中央研究院物理研究所
INSTITUTE OF PHYSICS, ACADEMIA SINICA

Status Update

2025/12/04

ZDC ML

WAI YUEN CHAN

Roadmap

Alan

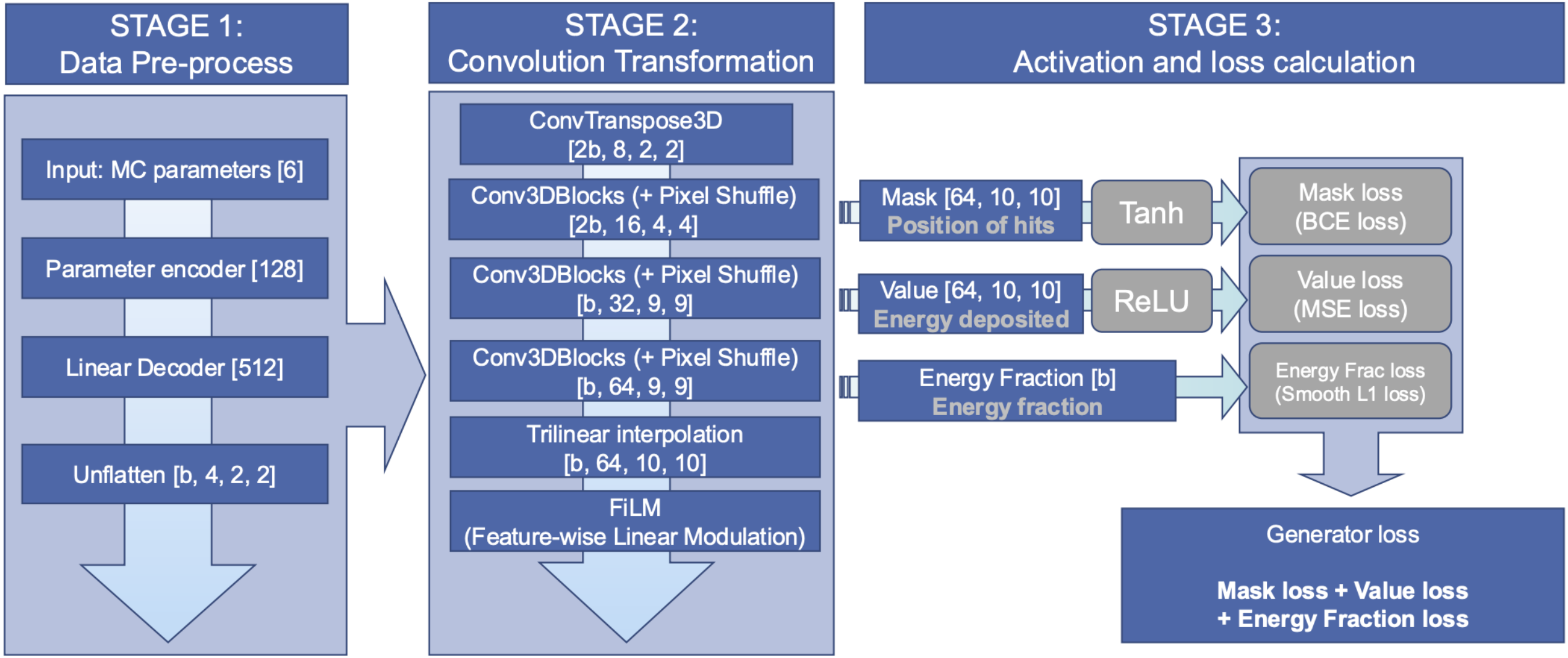
Phase 1A (NOW): v6	Phase 2A (~11/30): v7	Phase 3 (~12/31): v8
HCAL has been added (as a parallel path)	Losses coupling	PaNN development
	Z-X plane learning	

Yen-Chen

Phase 1B (~11/28): v3 -> v6	Phase 2B (~12/12): v6 & v7	Phase 3 (~12/31): v8
Understand framework structure	ECAL-HCAL coupling: Energy conservation layer	PaNN development

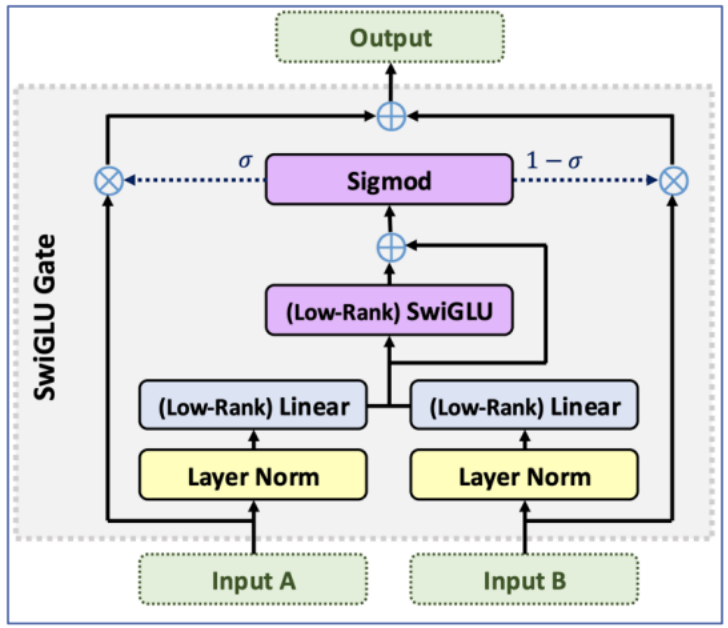
ZDC-ML framework: HCAL Generator workflow

Here is the basic workflow for HCAL, which is implemented at v5:



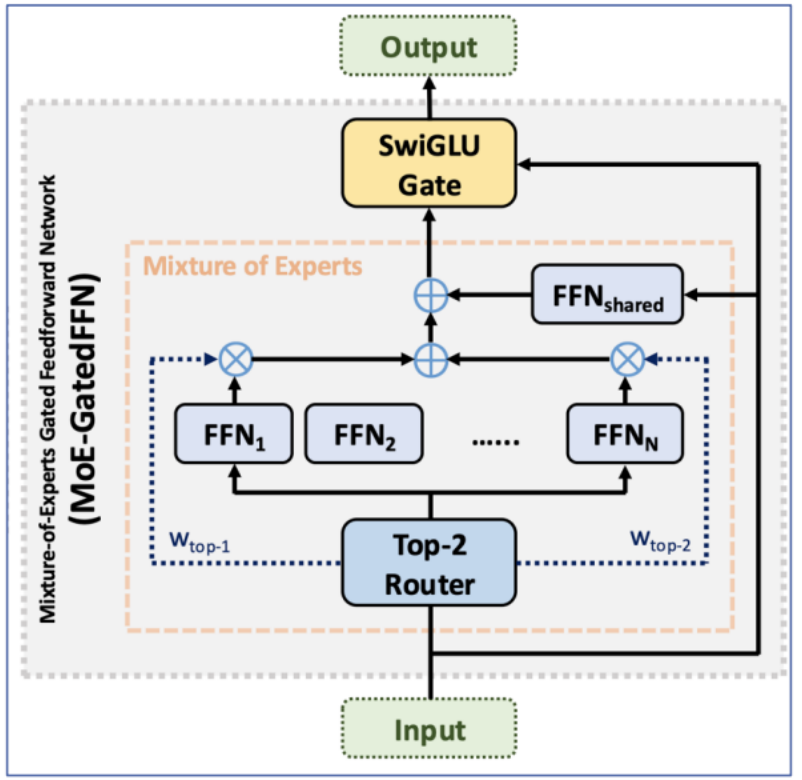
Advanced techniques

Here are 2 networks that was introduced by Hsin-Yu at the last meeting.



SwiGLU Gate:

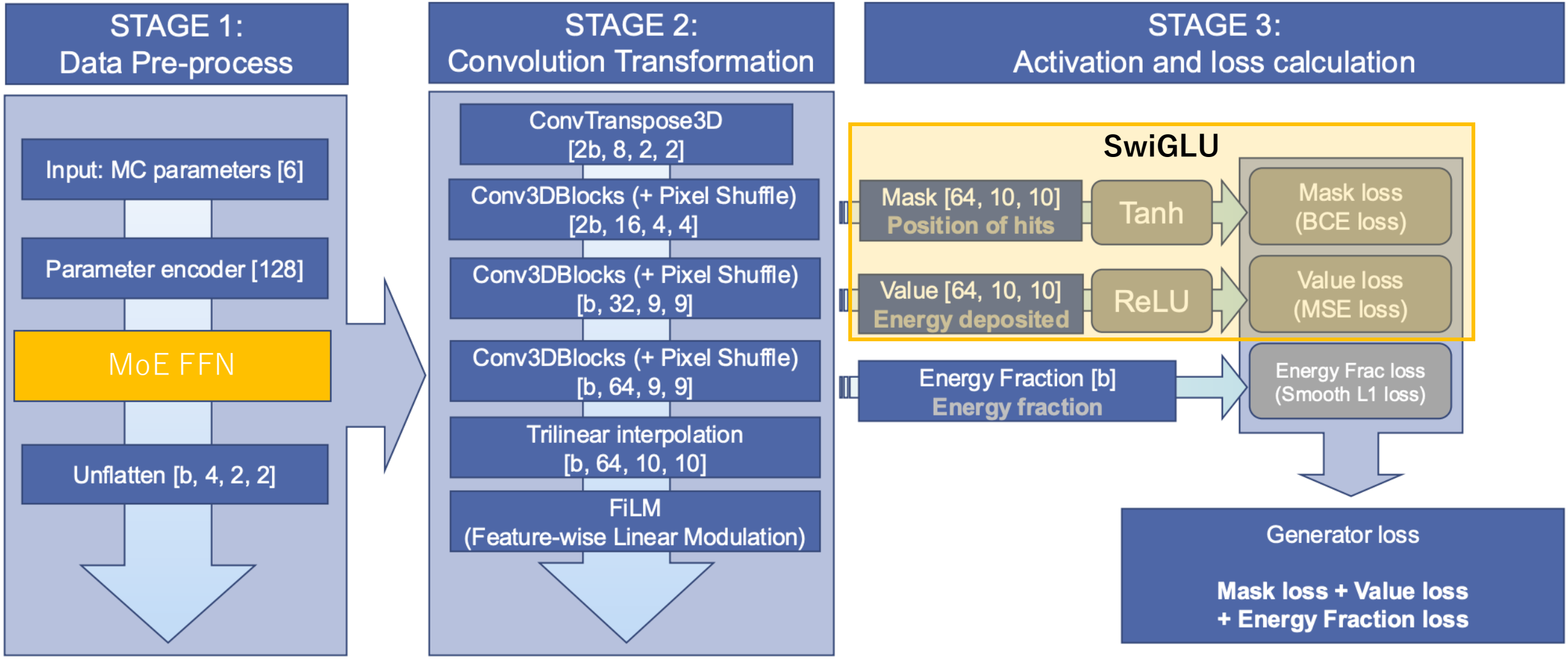
- Transform input A and B and learn the σ.
- Better balance in the coupling of A and B.



MoE-FFN:

- Use multiple experts (mini-networks) to learn the features of the input, then pick up the top-2 ranked as the input of SwiGLU.

HCAL Generator in v7



SwiGLU application explain

SwiGLU:
$\sigma \odot A + (1 - \sigma) \odot B$

What it means

- If `σ = 1.0`: output = `conv_a` (mask-only)
- If `σ = 0.0`: output = `conv_b` (value-only)
- If `σ = 0.5`: output = `0.5 * conv_a + 0.5 * conv_b` (balanced)

Example calculation

Assume at one spatial location:

- `conv_a = [0.8, 0.6, 0.4, ...]` (8 channels, low_rank=8)
- `conv_b = [0.2, 0.3, 0.1, ...]` (8 channels)
- `σ = [0.7, 0.3, 0.9, ...]` (8 channels, per-channel gating)

Per channel:

- Channel 0: `0.7 * 0.8 + 0.3 * 0.2 = 0.56 + 0.06 = 0.62` (mask-dominated)
- Channel 1: `0.3 * 0.6 + 0.7 * 0.3 = 0.18 + 0.21 = 0.39` (value-dominated)
- Channel 2: `0.9 * 0.4 + 0.1 * 0.1 = 0.36 + 0.01 = 0.37` (mask-dominated)

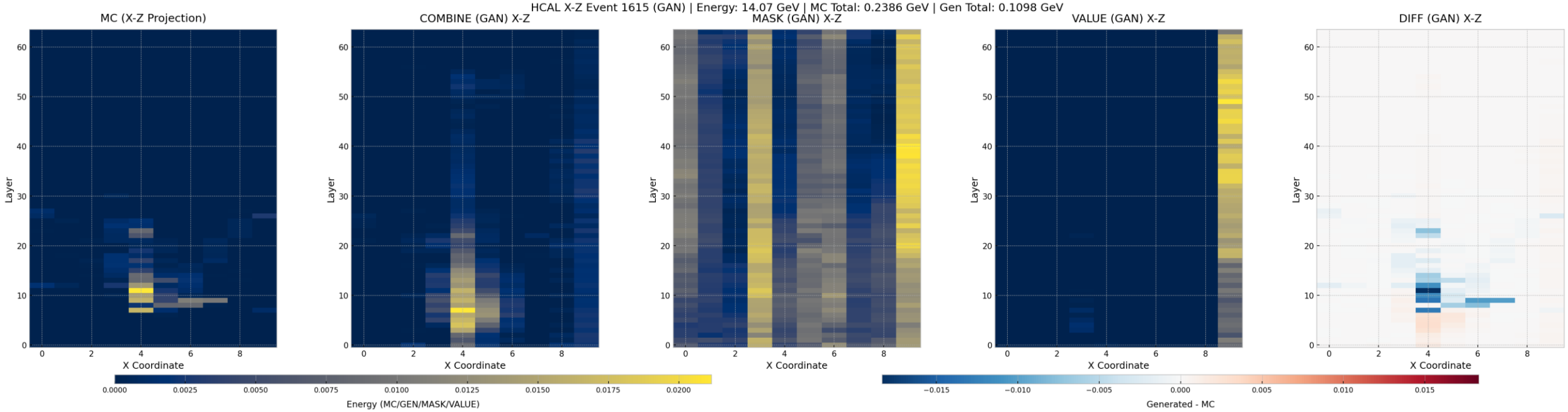
Final output: `[0.62, 0.39, 0.37, ...]` (8 channels) → projected to 1 channel

- Use Mask and Value as the input
- Use 8 channels low rank linear transformation to calculate σ
- Eventually get a better prediction of the relationship between mask and value
- Being use in both ECAL and HCAL
- ECAL didn't gain much but HCAL shows much better results.

- Mask and Value loss can still being used in the training, but a conflict would occur if the weighting is too much.
- On the other hand, without any mask and value loss would causing the SwiGLU output non-physical (will see later on)

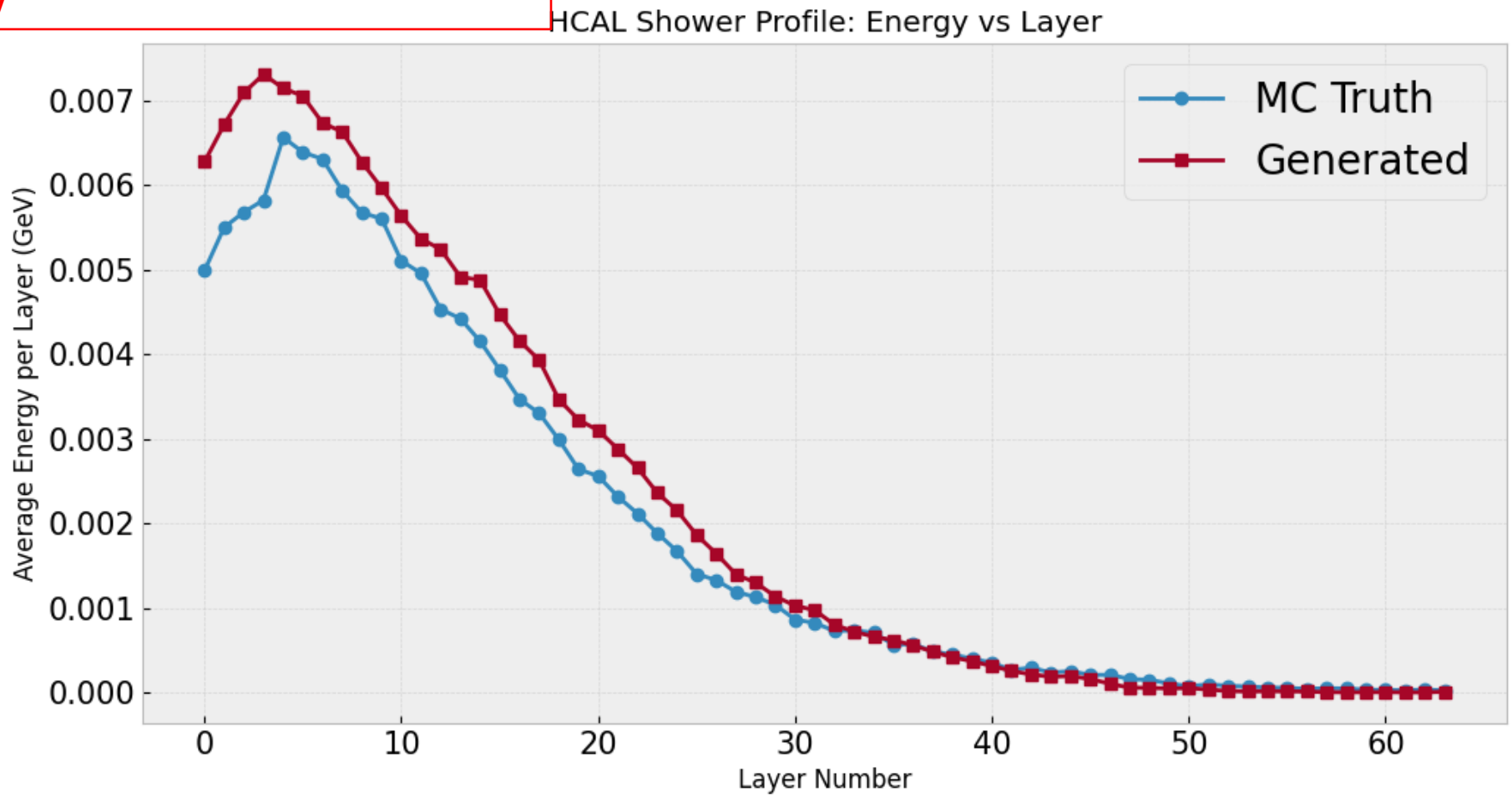
SwiGLU result 1: Pencil beam

8000 events, 50 – 250 GeV, pencil beam
SwiGLU only



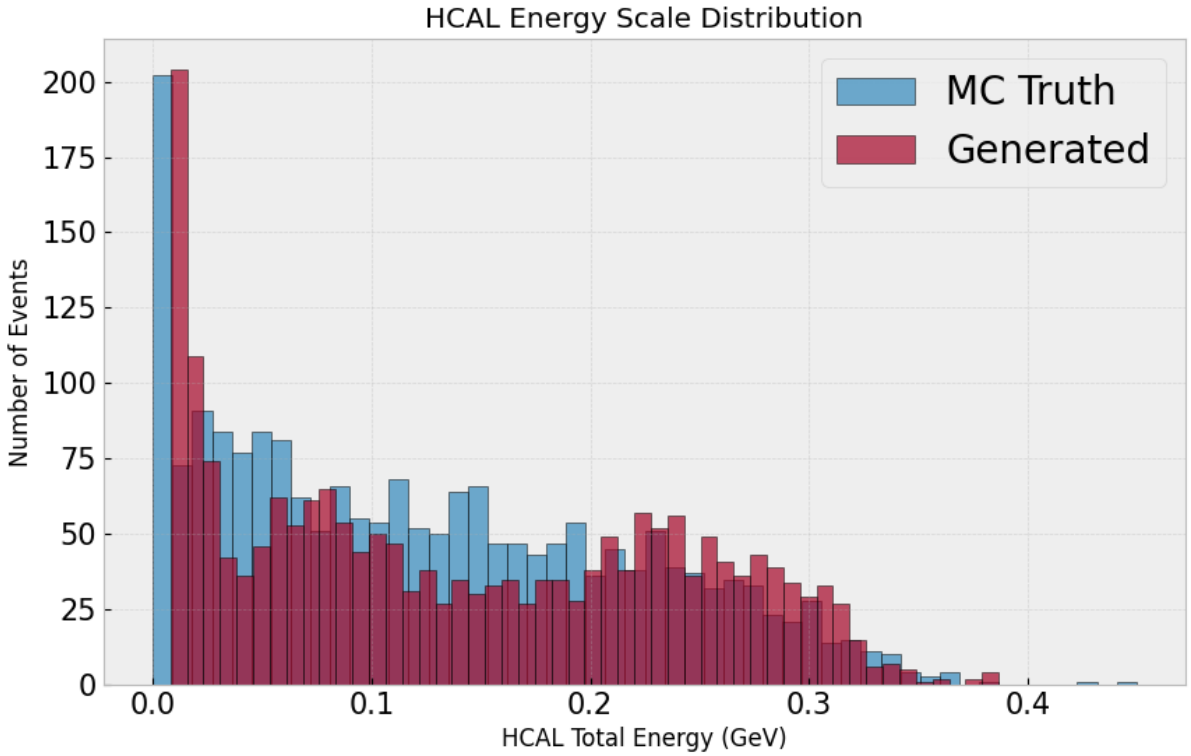
SwiGLU result 1: Pencil beam

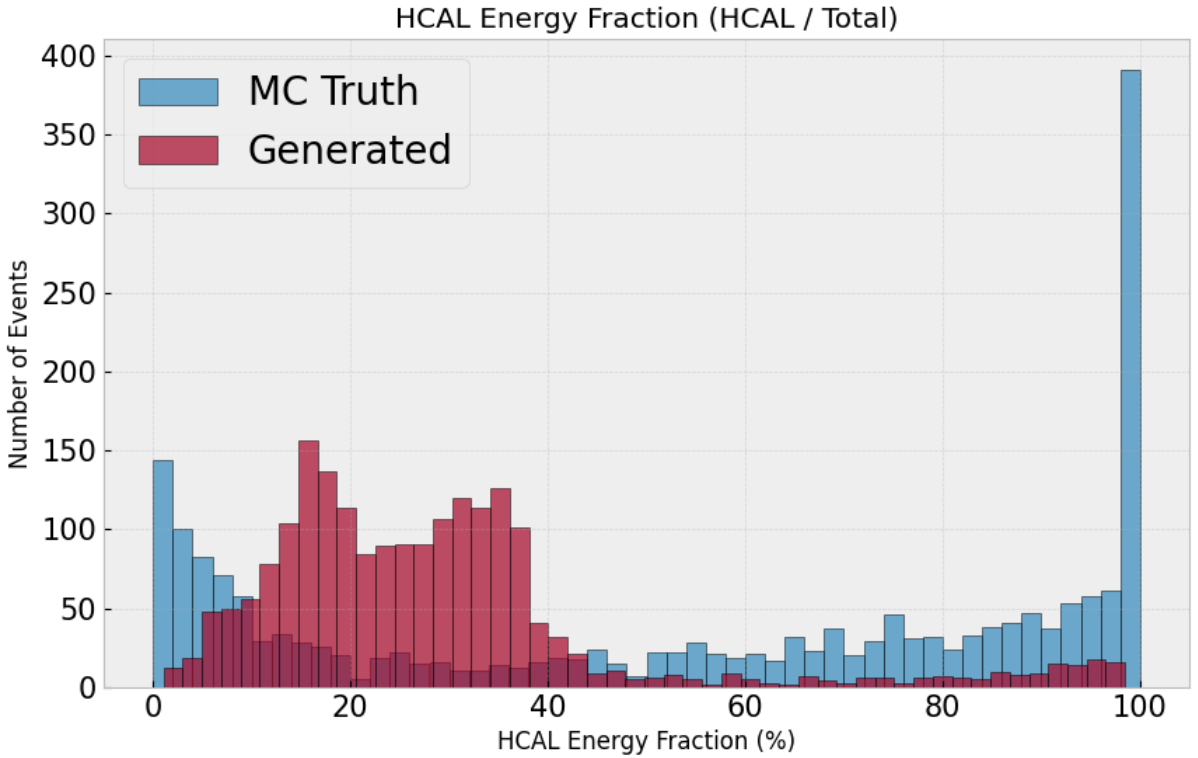
8000 events, 50 – 250 GeV, pencil beam
SwiGLU only



SwiGLU result 1: Pencil beam

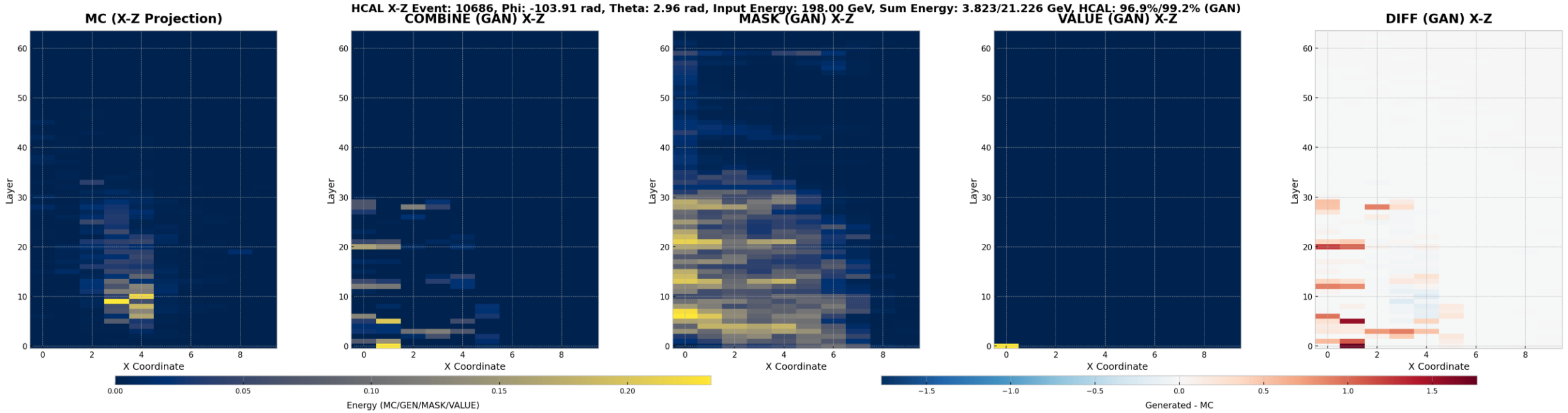
8000 events, 50 – 250 GeV, pencil beam
SwiGLU only





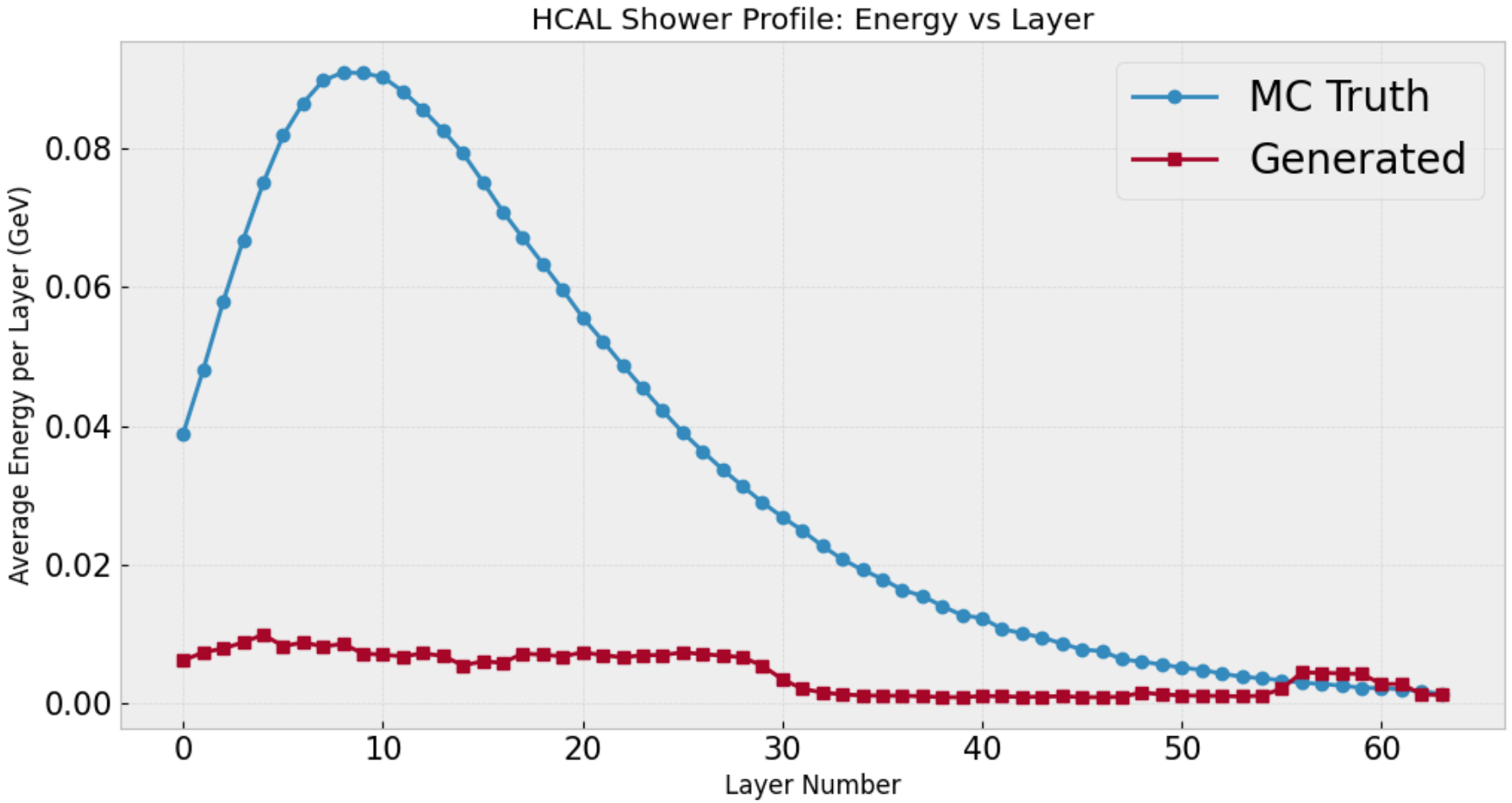
SwiGLU result 2: Full coverage

30000 events, 0 – 300 GeV
SwiGLU only



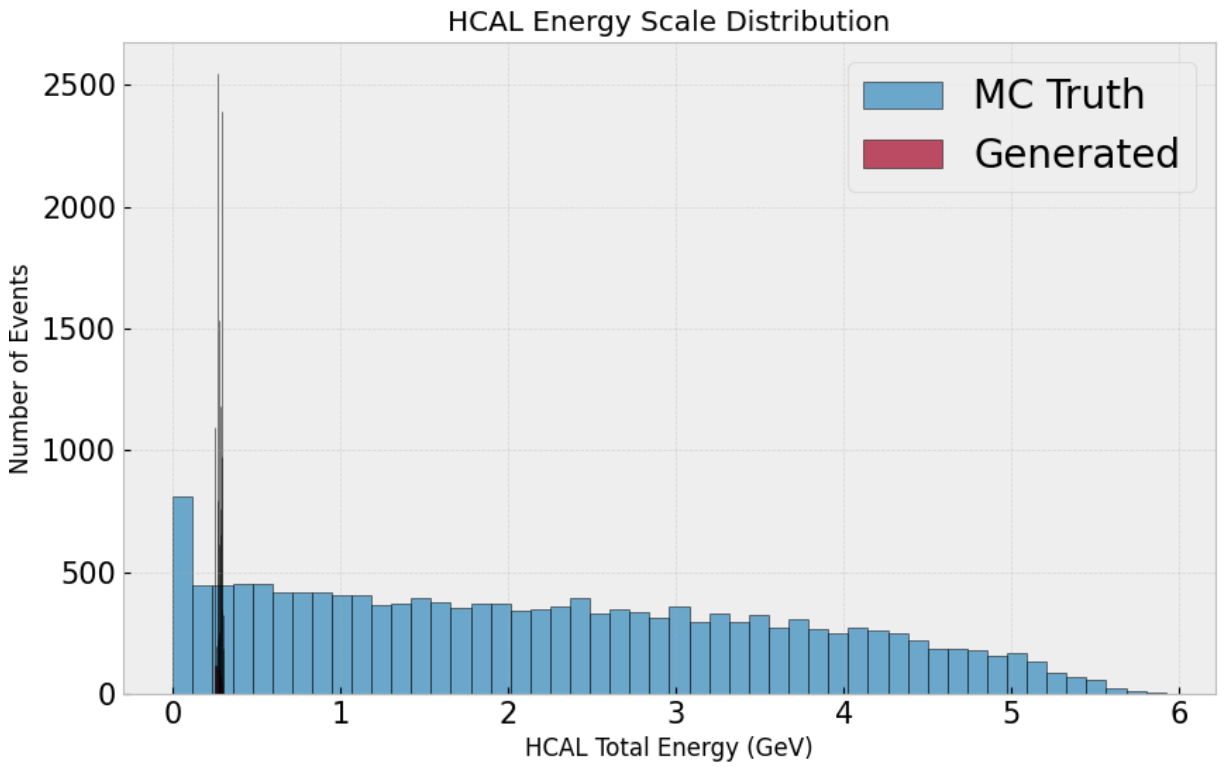
SwiGLU result 2: Full coverage

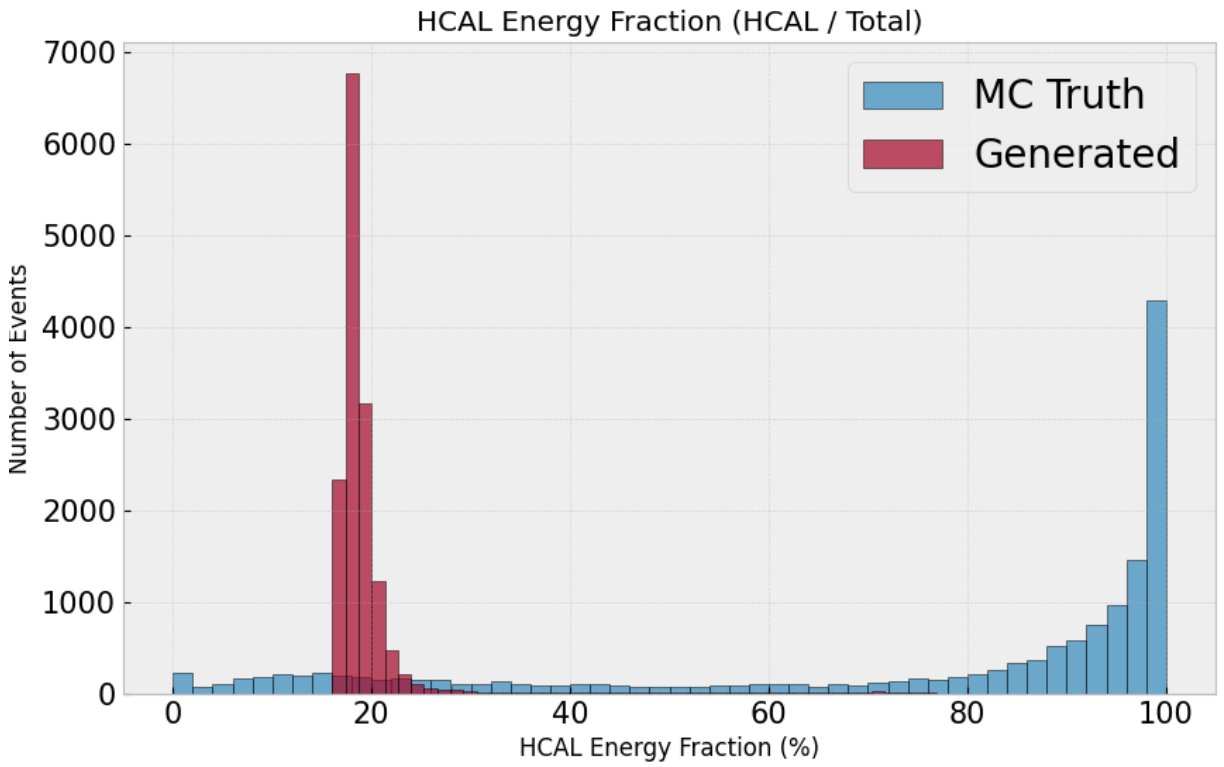
30000 events, 0 – 300 GeV
SwiGLU only



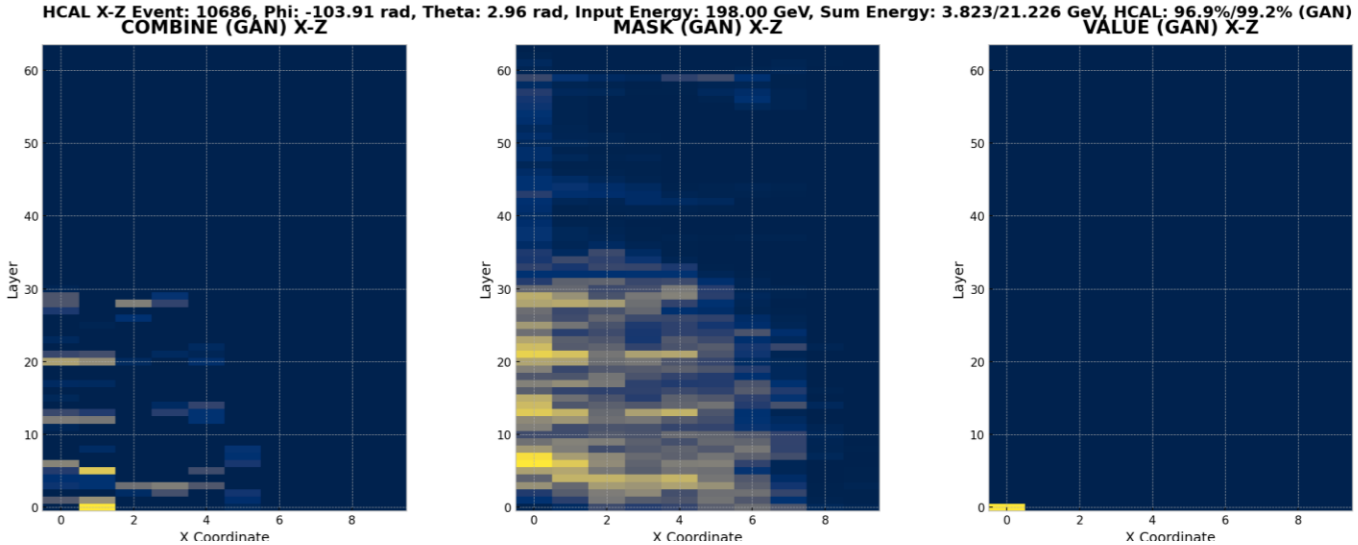
SwiGLU result 2: Full coverage

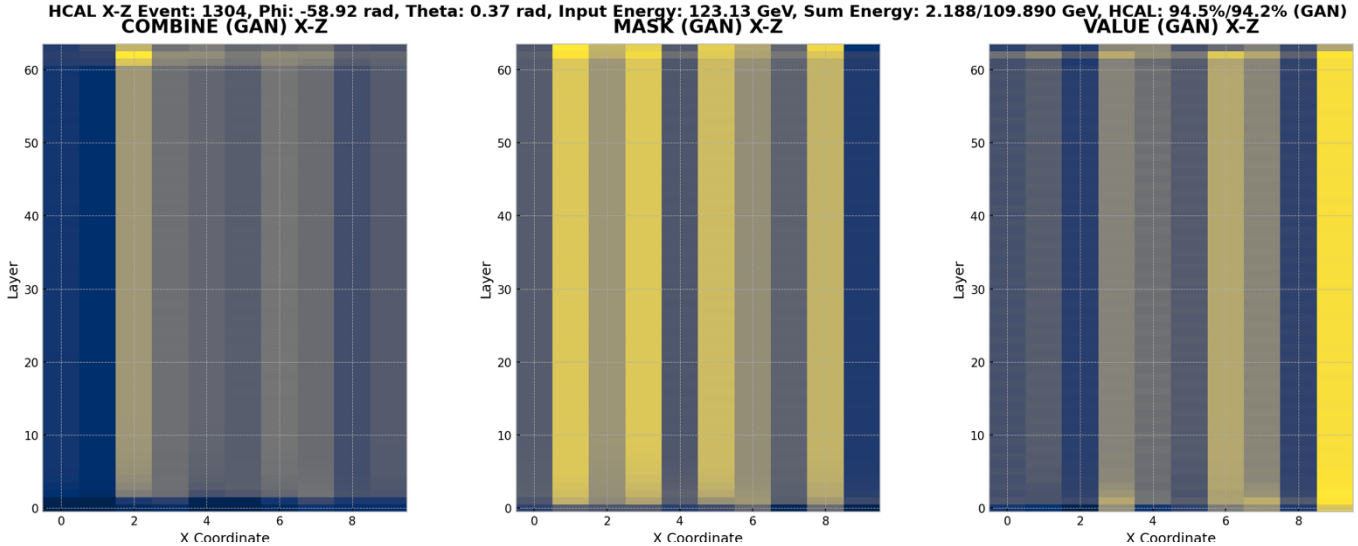
30000 events, 0 – 300 GeV
SwiGLU only





Effect from weighting





Here we show the effect from the mask and value loss tuning.
Weight = 1 : Model must learn from it
Weight = 0: Not necessary to learn from it

Top:

- Mask loss weight = 0.5
- Value loss weight = 1.0
- **Meaning:** Learn from mask and must learn the value. Since we use SwiGLU, forcing the model to learn Value will constraint the range of σ

Bottom:

- Mask loss weight = 1.0
- Value loss weight = 0.0
- **Meaning:** Force the model to learn the mask in all cases and learn the value freely. Eventually the model pick up almost all the cells since in general every cell would have some value (even extremely small). Indicated the σ highly depends on Mask while most of the cell should have **NO hits** per event.

Outlook

Summary

- In v7, we try to add SwiGLU and MoE FFN into the framework.
- In the pencil beam run, SwiGLU looks promising.
- However, in the full coverage run, more tuning is needed.
- So far the MoE FFN only being tested in ECAL, not much improvement. Need to check with HCAL as well.

Cross/self attention in the X-Z relation

- It takes almost ~26 times longer for 1 epoch due to the complexity of coupling all 3600 cells in HCAL.
- A better approach would be transformed the 3D matrix into 1D array or separate it into 2D + 1D.
- At the moment we disable this part in the training, as PaNN might be a better options.

Suggestion in PaNN

- We can add some hard constraint (requirement) and force the model to learn from the physics, rather than the spatial relation in the 3D matrix. For example:
  - The X-Z relation
  - Spreading in each layers
  - Energy fraction
  - Relation between ECAL and HCAL