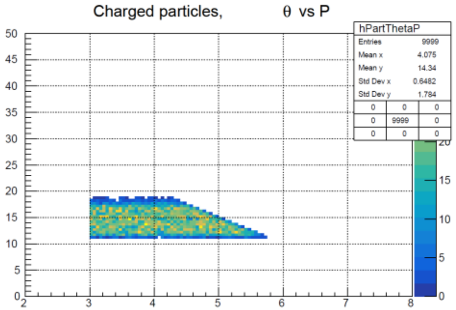
- hit position, path length and time, number of reflections, mirror hit position, etc

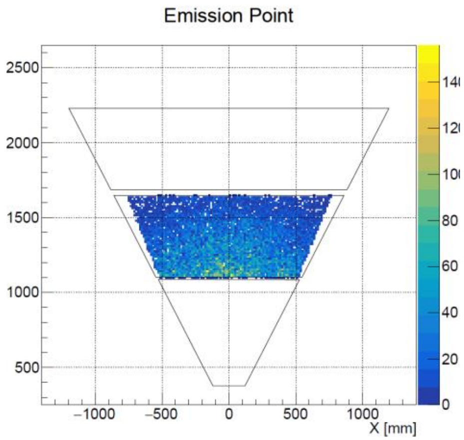
Direct photons

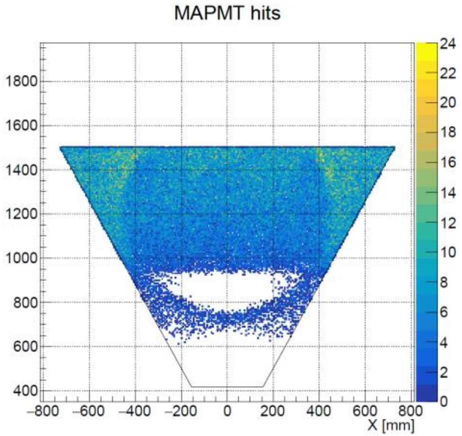


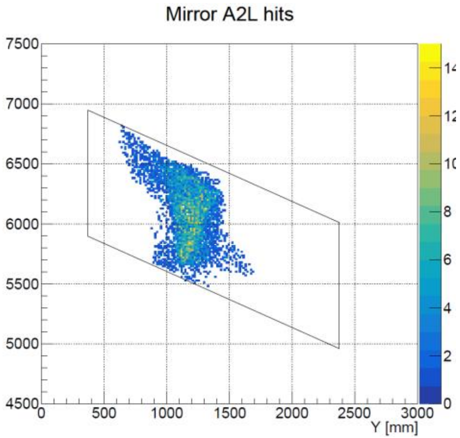


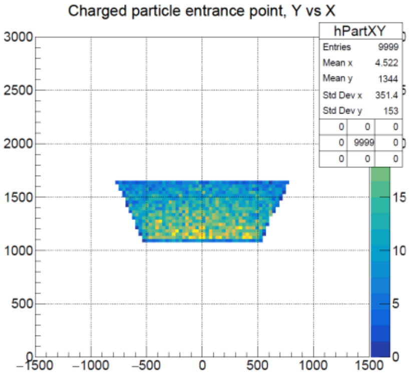
Generated events

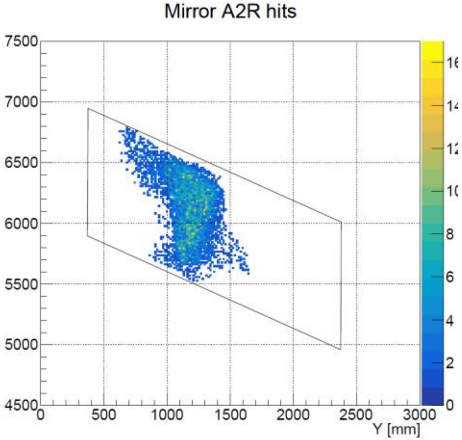


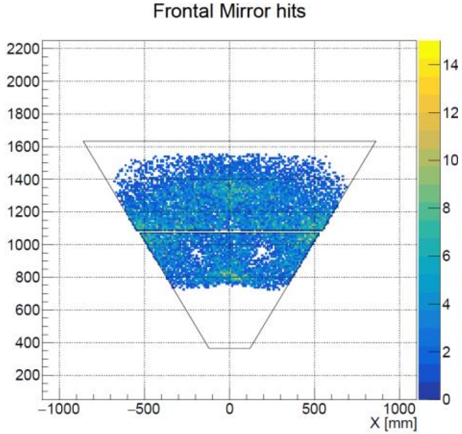


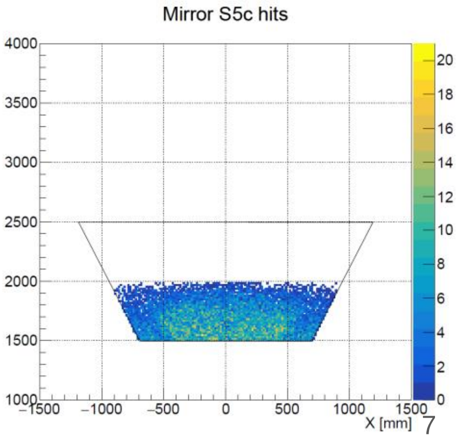




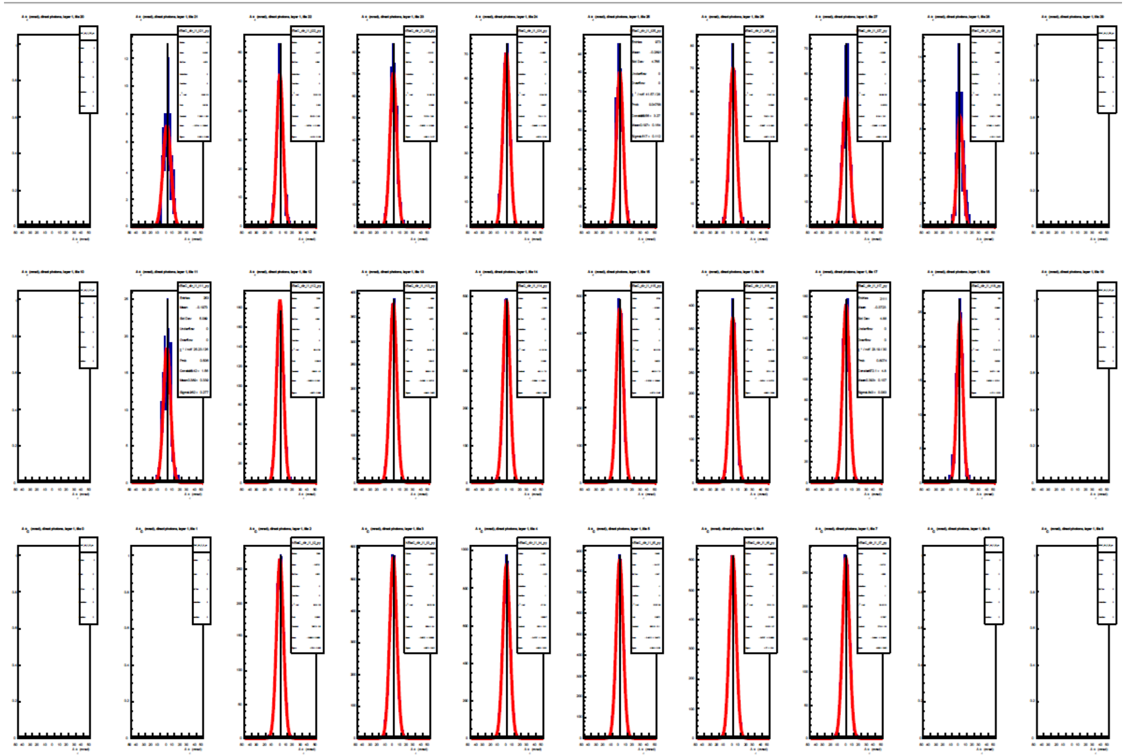








Cherenkov angle reconstruction by tiles



from gauss fit: mean m, sigma σ

- expected values are
  - mean: <m>=0
  - sigma: <σ>=input resolution
- chisquare

$$\chi^2 = \left(\frac{m - \langle m \rangle}{\Delta m}\right)^2 + \left(\frac{\sigma - \langle \sigma \rangle}{\Delta \sigma}\right)^2$$

Output for all tiles:

$$\chi^2 = \frac{1}{N}\sum_{i=1}^{N} \chi_i^2$$

FastMC reconstruction output

ID, Layer ID, ***Δθ*** mean, error, ***Δθ*** std, error, N entries, $\chi 2$

	0	0	0.000	0.000	0.000	0.000	0	0.0
Direct photons	0	1	-0.283	0.323	4.817	0.156	22	15.7
	0	2	0.000	0.000	0.000	0.000	0	0.0
	1	0	0.000	0.000	0.000	0.000	0	0.0
1 reflection left mirror	1	1	-0.047	0.199	4.835	0.055	4	1.9
	1	2	0.000	0.000	0.000	0.000	0	0.0
	2	0	0.000	0.000	0.000	0.000	0	0.0
1 reflection right mirror	2	1	-0.231	0.410	4.704	0.345	5	4.5
	2	2	0.000	0.000	0.000	0.000	0	0.0
	3	0	0.000	0.000	0.000	0.000	0	0.0
1 reflection bottom mirror	3	1	0.000	0.000	0.000	0.000	0	0.0
	3	2	0.000	0.000	0.000	0.000	0	0.0
	4	0	0.000	0.000	0.000	0.000	0	0.0
2 reflections spherical + b1	4	1	-0.178	0.125	4.856	0.476	10	2.4
	4	2	0.000	0.000	0.000	0.000	0	0.0
	5	0	0.000	0.000	0.000	0.000	0	0.0
2 reflections spherical + b2	5	1	-0.275	0.386	4.740	0.340	18	1.4
	5	2	0.000	0.000	0.000	0.000	0	0.0
	6	0	0.000	0.000	0.000	0.000	0	0.0
other	6	1	1.999	2.102	11.727	5.834	16	95.5
	6	2	0.000	0.000	0.000	0.000	0	0.0

FastMC reconstruction geometry input

```
AerogelB1 surface: shifts (mm), thetax,thetay,thetaz (rad)
0. 0. 1.
0.004 0.002 0.0
AerogelB2 surface: shifts (mm), thetax,thetay,thetaz (rad)
0. 0. 1.
0.004 0.00 0.0
AerogelB3 surface: shifts (mm), thetax,thetay,thetaz (rad)
0. 0. 0.
0.000 0.000 0.0
FrontalMirrorB1 surface: shifts (mm), thetax,thetay,thetaz (rad)
0. 0. 1.
0.004 0.002 0.0
FrontalMirrorB2 surface: shifts (mm), thetax,thetay,thetaz (rad)
0. 0. 1.
0.004 0.000 0.0
PlanarMirrorL surface: shifts (mm), thetax,thetay,thetaz (rad)
0. 0. 0.
0.00 0.00 0.0
PlanarMirrorR surface: shifts (mm), thetax,thetay,thetaz (rad)
0. 0. -1.
-0.010 -0.002 0.0
BottomMirror surface: shifts (mm), thetax,thetay,thetaz (rad)
0. 0. 0.
0. 0. 0.0
SphericalMirror surface: shifts (mm), thetax,thetay,thetaz (rad)
0. 0. 0.
0.002 0.001 0.0
MAPMT surface: shifts (mm), thetax,thetay,thetaz (rad)
0 0 0
0.0 -0.0 0.0
```

- Generate data with z = 1mm and θ_y = 0 mrad
- Reconstruct data with grid z = (-5, 10, 1) mm and θ_y = (-20, 20, 5) mrad

Total 144 grid points

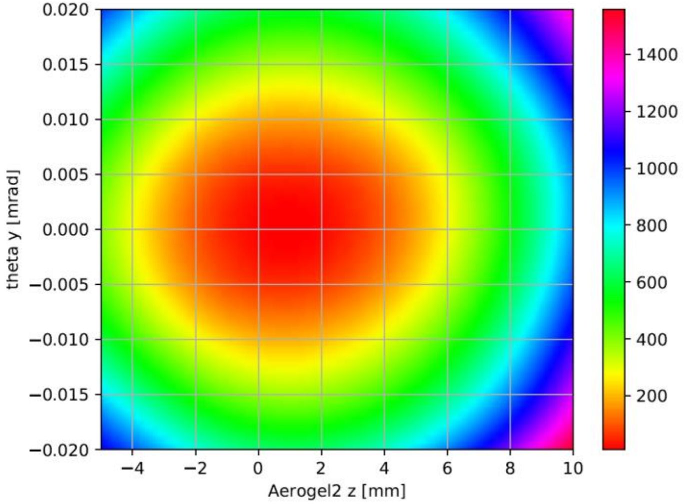
Optimal point in the reconstruction grid should be close to generated data

NN training and optimal parameters finding

(z, θ_y) → Neural Network → $(\chi^2_{dp}, \chi^2_{a2l}, \chi^2_{a2r}, \chi^2_{s5cb1}, \chi^2_{s5cb2})$ → Optimal parameter finding for

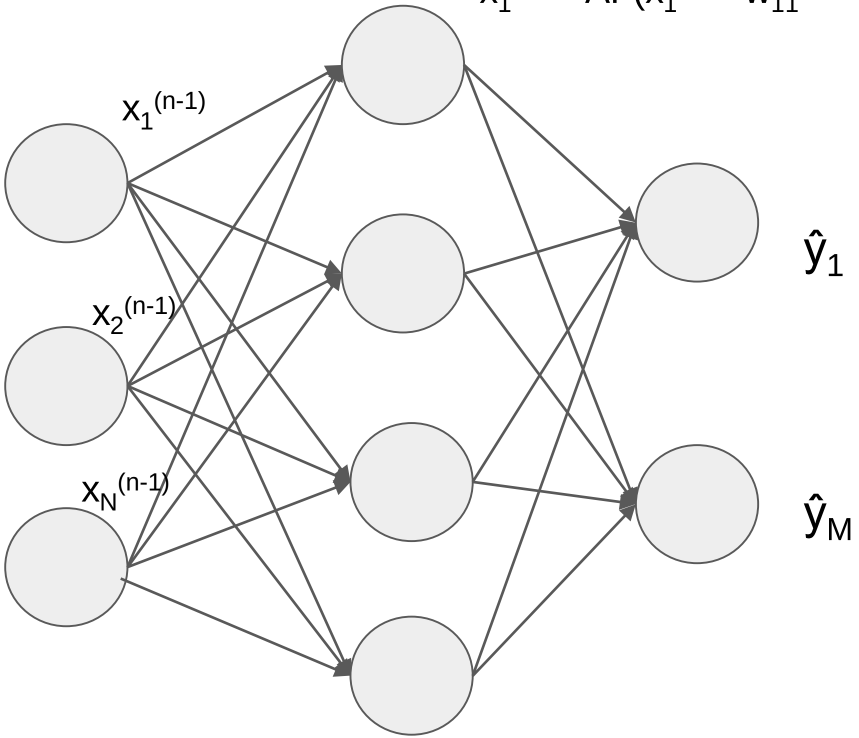
$$\sqrt{(\chi^2_{dp})^2+(\chi^2_{a2l})^2+(\chi^2_{a2r})^2+(\chi^2_{s5cb1})^2+(\chi^2_{s5cb2})^2}$$

- Train neural network on 80% of grid data points and validate on 20% data points. Best model on validation data points will be our model
- Use stochastic gradient descent with momentum to find the optimal parameters for the best model.



Fully Connected Neural Network

$$x_1^{(n)} = AF(x_1^{(n-1)} * w_{11}^{(n-1)} + x_2^{(n-1)} * w_{12}^{(n-1)} + \ldots + x_N^{(n-1)} * w_{1N}^{(n-1)})$$



Used activation function $S(x) = \dfrac{1}{1+e^{-x}}$

Best model (best weights) $min\left(\dfrac{1}{M}\sum_{i=1}^{M}(\hat{y}_i - y_i)^2\right)$

In our case:

$x_1^{(1)}$, $x_2^{(1)}$ are z, θ_y

In our case:

$\hat{y}_1$, $\hat{y}_2$, $\hat{y}_3$, $\hat{y}_4$, $\hat{y}_5$ are

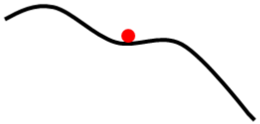
X^2_{dp}, X^2_{a2l}, X^2_{a2r}, X^2_{s5cb1}, X^2_{s5cb2}

Stochastic gradient descent with momentum

$$\frac{dy}{dx} = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

Local minima

Saddle points



SGD

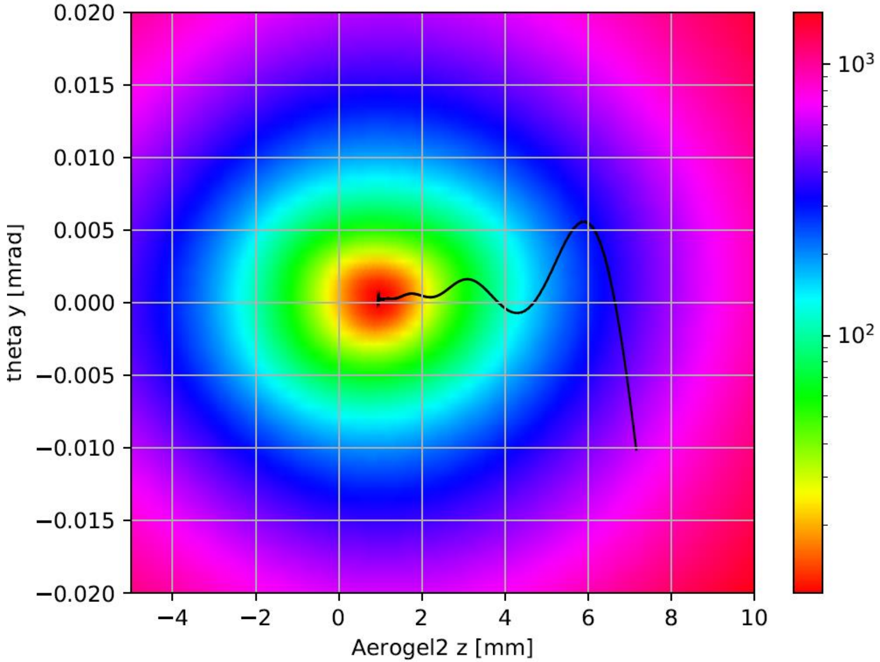
$$x_{t+1} = x_t - \alpha * \nabla f(x_t)$$

SGD + momentum

$$v_{t+1} = \rho v_t + \nabla f(x_t)$$

$$x_{t+1} = x_t - \alpha * v_{t+1}$$

"α" and "ρ" are constants

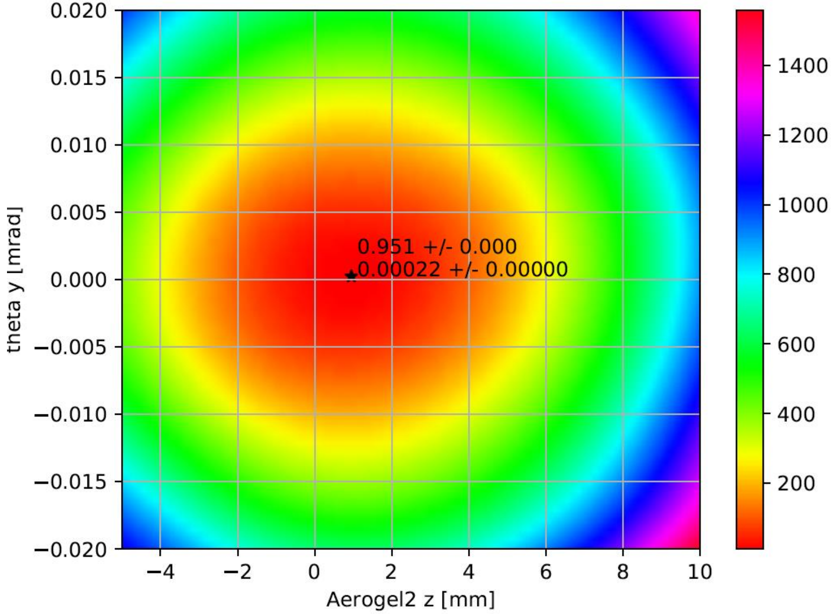


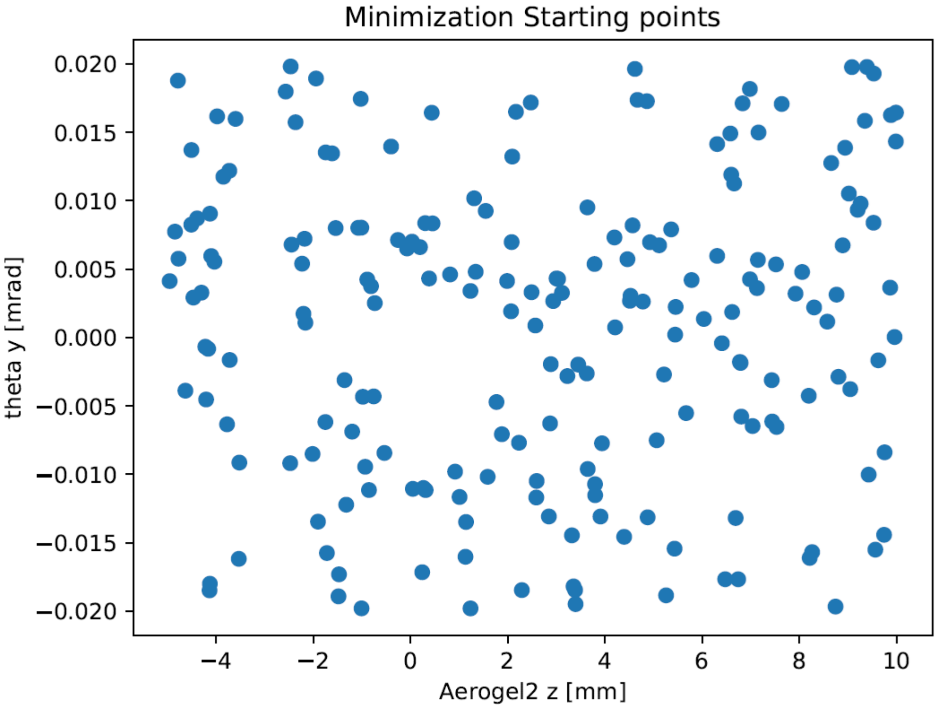
In our experiment

h=10^{-1} mm for aerogel2 z
h=10^{-4} mrad for aerogel2 θ_y

Results

Generated events geometry z = 1 mm, θ_y = 0 mrad





Generated 200 starting point and the results are average of optimized points and error is standard deviation

Summary and Next steps

Summary

- Tested machine learning algorithm for 2d input parameters model training and predictions on simulated data
- Tested algorithm to find best parameter on simulated data for the given model

Next steps

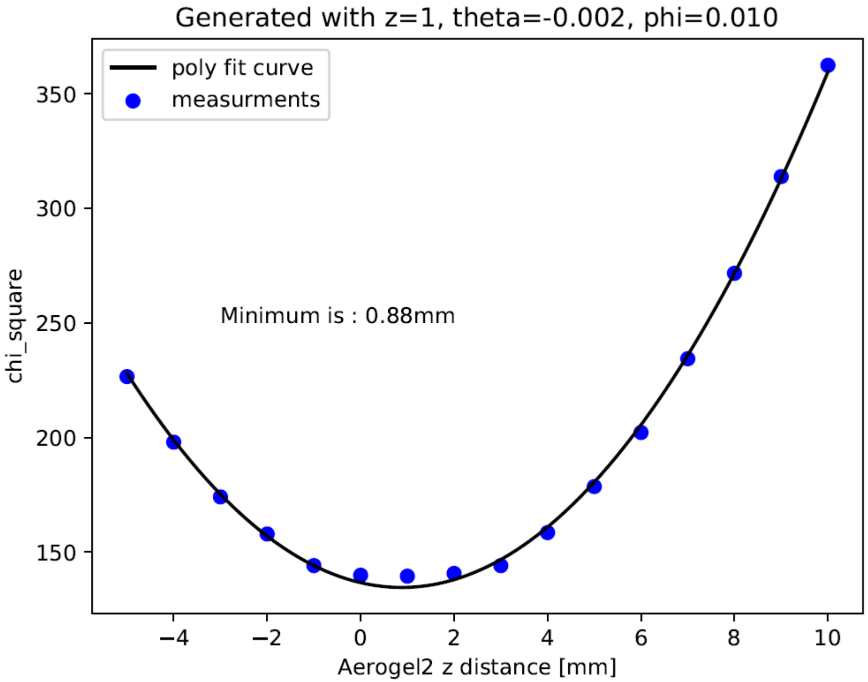
- Implement errors for the output parameters
- Add more parameters for training and best parameters finding
- Use real data in place of simulated data

Thank you for your attention!

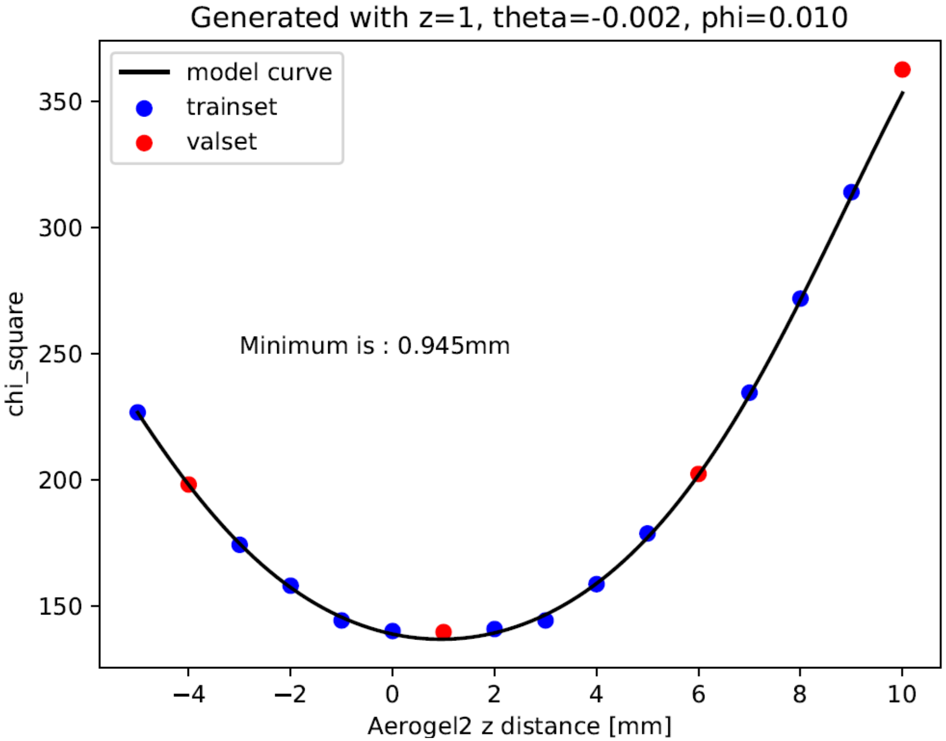
Questions?

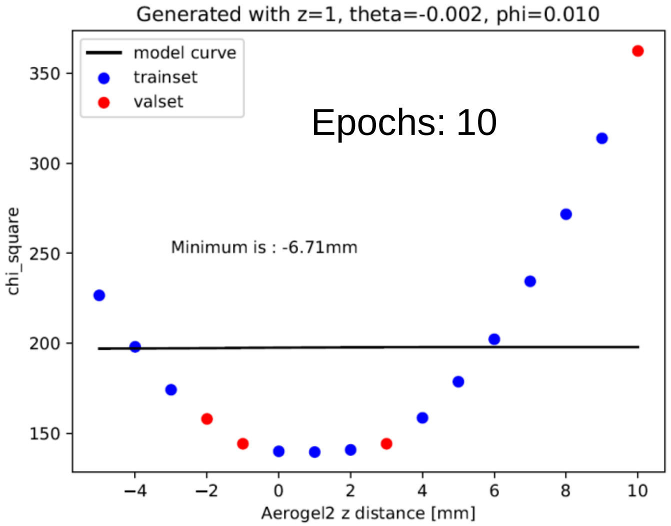
Second Aerogel z-distance alignment comparison

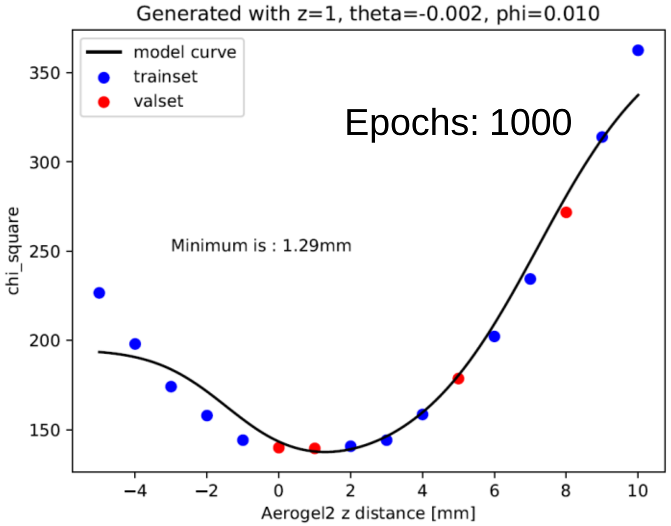
Polynomial fit results

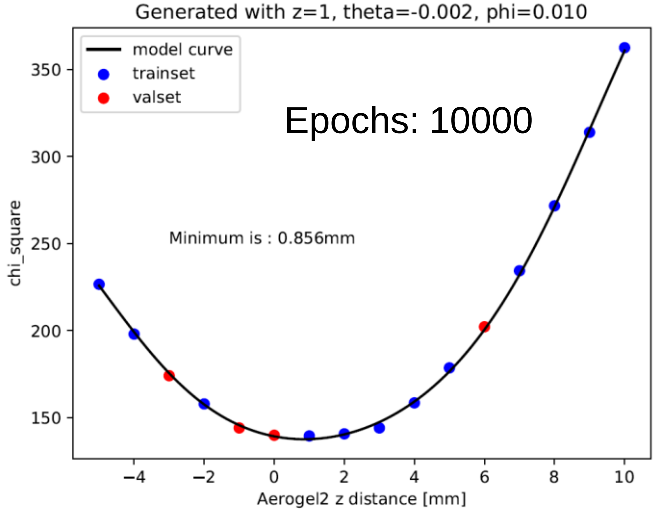


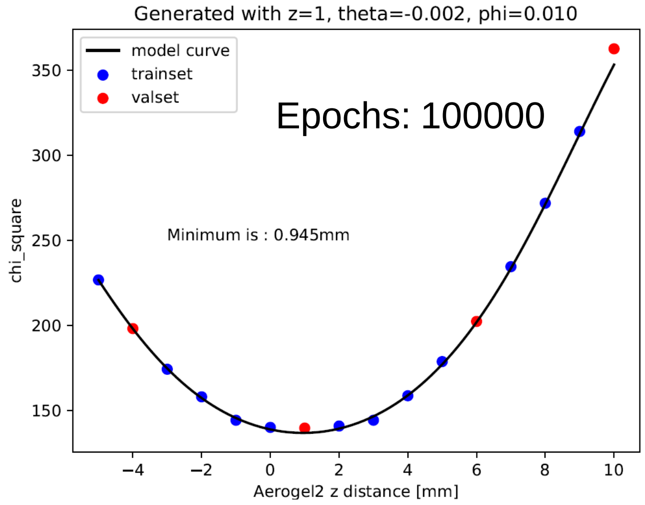
Machine learning results




Generated with z=1, theta=-0.002, phi=0.010
model curve
trainset
valset
Epochs: 10
Minimum is : -6.71mm
chi_square
Aerogel2 z distance [mm]

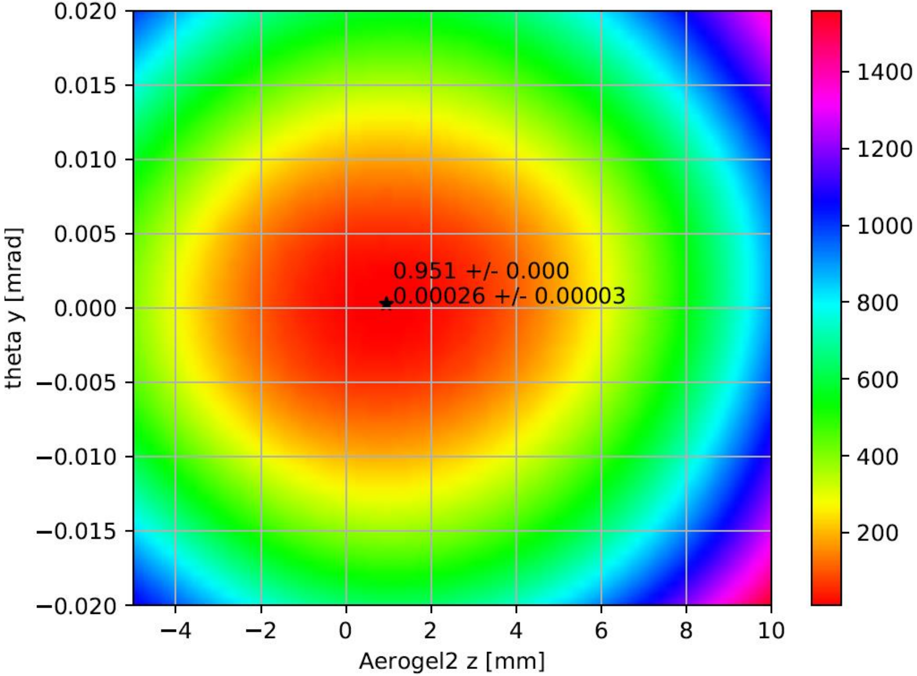

Generated with z=1, theta=-0.002, phi=0.010
model curve
trainset
valset
Epochs: 1000
Minimum is : 1.29mm
chi_square
Aerogel2 z distance [mm]


Generated with z=1, theta=-0.002, phi=0.010
model curve
trainset
valset
Epochs: 10000
Minimum is : 0.856mm
chi_square
Aerogel2 z distance [mm]

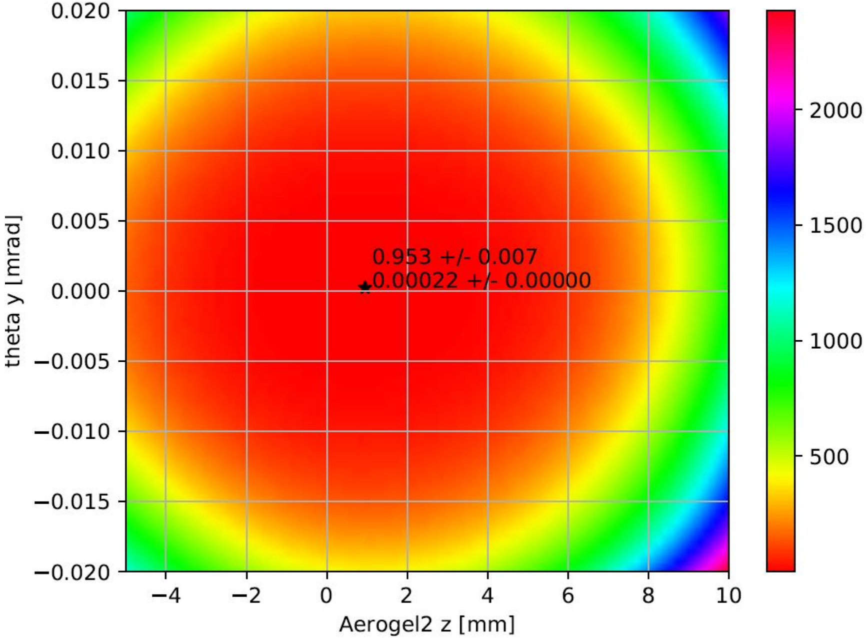

Generated with z=1, theta=-0.002, phi=0.010
model curve
trainset
valset
Epochs: 100000
Minimum is : 0.945mm
chi_square
Aerogel2 z distance [mm]

Comparison of RMS and sum of chi-squares

RMS of Chi-squares



Sum of Chi-squares



Comparison of real vs generated photons

RGA electron tracks
Nrefl = 0