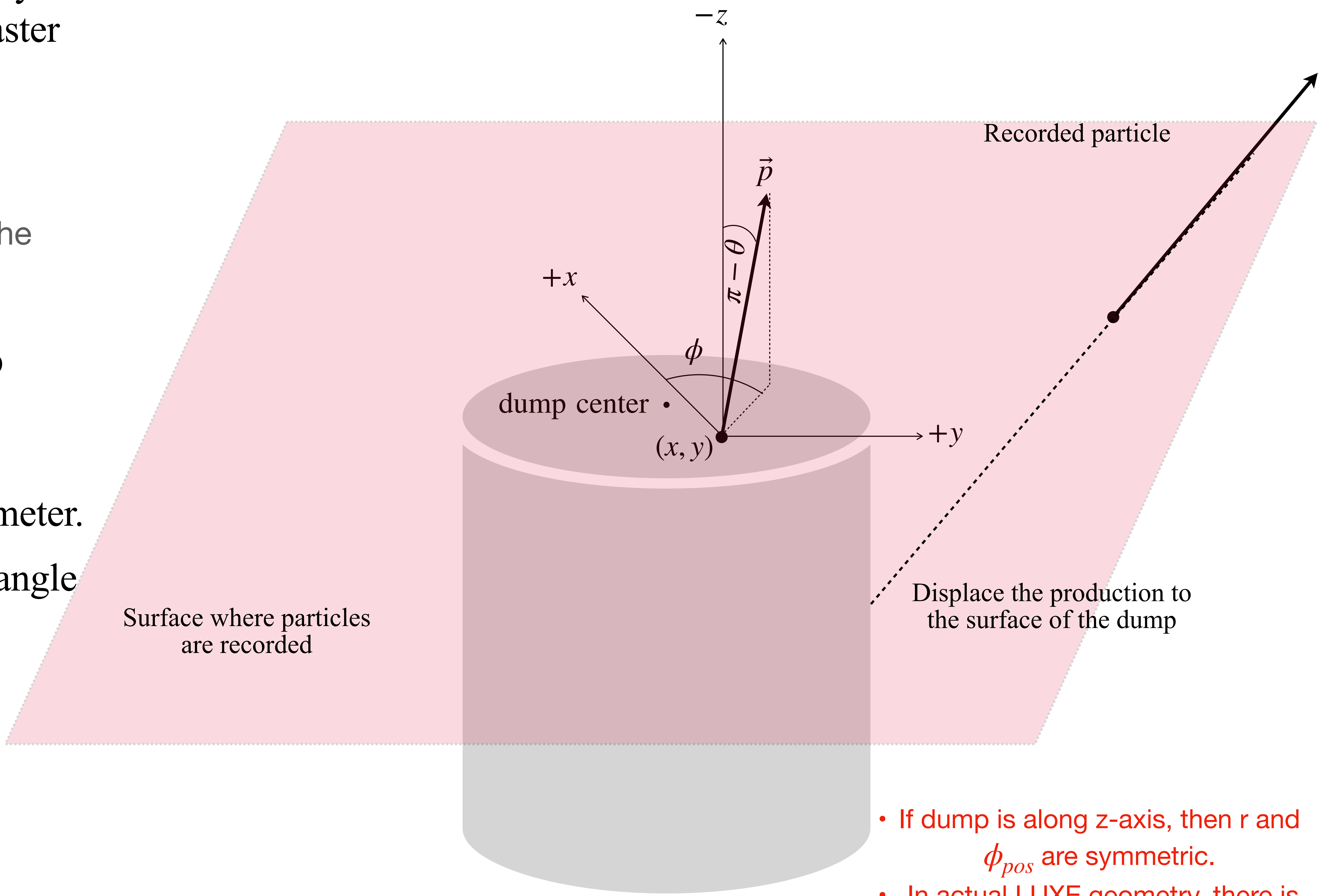
# FastSim Parametrization of Beam Dump using Generative Adversarial Network (GAN)

Oleksandr Borysov, Alon Levi, Arka Santra, Noam Tal Hod
Oct 30, 2023,
LUXE Software and Analysis Meeting,

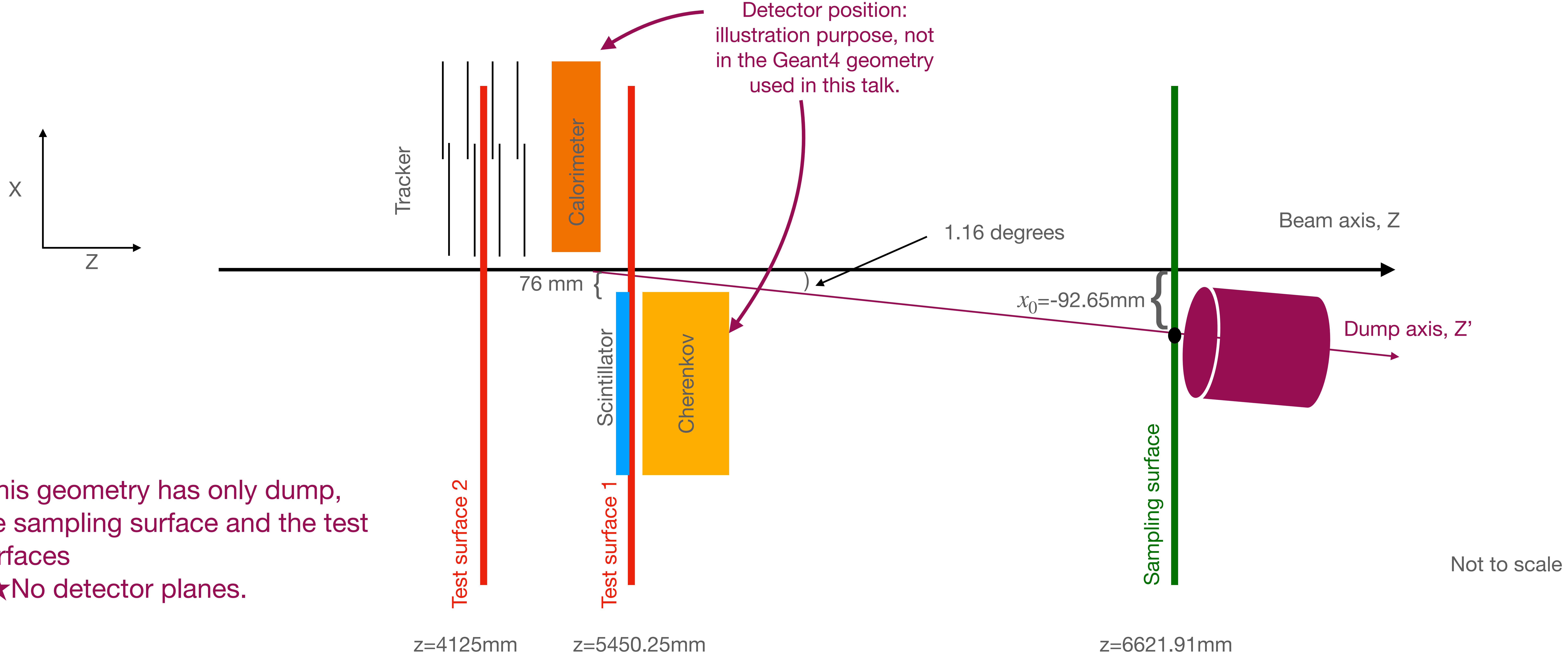
# Fast simulation strategy

- Disable the dump and replace all its “output” by source-like particles -> computation will be faster by at least 1 order of magnitude.
  - ★At the tracker last layer
    - ★100% of the neutron come from dump
    - ★>10% of the photons come from dump
    - ★~1% of the electrons and positrons from the dump
- Look at the dummy volume (sampling plane) located just outside of the surface of the dump
  - plot $(\frac{dN}{dE}$ and $\frac{dN}{dt})$ or $\frac{d^2N}{dEdt}, \frac{d^2N}{drd\theta_p}, \frac{d^2N}{d\phi_p d\phi_{pos}}, \ldots$
    - Here r is just $\sqrt{x^2+y^2}$ , position parameter.
    - $\theta_p, \phi_p$ is the polar angle and azimuthal angle of the momentum → representative of direction of the particle.
    - $\phi_{pos}$ is position azimuthal angle, and t is time.
  - Later: generate from the sampling plane according to these plots
    - ⋆ Use **Generative Adversarial Network**.




- If dump is along z-axis, then r and $\phi_{pos}$ are symmetric.
- In actual LUXE geometry, there is a shift - r and $\phi_{pos}$ are asymmetric there.

# Schematic diagram of the dump in the LUXE geometry




★This geometry has only dump, the sampling surface and the test surfaces

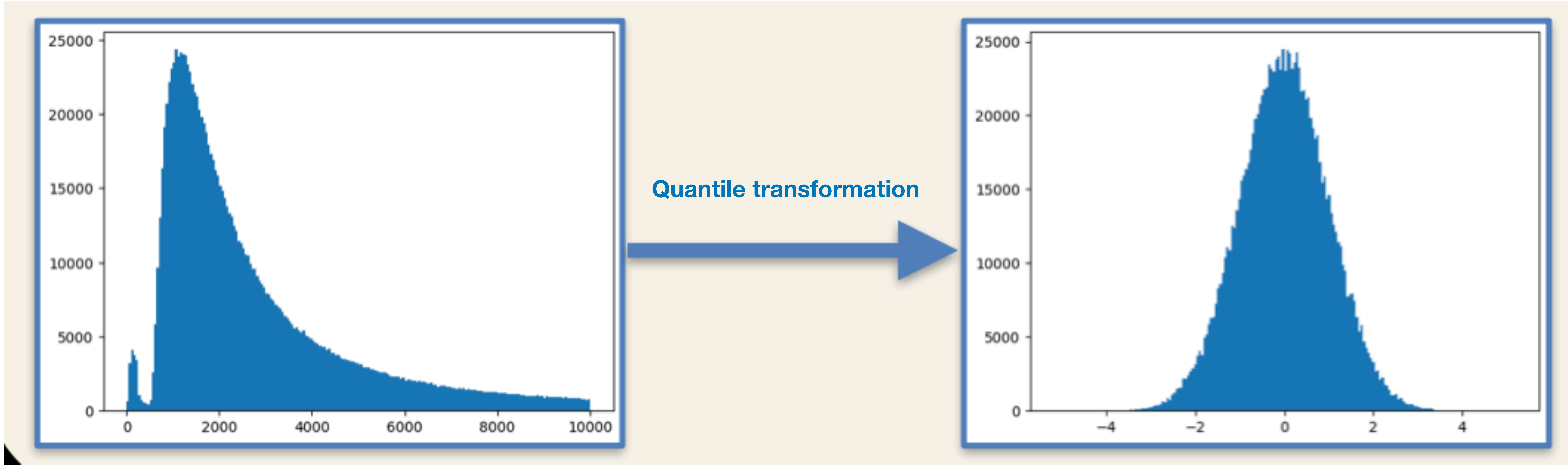
★No detector planes.

# Generative Adversarial Network (GAN)

# Generative Adversarial Network architecture

## Steps to achieve our goal:

- **Step 1:** Quantile transformation: This is done to "Gaussianize" the Geant4 dump distributions.
  - Each variable distribution has its own quantile transformation.
  - This is to **smoothen sharp features in distributions**, making it easier for the network to learn.




r [mm]

Transformed r

Plots prepared by Alon Levi

# Generative Adversarial Network architecture

## Steps to achieve our goal:

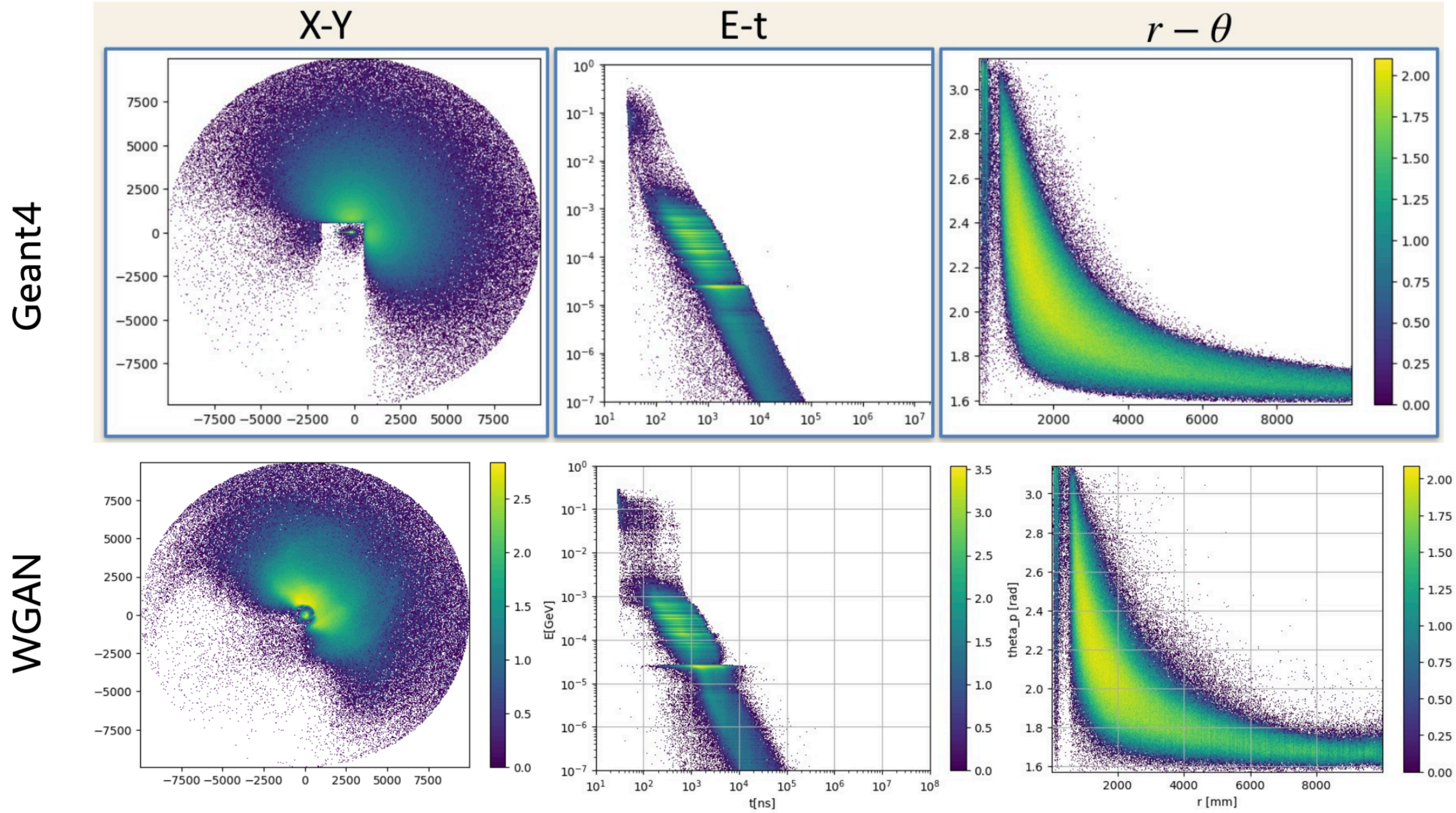
- **Step 2:** Generator: tries to generate meaningful distributions from random noise.
  - Consists of 4 hidden layers, LeakyReLU as the activation function.
  - *Batch Normalization* in each layer.
- **Step 3:** Discriminator: given the original distribution, it tells us how good the generation in Step 2 was.
  - Consists of 4 hidden layers, LeakyReLU as the activation function.
  - *Instance Normalization* in each layer.
- **Step 4:** the Loss function:

$$\mathscr{L}(p_r, p(z)) = \max_{w \in W, \theta \in \Theta} \Big[ \underbrace{\mathbb{E}_{x \sim p_r}[f_w(x)]}_{\text{Expectation value from the original distribution}} - \underbrace{\mathbb{E}_{x \sim p(z)}[f_w(g_\theta(z))]}_{\text{Expectation value from the generated distribution}} \Big]$$ —> Wasserstein distance

- **Step 5:** Inverse quantile transformation to get physical distribution.

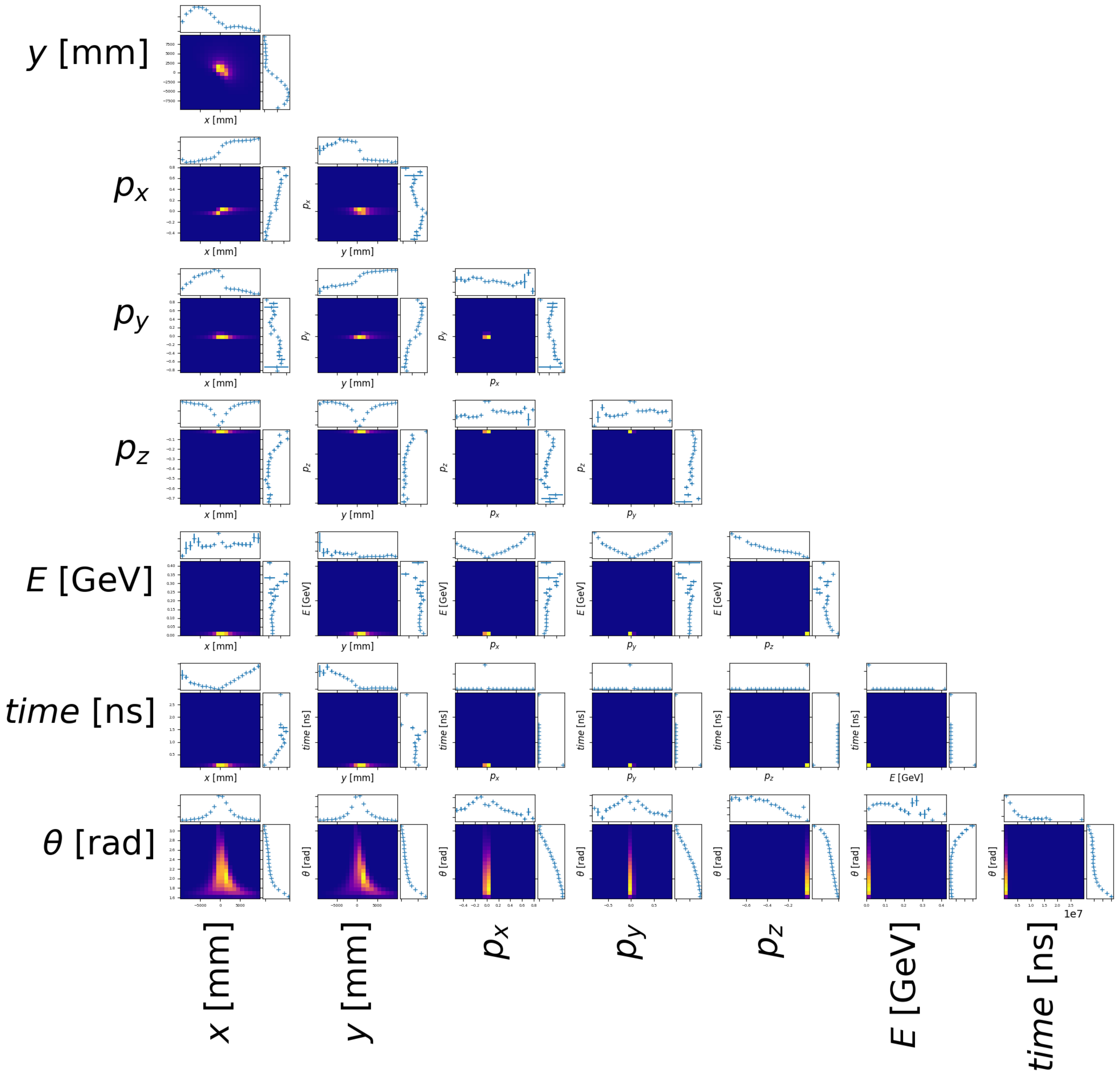
# Generating background using WGAN

Generative Adversarial Network (GAN) with Wasserstein loss function




**neutron at sampling surface**

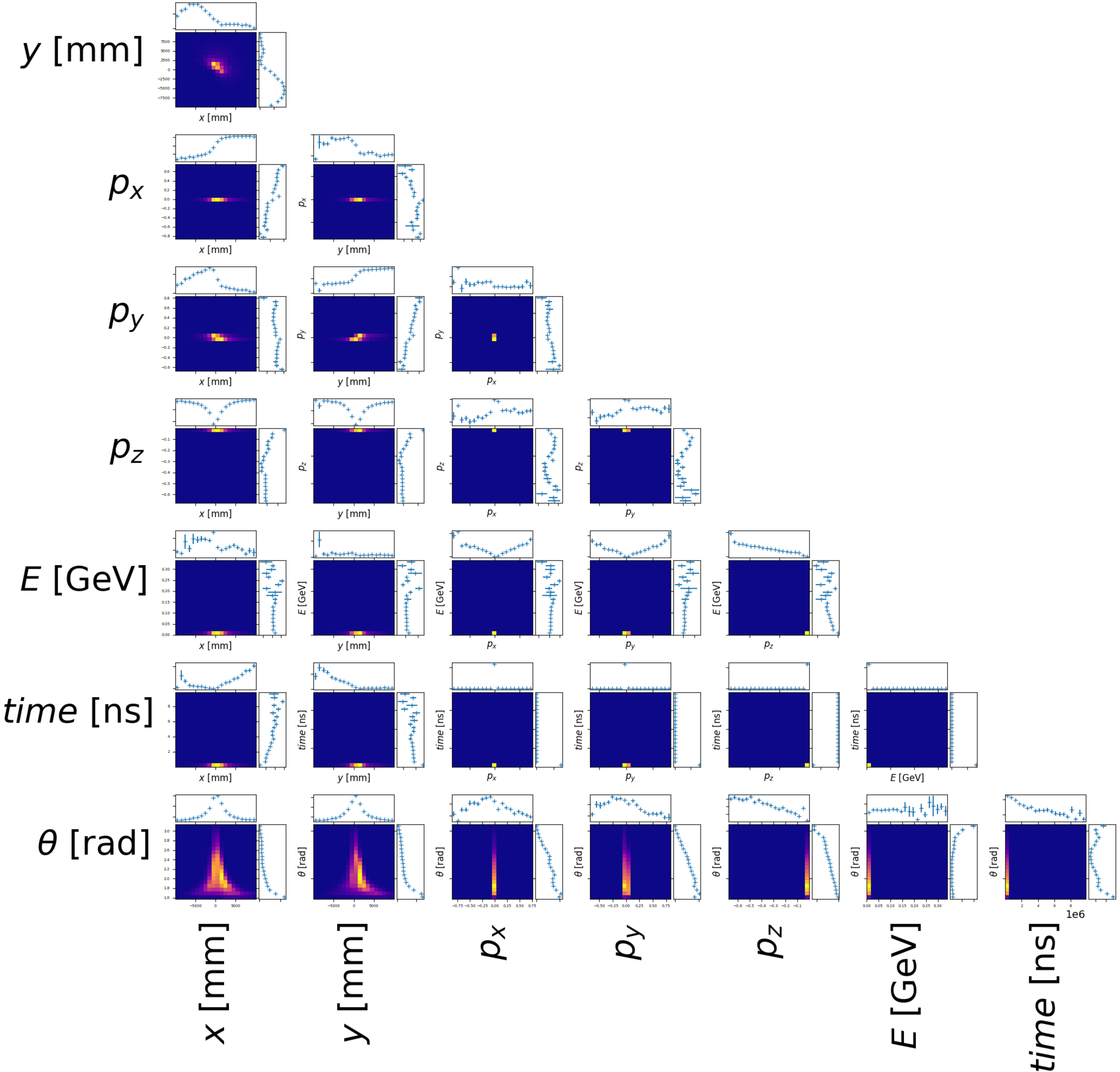
Plots prepared by Alon Levi




# All variables correlation plots: From Geant4

**Neutron distributions**

Plots prepared by Alon Levi




# All variables correlation plots: From GAN distributions

**Neutron distributions**

Plots prepared by Alon Levi

# Comparison with original (Geant4) distributions

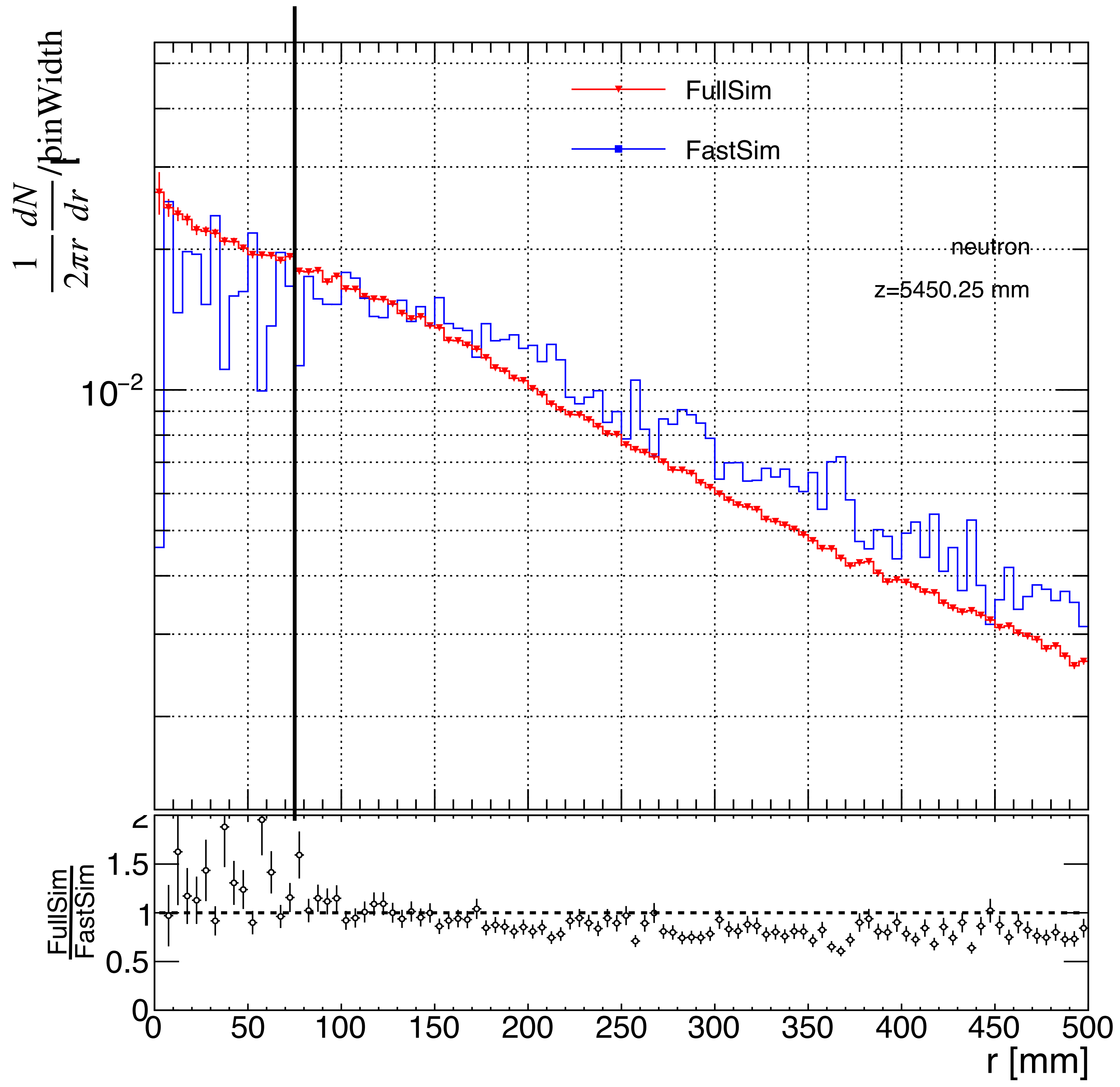
# Plot labels:

- In this talk there will be **two types of plots** compared:
    1. **FullSim** - Distributions from full Geant4 processing of the dump
    2. **FastSim** - Geant4 processing where dump is replaced by particles following GAN neural network.
        (i) Plots made at test surfaces.

# Neutron at test surface 1: r at very low value

## (Looking at the shape of the distribution)

The distance of scintillator screen from the beam pipe is ~76 mm.



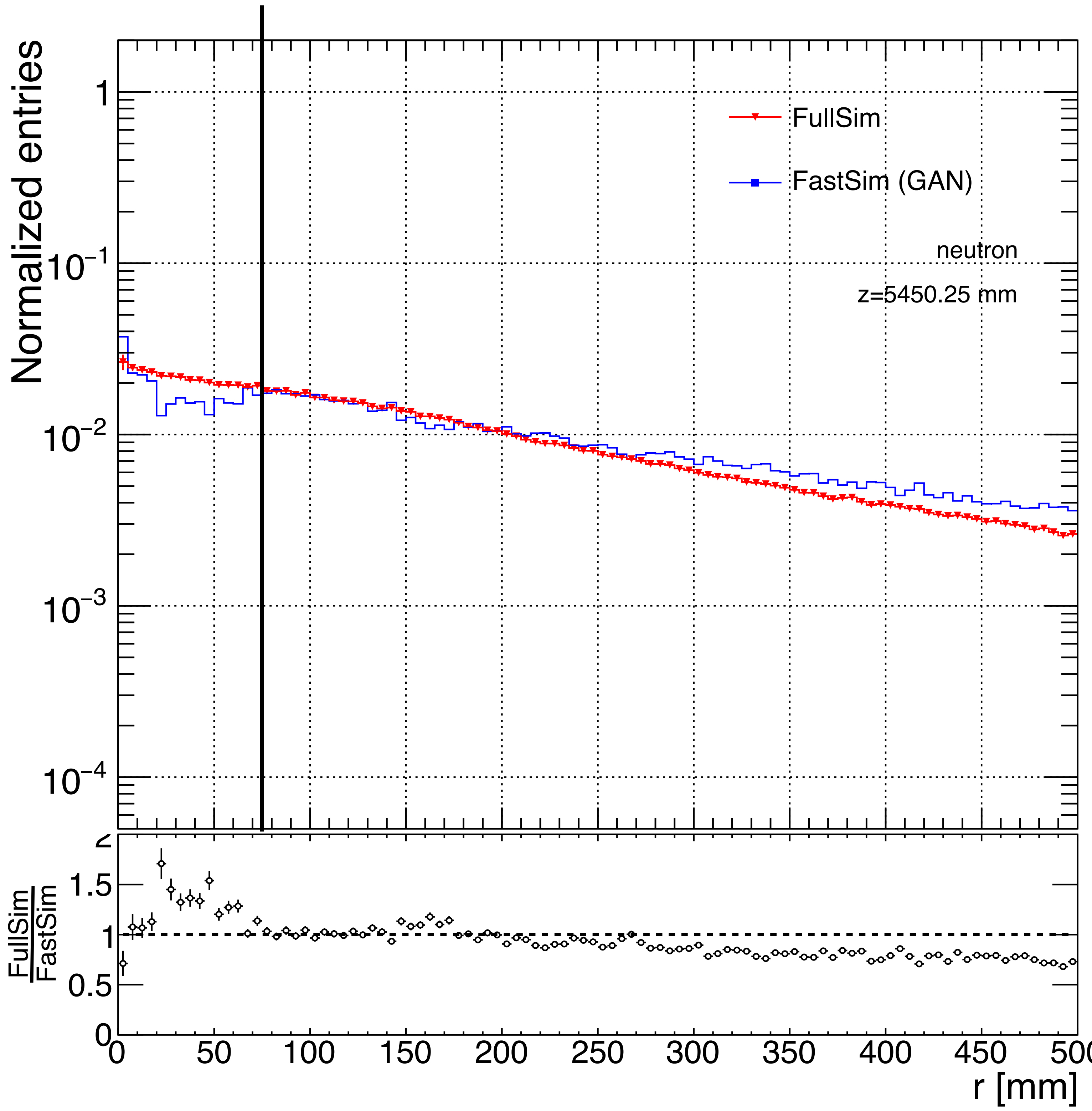

FastSim was generated from 2D distributions using ROOT::TH2D::GetRandom2()

FastSim r matches with FullSim after the application of the scale factor

**With scale factor on the sampling plot, FastSim has an ad-hoc weight to match the FullSim**

Shown at the July 10 SAS meeting

The distance of scintillator screen from the beam pipe is ~76 mm.




FastSim from GAN, considering all correlations

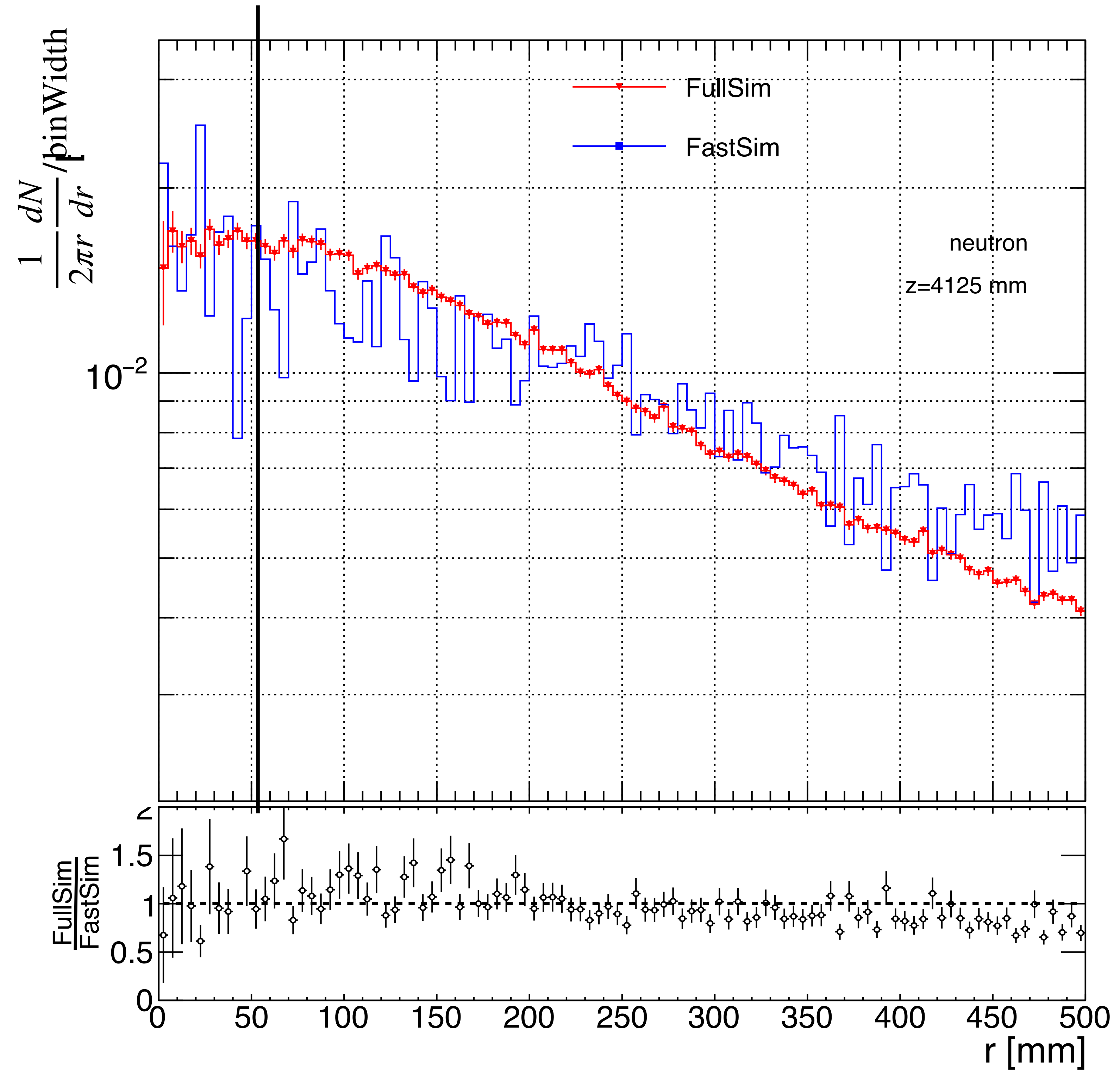
FastSim r matches with FullSim without any ad-hoc weight, natural distribution from GAN

**FastSim generated by GAN, no ad-hoc weight applied**

# Neutron at test surface 2: r at very low value

## (Looking at the shape of the distribution)

The distance of tracker inner layer from the beam pipe is ~52 mm.

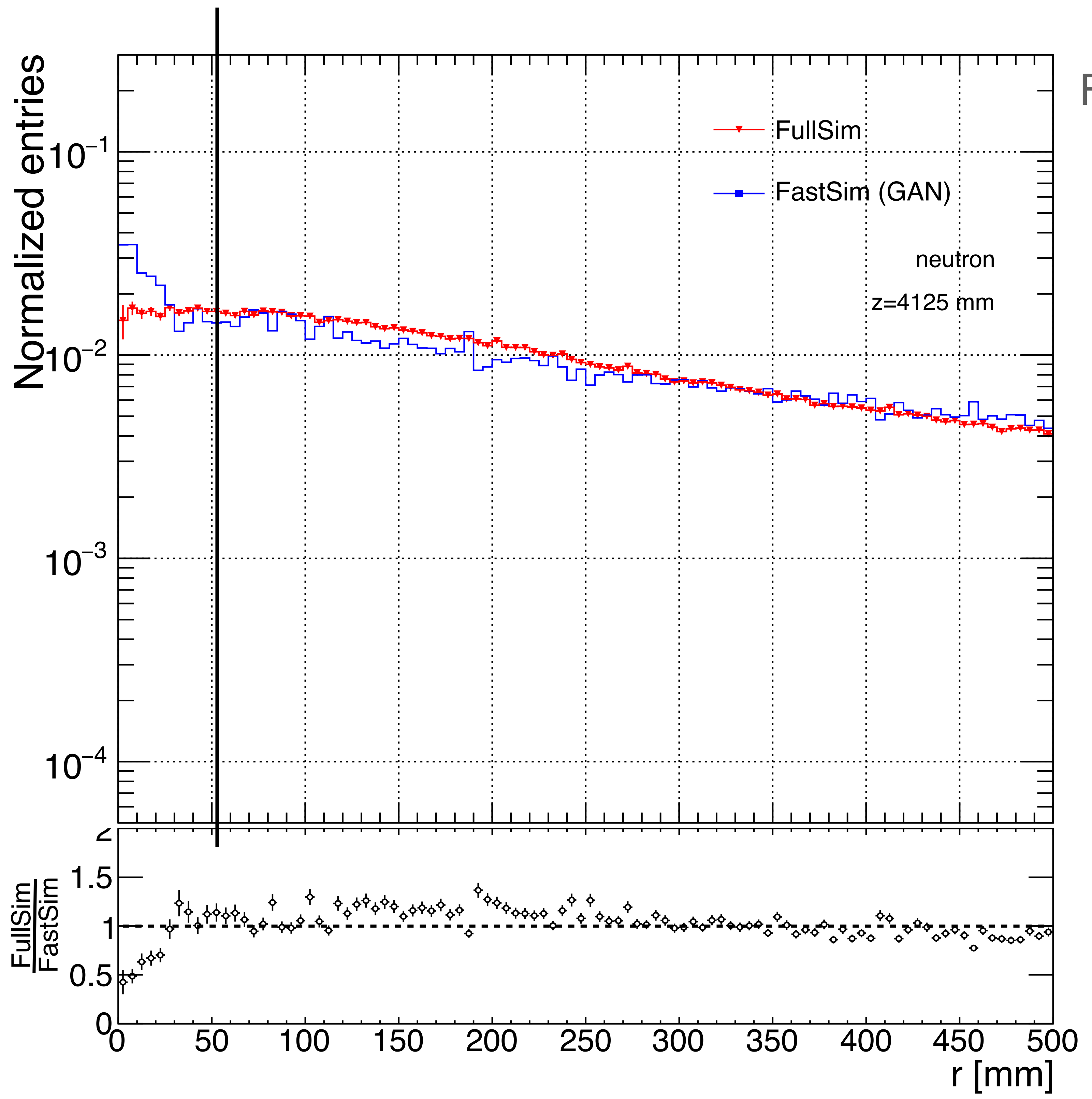



FastSim was generated from 2D distributions using ROOT::TH2D::GetRandom2()

FastSim r matches with FullSim after the application of the scale factor

**With scale factor on the sampling plot, FastSim has an ad-hoc weight to match the FullSim**

The distance of tracker inner layer from the beam pipe is ~52 mm.




FastSim from GAN, considering all correlations

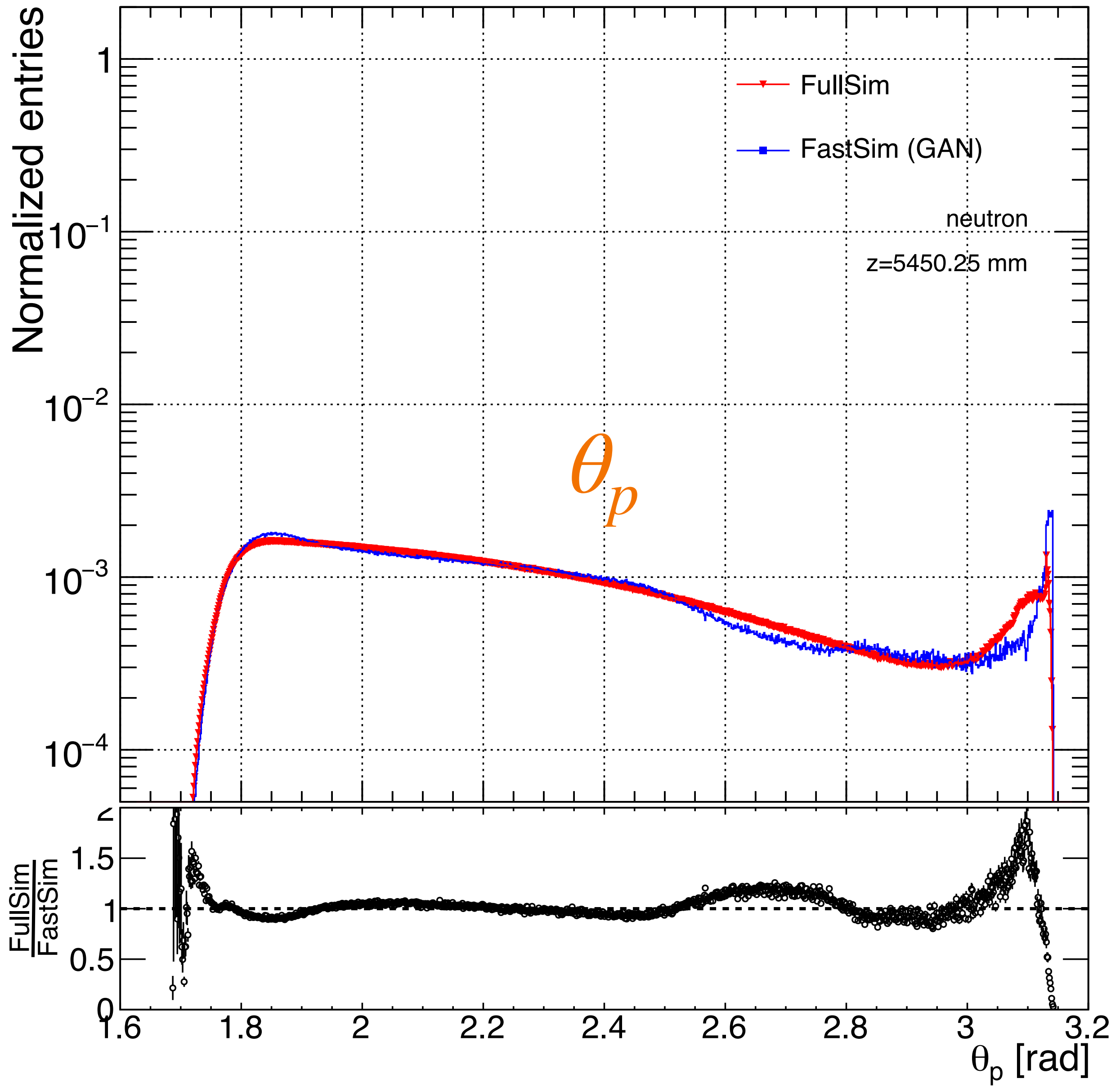
FastSim r matches with FullSim without any ad-hoc weight, natural distribution from GAN

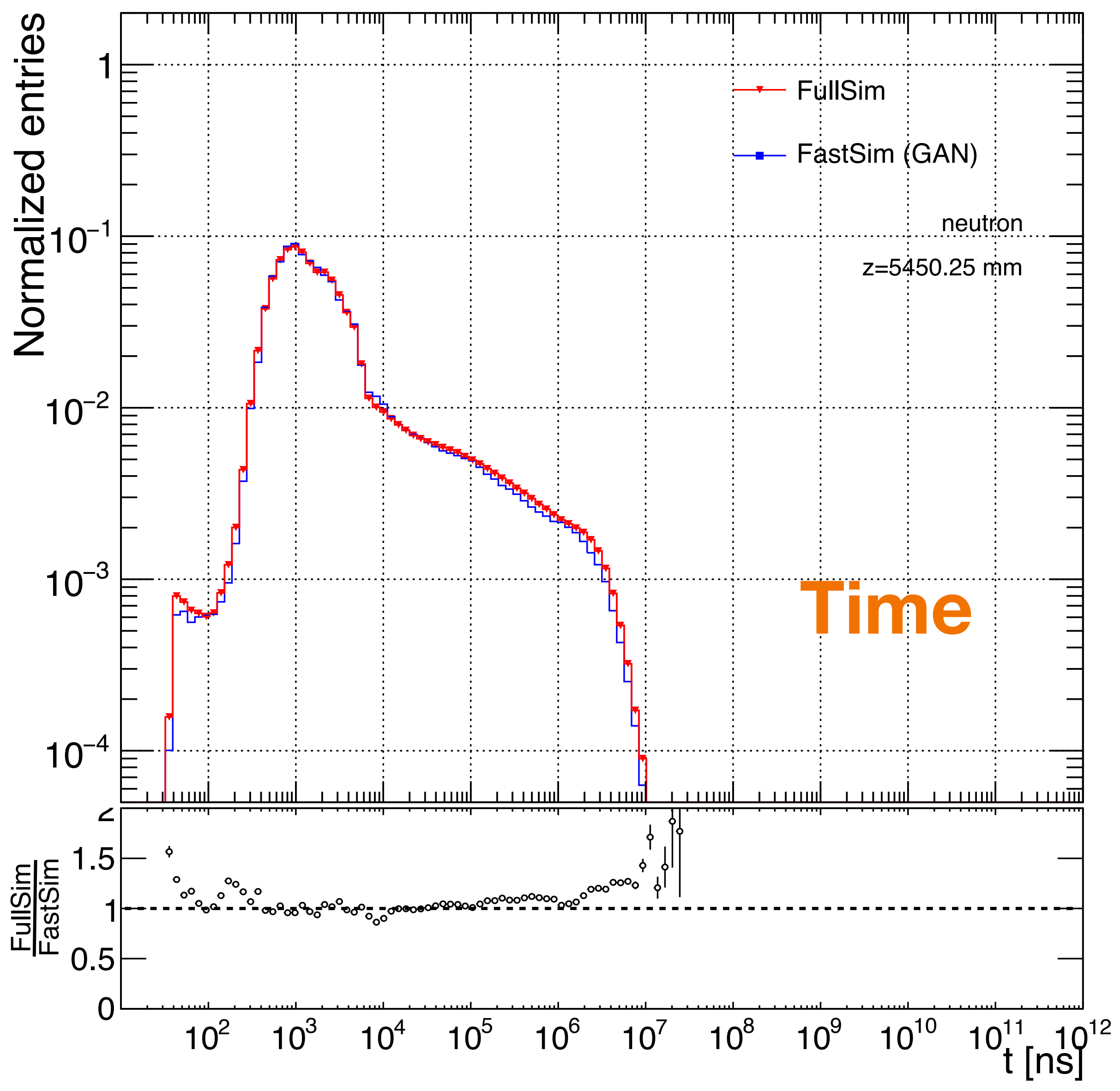
**FastSim generated by GAN, no ad-hoc weight applied**

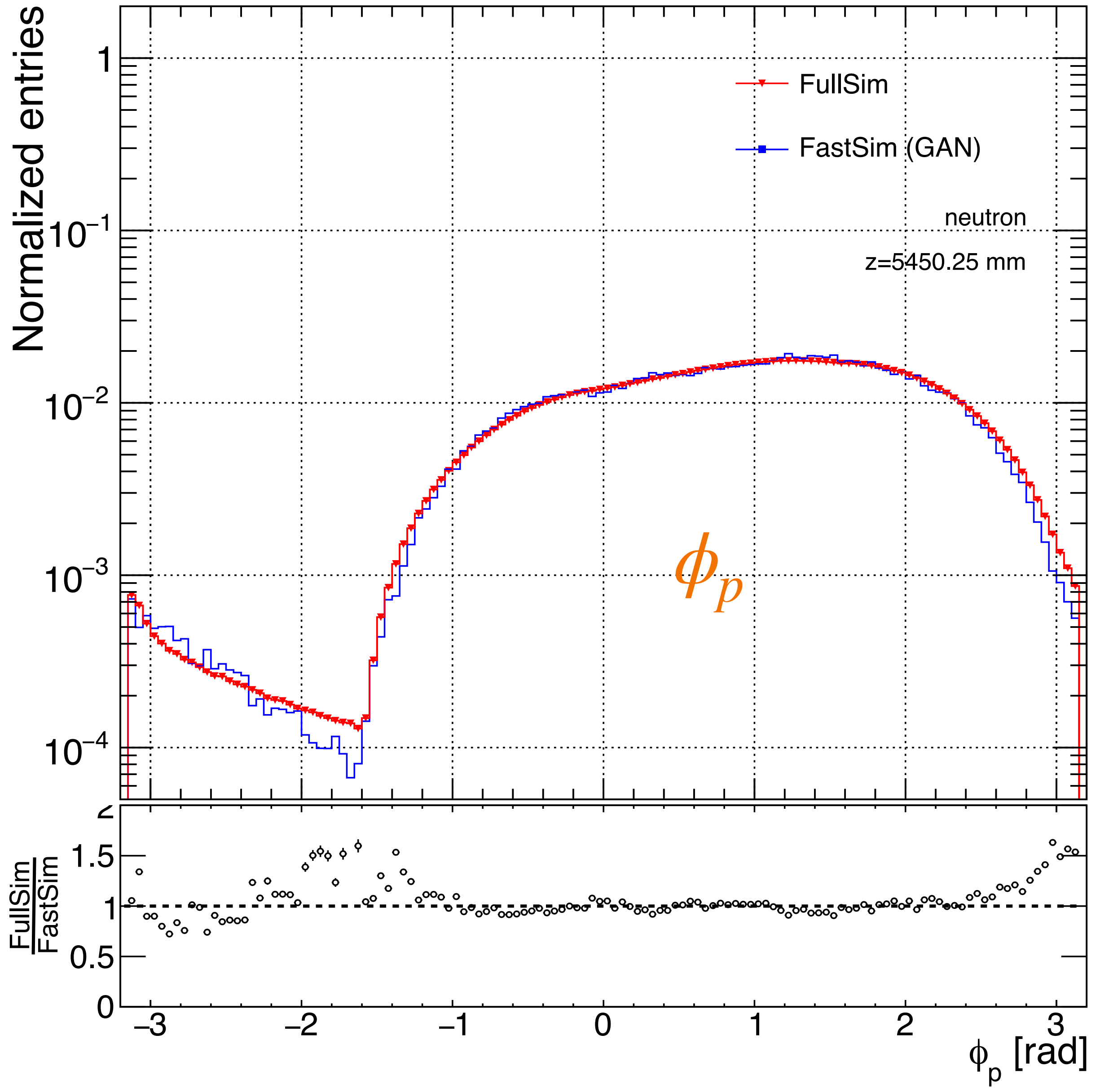
Shown at the July 10 SAS meeting

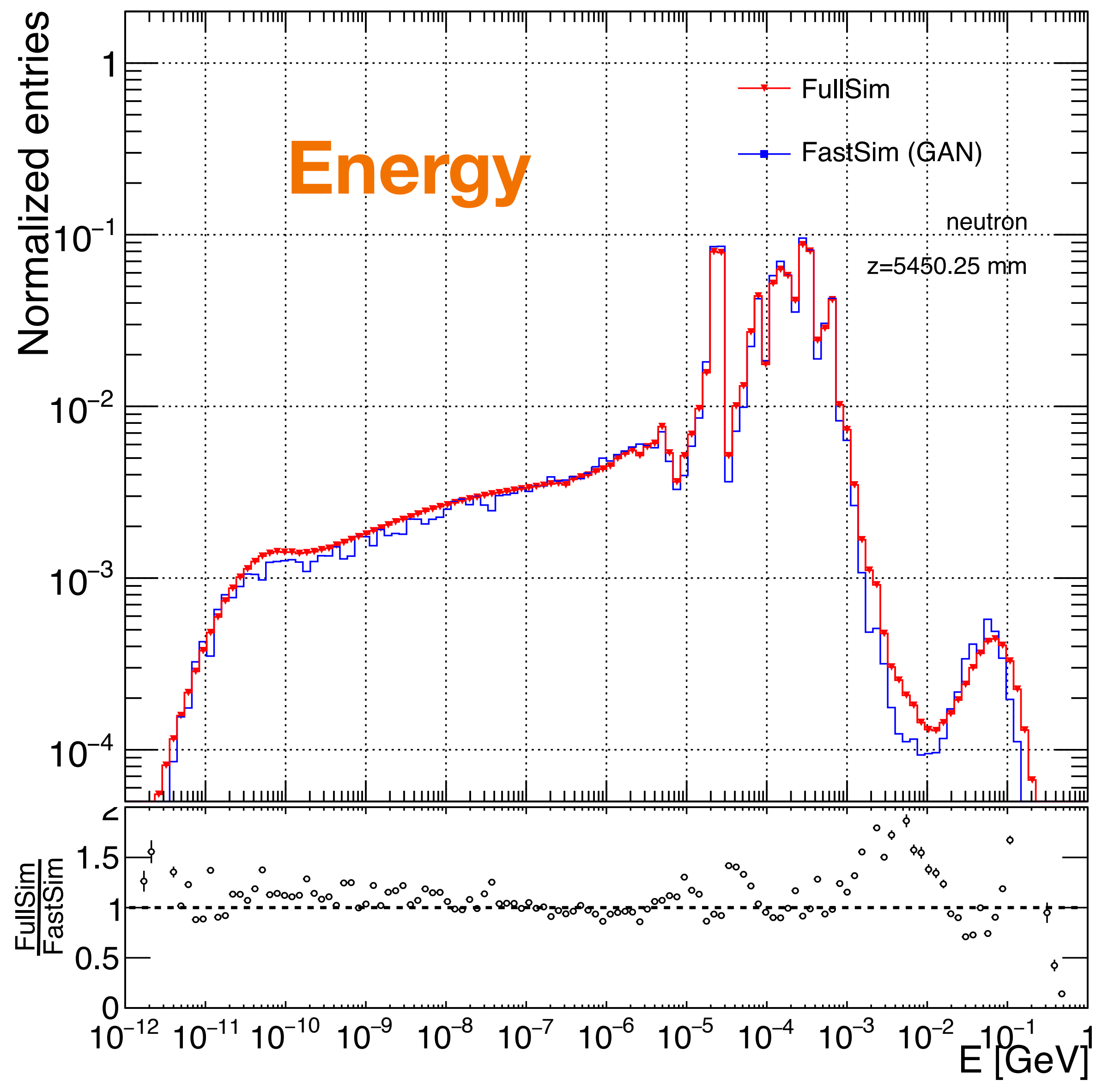
# Other distributions from FastSim (GAN)

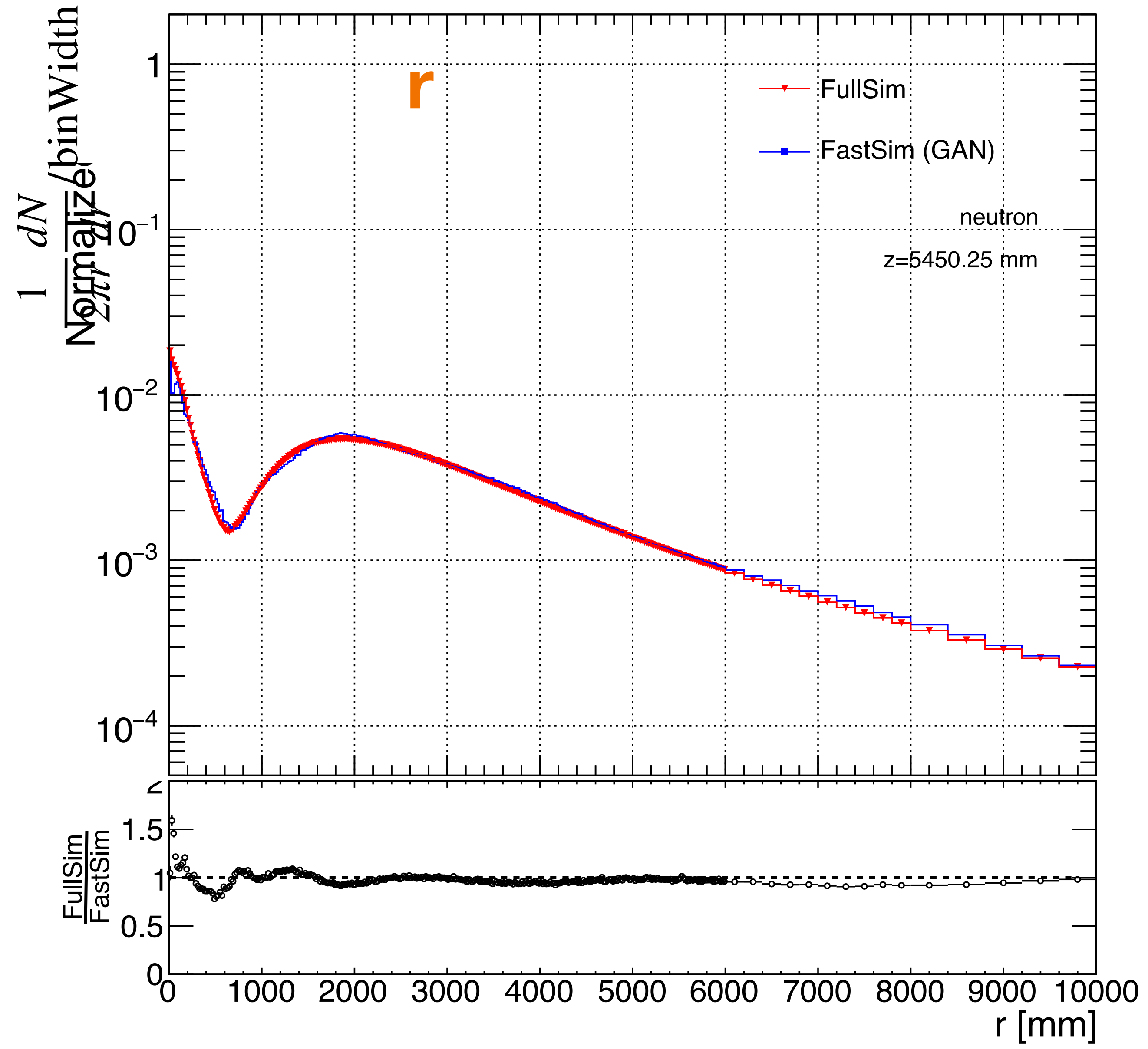
# A few 1D distributions between FastSim (GAN) and FullSim: neutron at the test surface 1



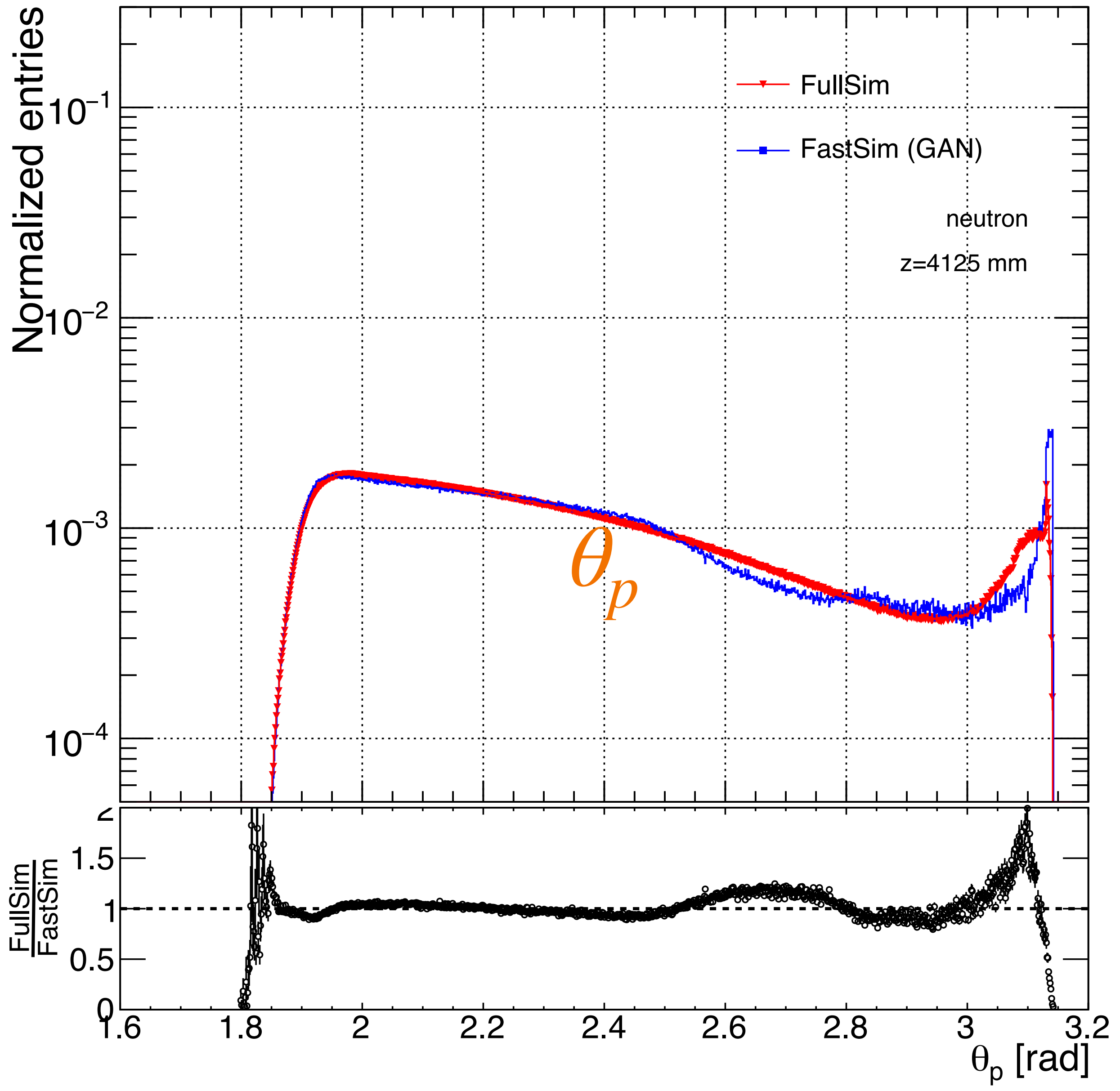



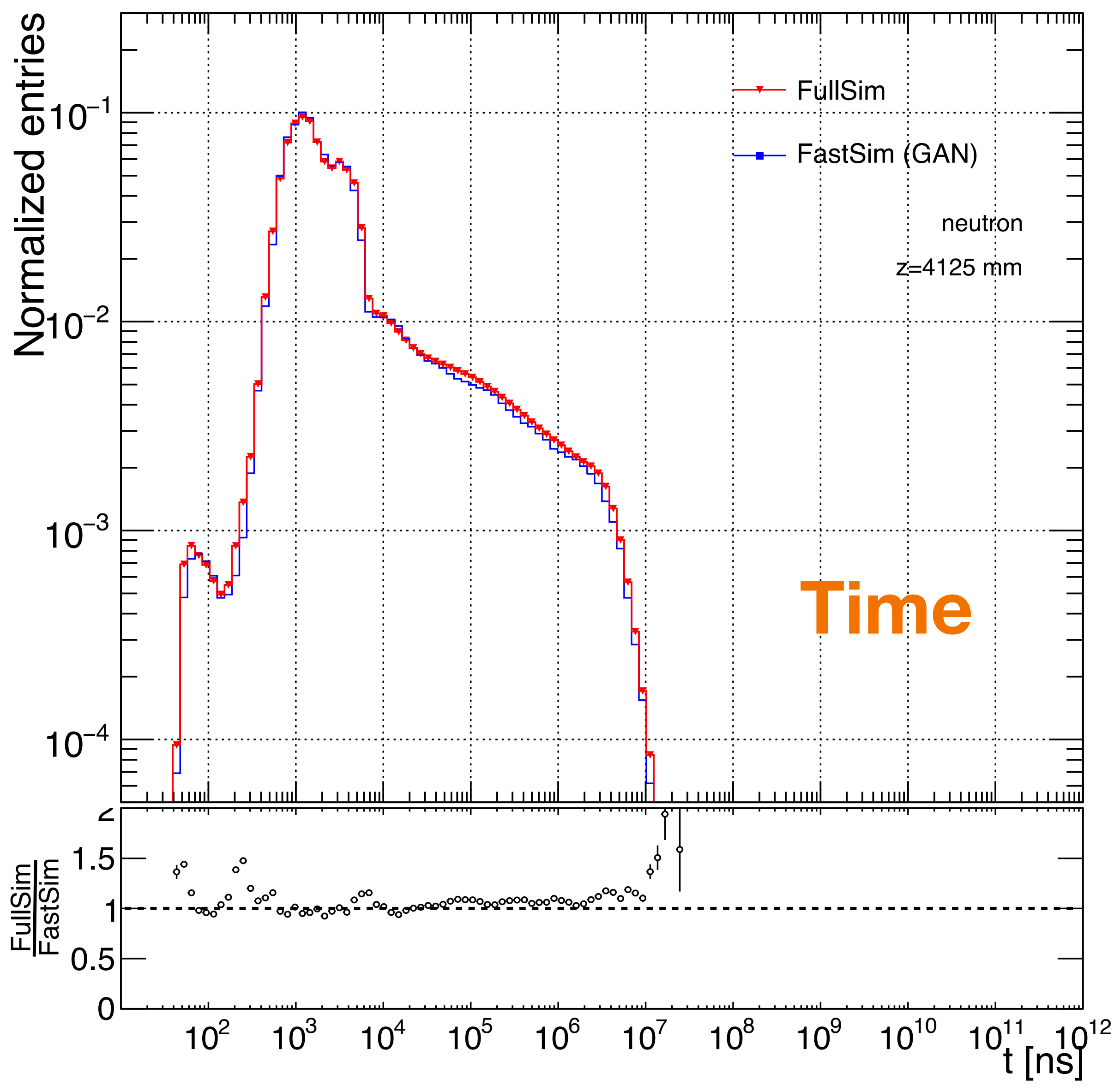



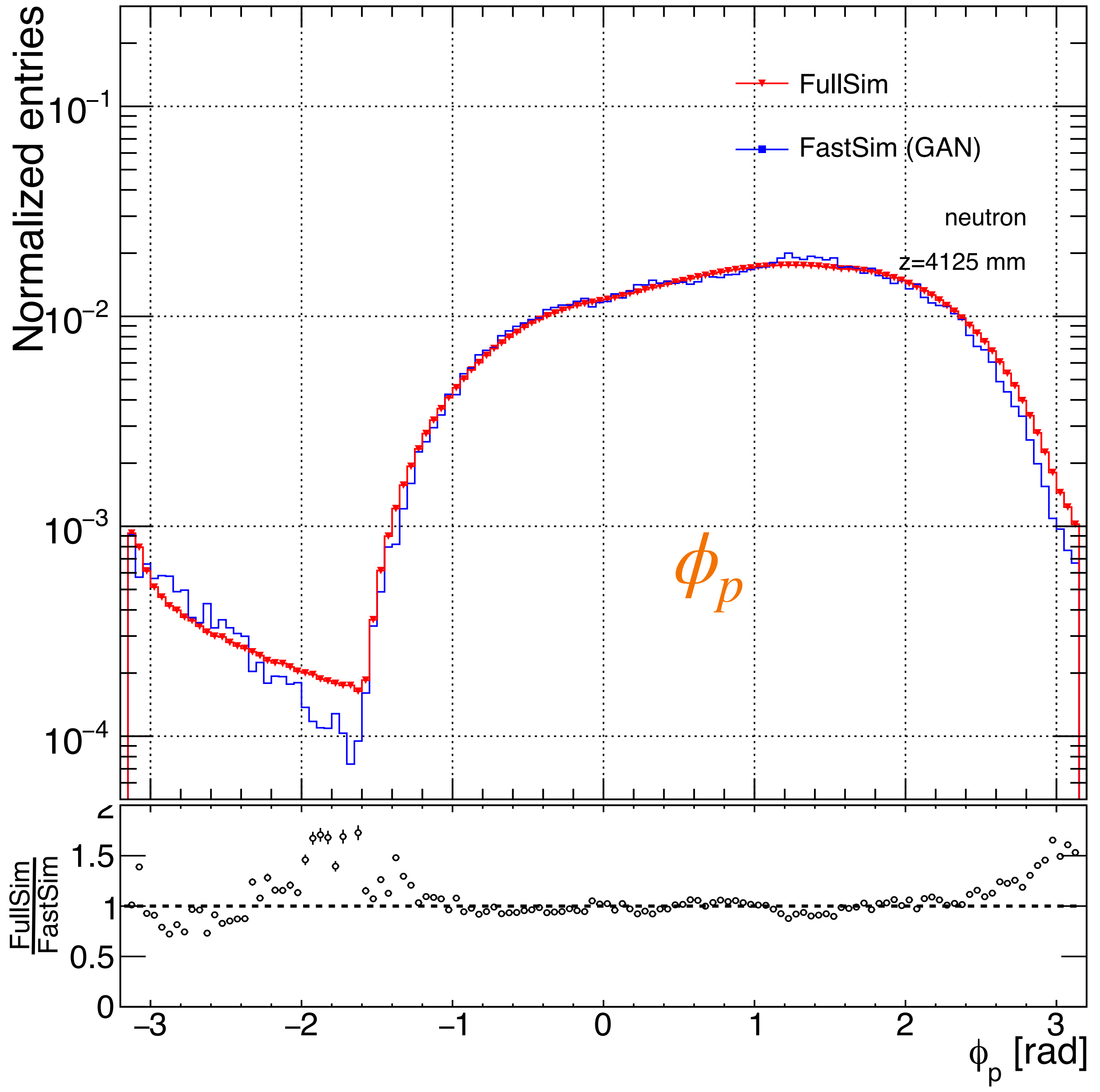



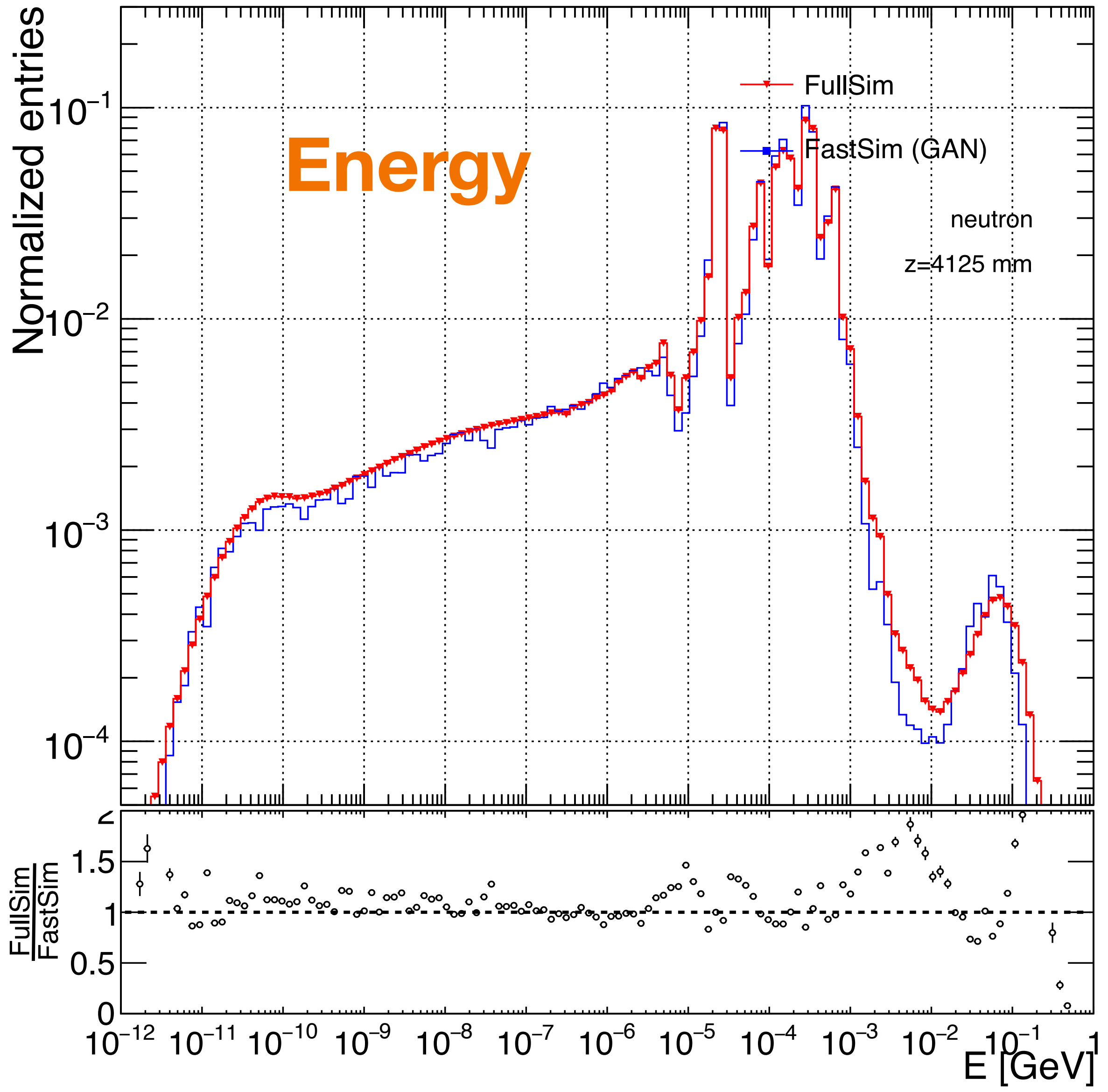

★Distributions are looked at **z=5450.25mm**.
★FullSim and FastSim distributions are matching quite well.

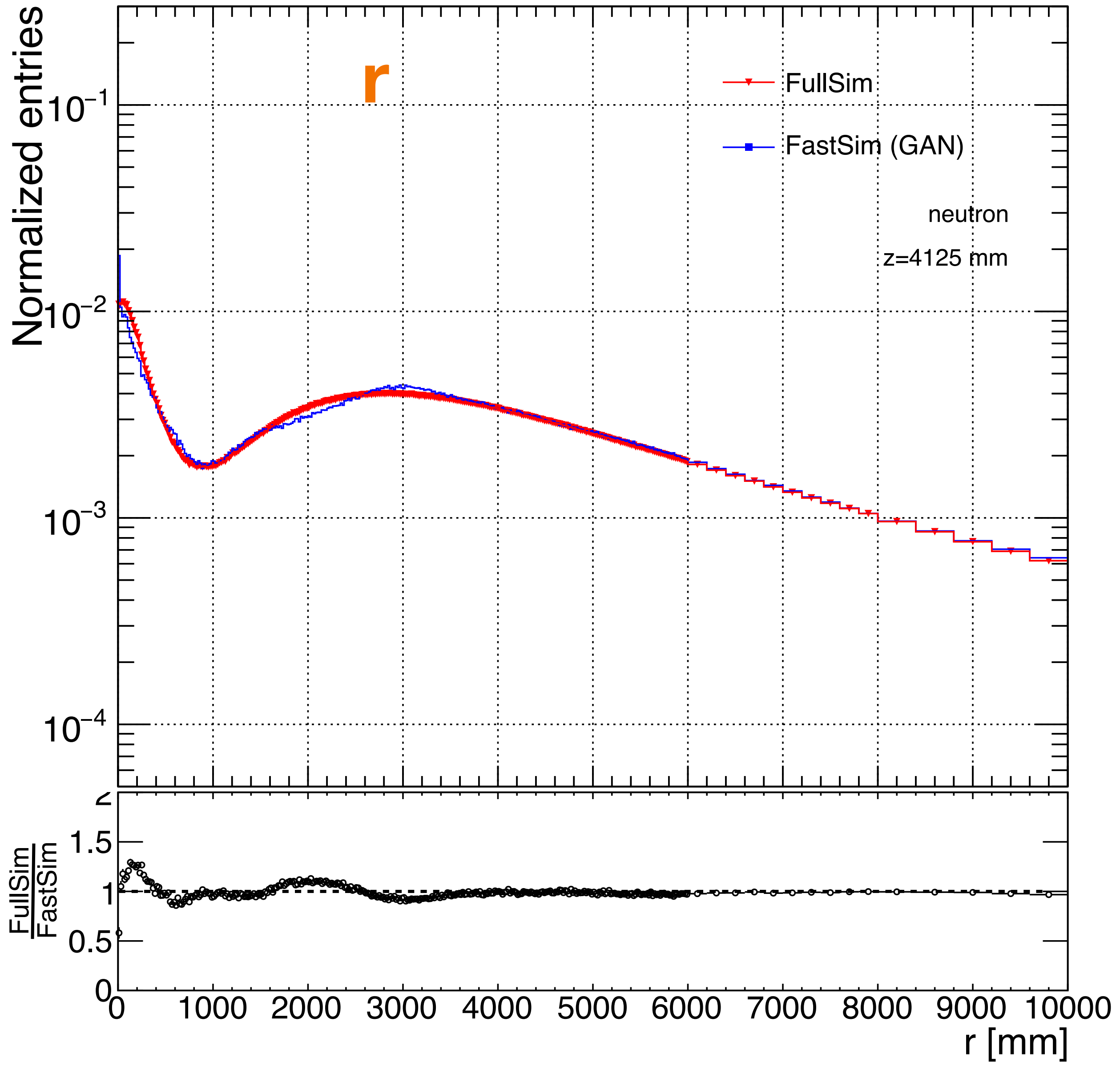
# A few 1D distributions between FastSim (GAN) and FullSim: neutron at the test surface 2




★Distributions are looked at **z=4125mm.**
★FullSim and FastSim distributions are matching quite well.

# Summary and Outlook

# Summary and next steps:

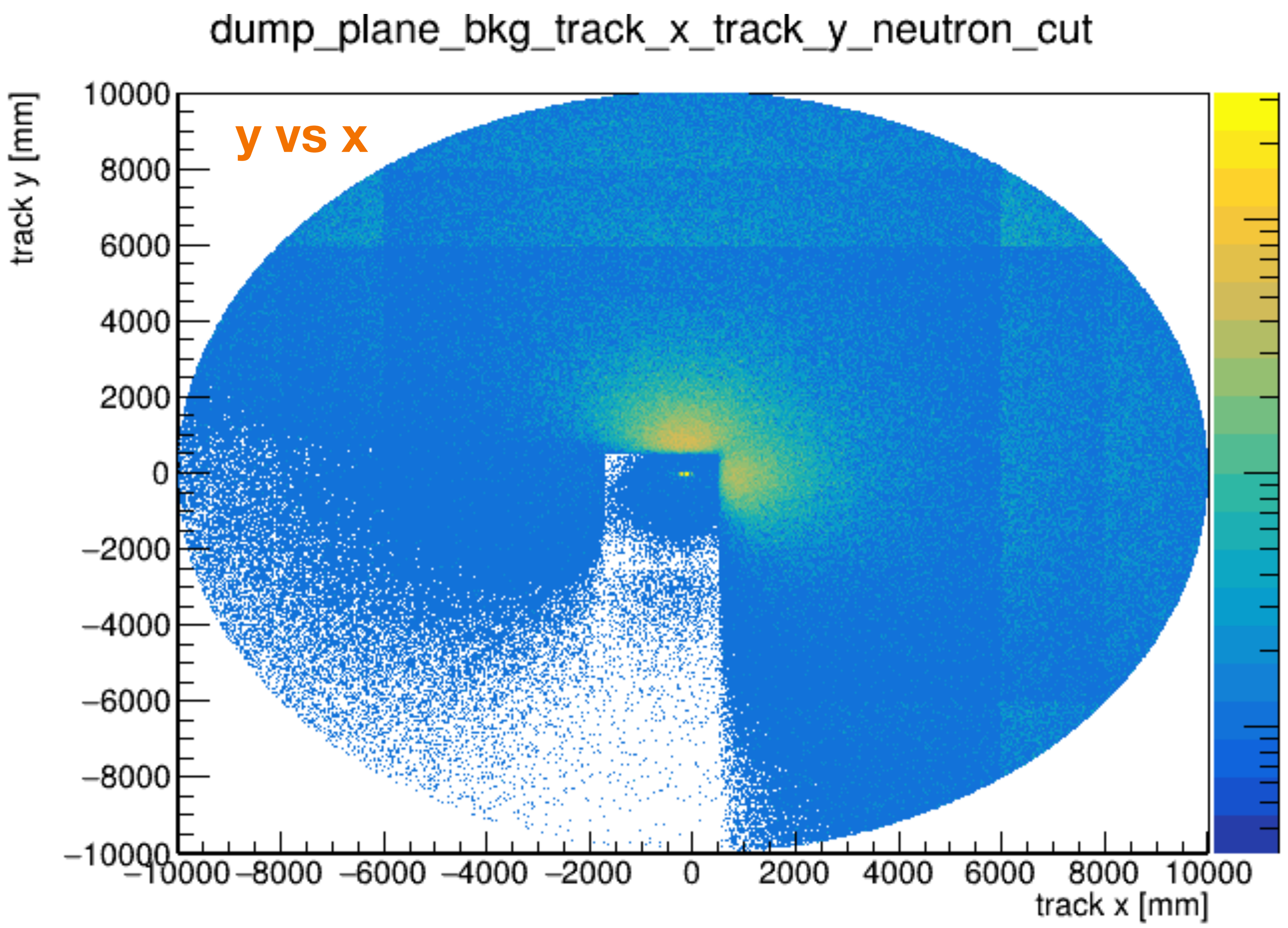
- **Mis-modeling in very backward particles**
  - The problem is being solved by using Generative Adversarial Network (GAN).
- **<u>Neutrons:</u>**
  - GAN FastSim sample gives similar distribution shapes of neutrons as FullSim in test surface 1 and test surface 2.
- Looking into GANs
  - More natural to parameterize the correlations.
    - Consider all variable correlations unlike 2D plots which take only 2 variable correlation at a time.
  - The result is not perfect (yet), but trends **going in the right direction** in mimicking the FullSim distribution.
- Worked only with neutron, but photon distributions will be looked at next.

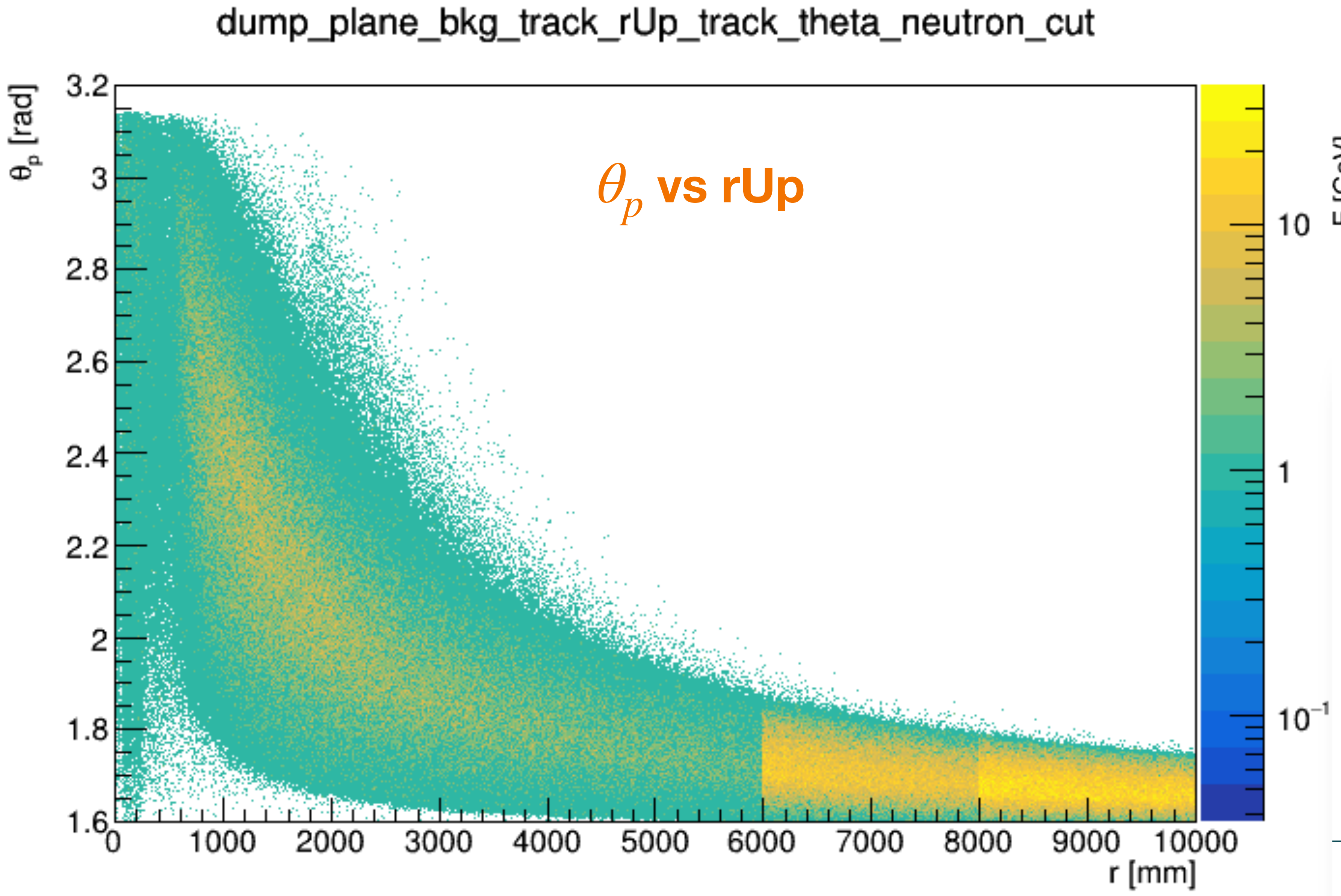
# Thank you!

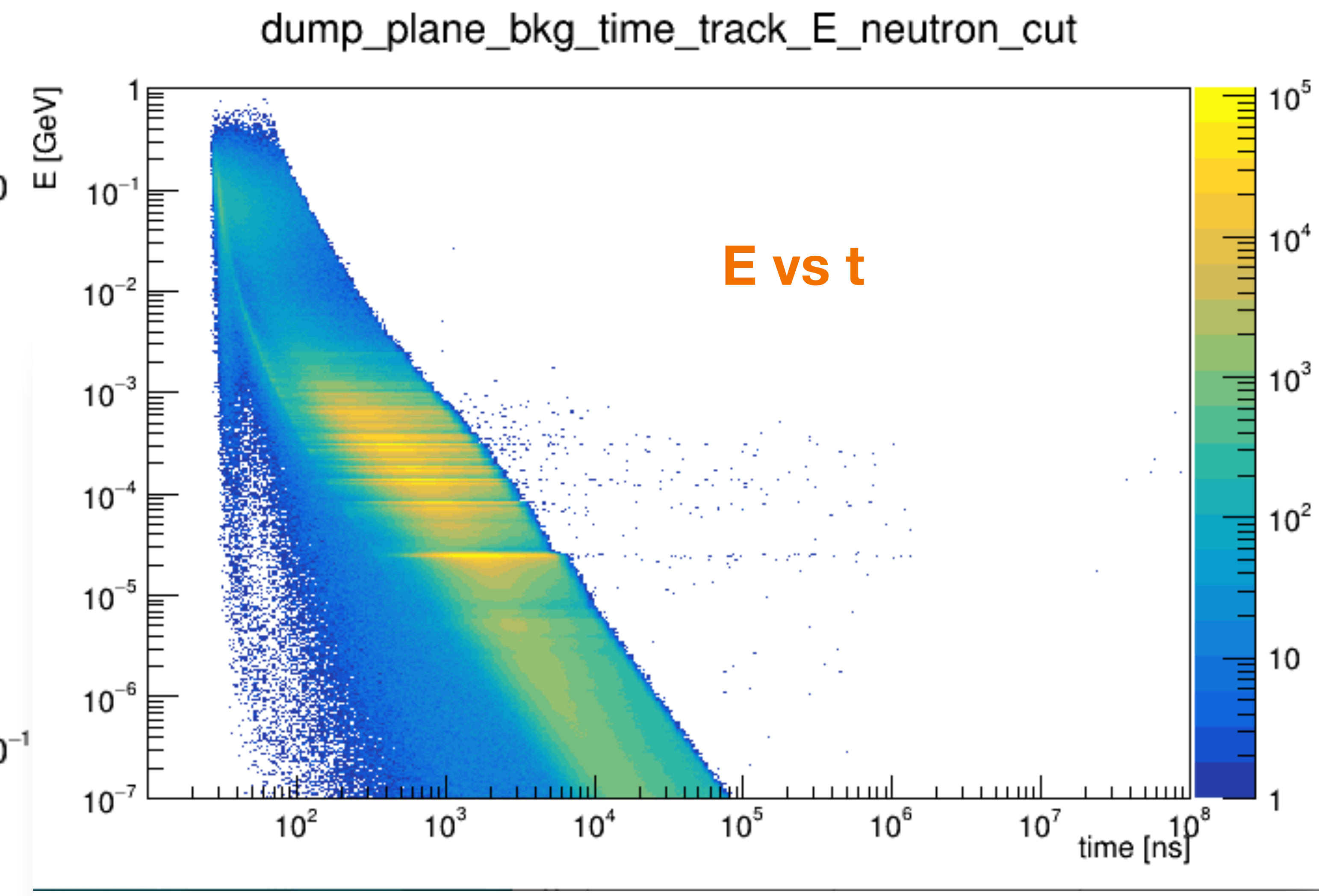
# Baseline distribution plots for LUXE geometry at the sampling surface (z=6621.91mm)

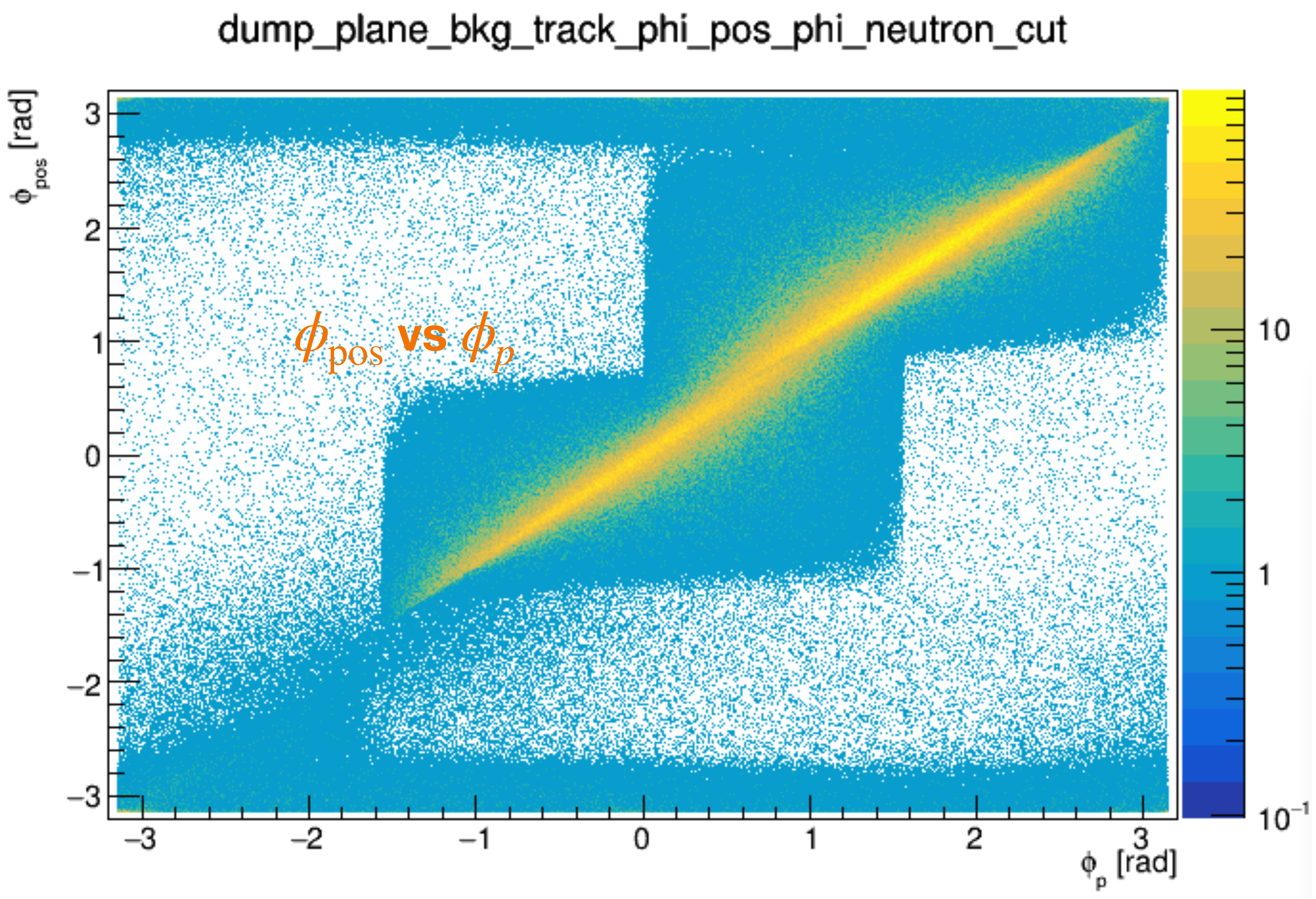
★ Plots used for sampling.

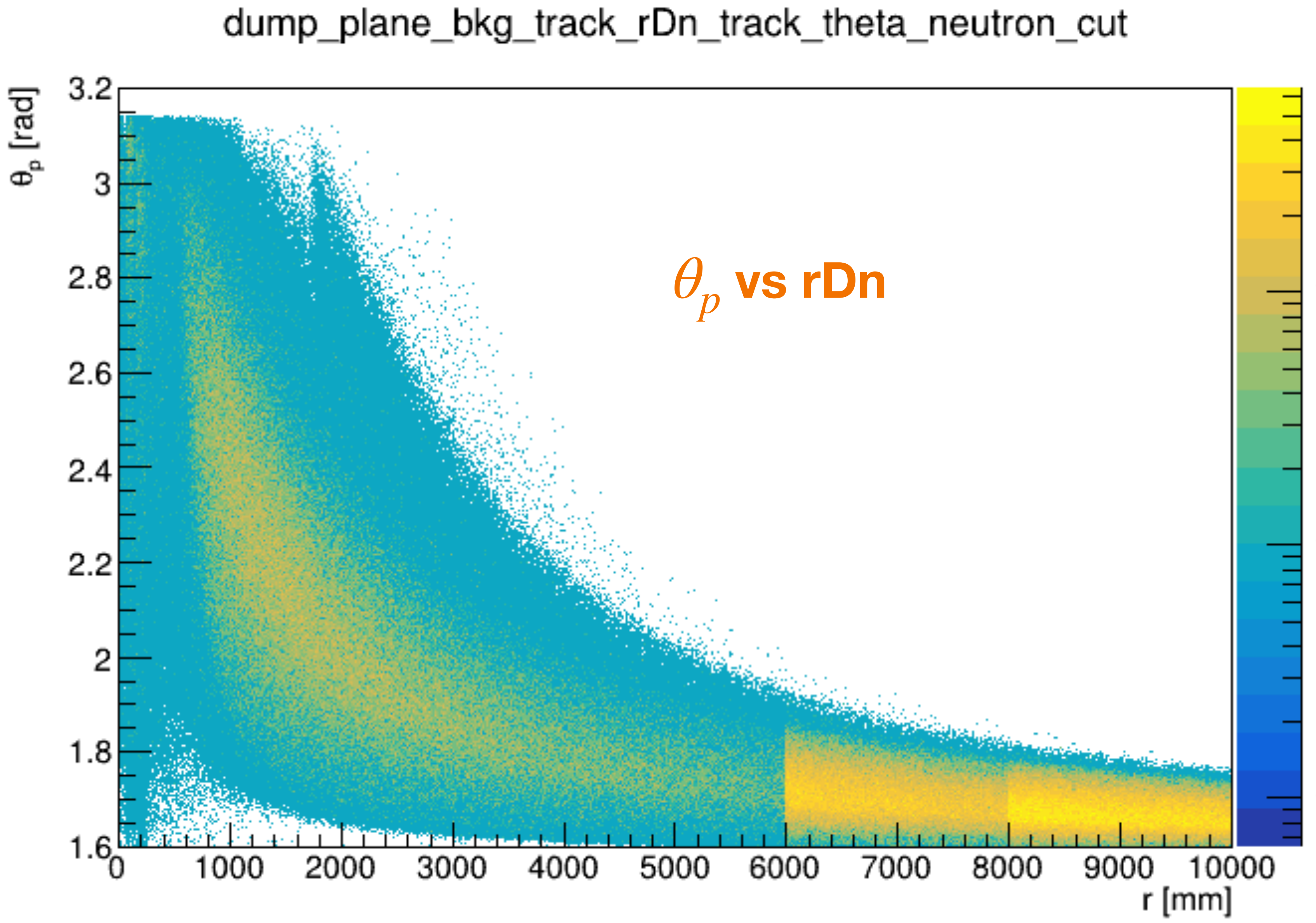
# Baseline distributions from FullSim in LUXE: neutron at sampling surface



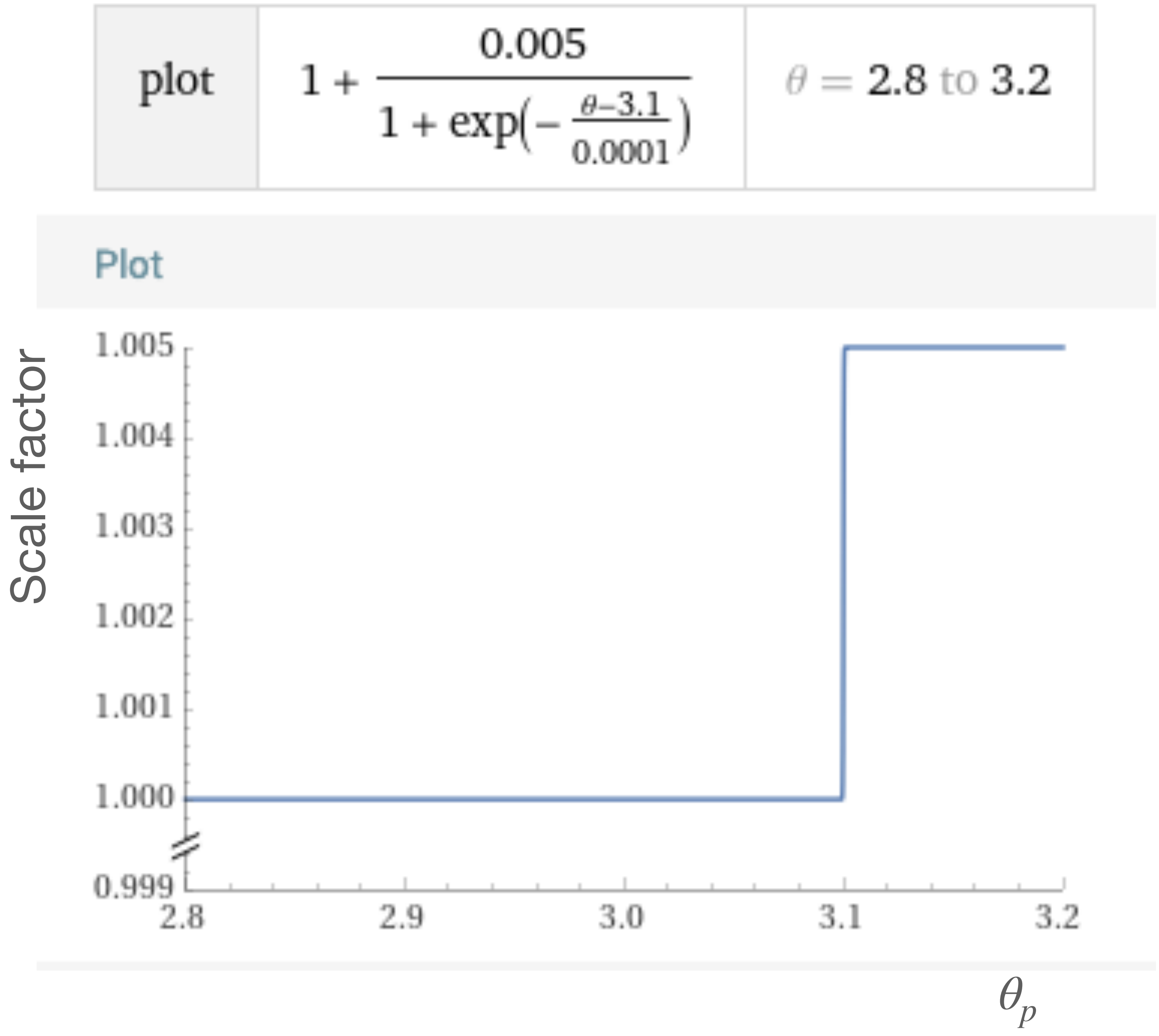




★For E vs t plot of neutron, we only go up to 100 eV of neutron.

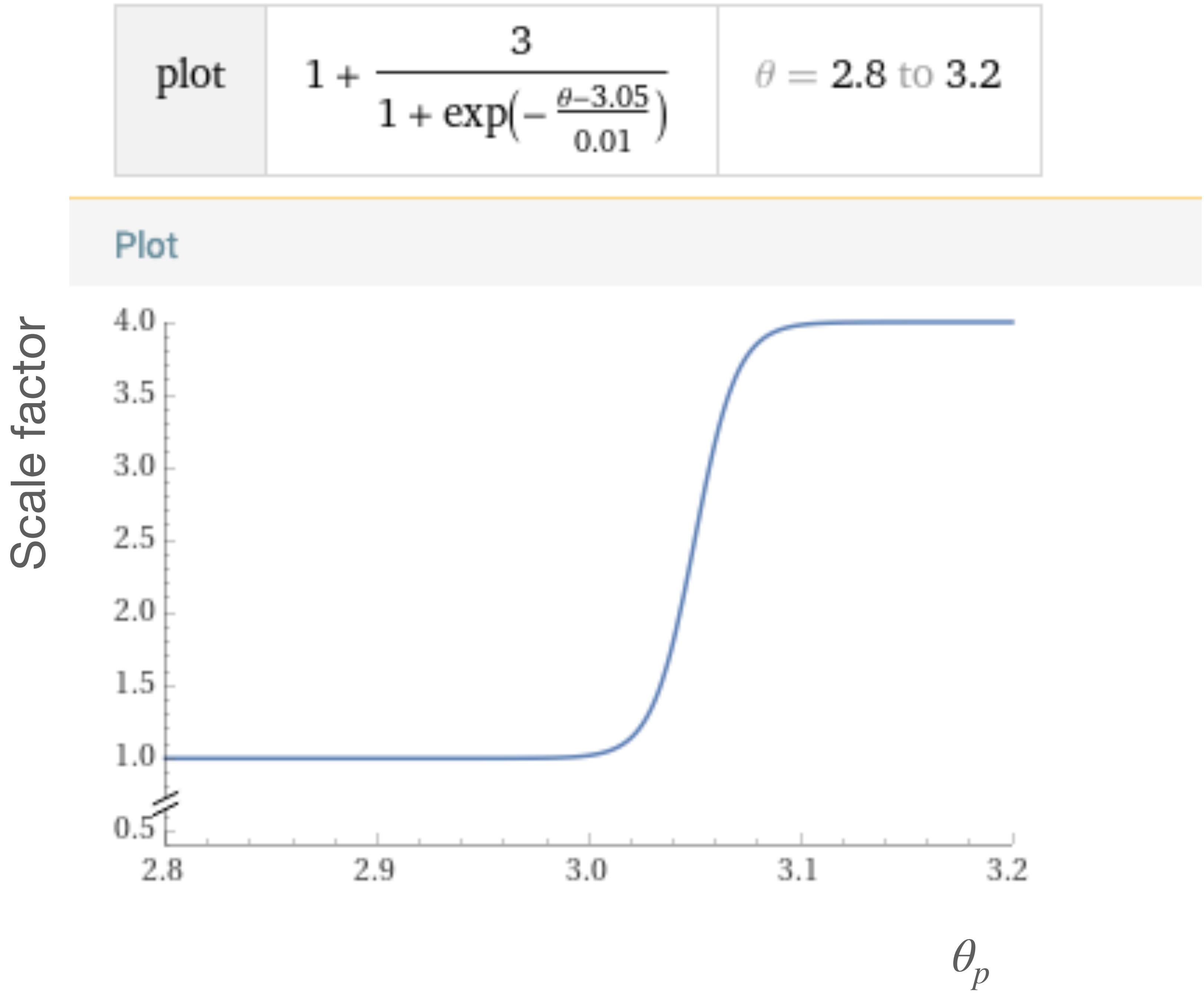
**★Neutron less energetic than that are not interesting.**

# Scale factor:

- Neutron and photon need different scale factor.
- Played with many different values and checked the FastSim distributions.
- Settled with scale factor depending only on $\theta_p$:




Neutron scale factor




Photon scale factor