# Fast Simulation of Calorimetry showers (FastSim)

Mikołaj Piórczyński

FastSim Summer Students Meeting, 12.07.2024

# Who am I?

- Live in Warsaw, Poland.
- Defended bachelor's thesis with honors at Warsaw University of Technology. Thesis title: 'Efficient Inference in Transformer Models with Dense to Dynamic-k Mixture-of-Experts Model Conversion'.
- Worked 1.5 years as an intern in the Machine Translation Team at Samsung R&D Institute Poland and 0.5 years as an MLE intern at AI Clearing (AI-powered construction progress tracking based on drone-captured data).
- Co-organizing ML in PL Conference, one of the biggest ML-oriented conferences in Poland.
- Visited CERN during a high-school trip a few years ago.
- In free time: playing the violin and running

SAMSUNG

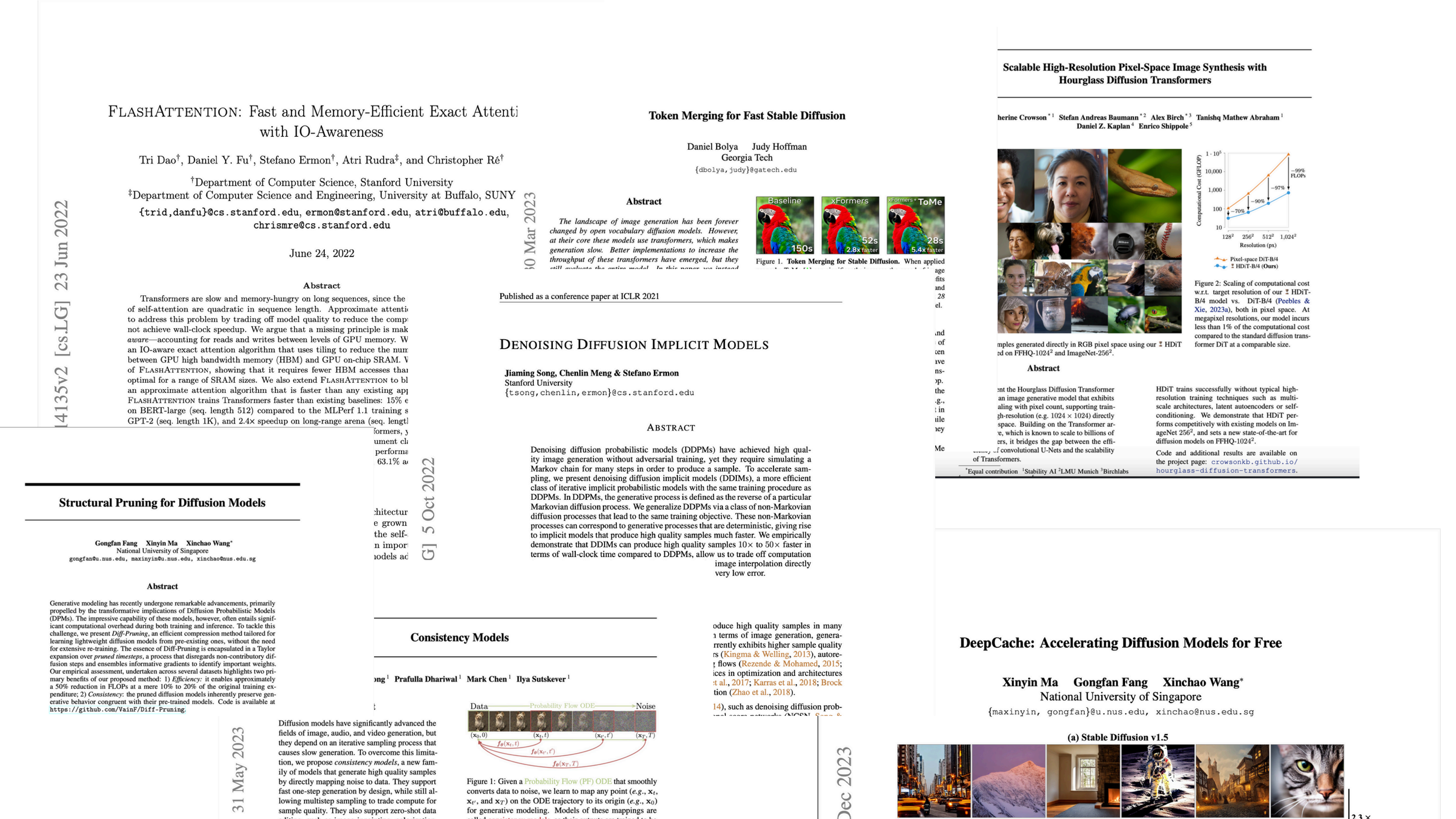
AI Clearing

ML in PL ASSOCIATION

Faculty of Mathematics and Information Science
WARSAW UNIVERSITY OF TECHNOLOGY

# How to make diffusion models faster?*

***while maintaining a high quality of samples**


FLASHATTENTION: Fast and Memory-Efficient Exact Attenti
with IO-Awareness
Tri Dao†, Daniel Y. Fu†, Stefano Ermon†, Atri Rudra‡, and Christopher Ré†
†Department of Computer Science, Stanford University
‡Department of Computer Science and Engineering, University at Buffalo, SUNY
{trid,danfu}@cs.stanford.edu, ermon@stanford.edu, atri@buffalo.edu,
chrismre@cs.stanford.edu
June 24, 2022
Abstract
Token Merging for Fast Stable Diffusion
Daniel Bolya Judy Hoffman
Georgia Tech
{dbolya,judy}@gatech.edu
Abstract
Baseline
150s
xFormers
52s
2.8x faster
xFormers + ToMe
28s
5.4x faster
Figure 1. Token Merging for Stable Diffusion.
Published as a conference paper at ICLR 2021
DENOISING DIFFUSION IMPLICIT MODELS
Jiaming Song, Chenlin Meng & Stefano Ermon
Stanford University
{tsong,chenlin,ermon}@cs.stanford.edu
ABSTRACT
Denoising diffusion probabilistic models (DDPMs) have achieved high quality image generation without adversarial training, yet they require simulating a Markov chain for many steps in order to produce a sample. To accelerate sampling, we present denoising diffusion implicit models (DDIMs), a more efficient class of iterative implicit probabilistic models with the same training procedure as DDPMs. In DDPMs, the generative process is defined as the reverse of a particular Markovian diffusion process. We generalize DDPMs via a class of non-Markovian diffusion processes that lead to the same training objective. These non-Markovian processes can correspond to generative processes that are deterministic, giving rise to implicit models that produce high quality samples much faster. We empirically demonstrate that DDIMs can produce high quality samples 10× to 50× faster in terms of wall-clock time compared to DDPMs, allow us to trade off computation
Scalable High-Resolution Pixel-Space Image Synthesis with
Hourglass Diffusion Transformers
Stefan Andreas Baumann Alex Birch Tanishq Mathew Abraham
Daniel Z. Kaplan Enrico Shippole
Figure 2: Scaling of computational cost w.r.t. target resolution of our HDiT-B/4 model vs. DiT-B/4 (Peebles & Xie, 2023a), both in pixel space. At megapixel resolutions, our model incurs less than 1% of the computational cost compared to the standard diffusion transformer DiT at a comparable size.
Abstract
HDiT trains successfully without typical high-resolution training techniques such as multiscale architectures, latent autoencoders or self-conditioning. We demonstrate that HDiT performs competitively with existing models on ImageNet 256², and sets a new state-of-the-art for diffusion models on FFHQ-1024².
Code and additional results are available on the project page: crowsonkb.github.io/hourglass-diffusion-transformers.
Structural Pruning for Diffusion Models
Gongfan Fang Xinyin Ma Xinchao Wang*
National University of Singapore
gongfan@u.nus.edu, maxinyin@u.nus.edu, xinchao@nus.edu.sg
Abstract
Generative modeling has recently undergone remarkable advancements, primarily propelled by the transformative implications of Diffusion Probabilistic Models (DPMs). The impressive capability of these models, however, often entails significant computational overhead during both training and inference. To tackle this challenge, we present Diff-Pruning, an efficient compression method tailored for learning lightweight diffusion models from pre-existing ones, without the need for extensive re-training. The essence of Diff-Pruning is encapsulated in a Taylor expansion over pruned timesteps, a process that disregards non-contributory diffusion steps and ensembles informative gradients to identify important weights. Our empirical assessment, undertaken across several datasets highlights two primary benefits of our proposed method: 1) Efficiency: it enables approximately a 50% reduction in FLOPs at a mere 10% to 20% of the original training expenditure; 2) Consistency: the pruned diffusion models inherently preserve generative behavior congruent with their pre-trained models. Code is available at https://github.com/VainF/Diff-Pruning.
Consistency Models
Prafulla Dhariwal Mark Chen Ilya Sutskever
Diffusion models have significantly advanced the fields of image, audio, and video generation, but they depend on an iterative sampling process that causes slow generation. To overcome this limitation, we propose consistency models, a new family of models that generate high quality samples by directly mapping noise to data. They support fast one-step generation by design, while still allowing multistep sampling to trade compute for sample quality. They also support zero-shot data
Data
Probability Flow ODE
Noise
Figure 1: Given a Probability Flow (PF) ODE that smoothly converts data to noise, we learn to map any point (e.g., $x_t$, $x_{t'}$, and $x_T$) on the ODE trajectory to its origin (e.g., $x_0$) for generative modeling. Models of these mappings are
DeepCache: Accelerating Diffusion Models for Free
Xinyin Ma Gongfan Fang Xinchao Wang*
National University of Singapore
{maxinyin, gongfan}@u.nus.edu, xinchao@nus.edu.sg
(a) Stable Diffusion v1.5

**speeding-up diffusion models**

**model architecture**

**diffusion process**

speeding-up diffusion models

model architecture

diffusion process

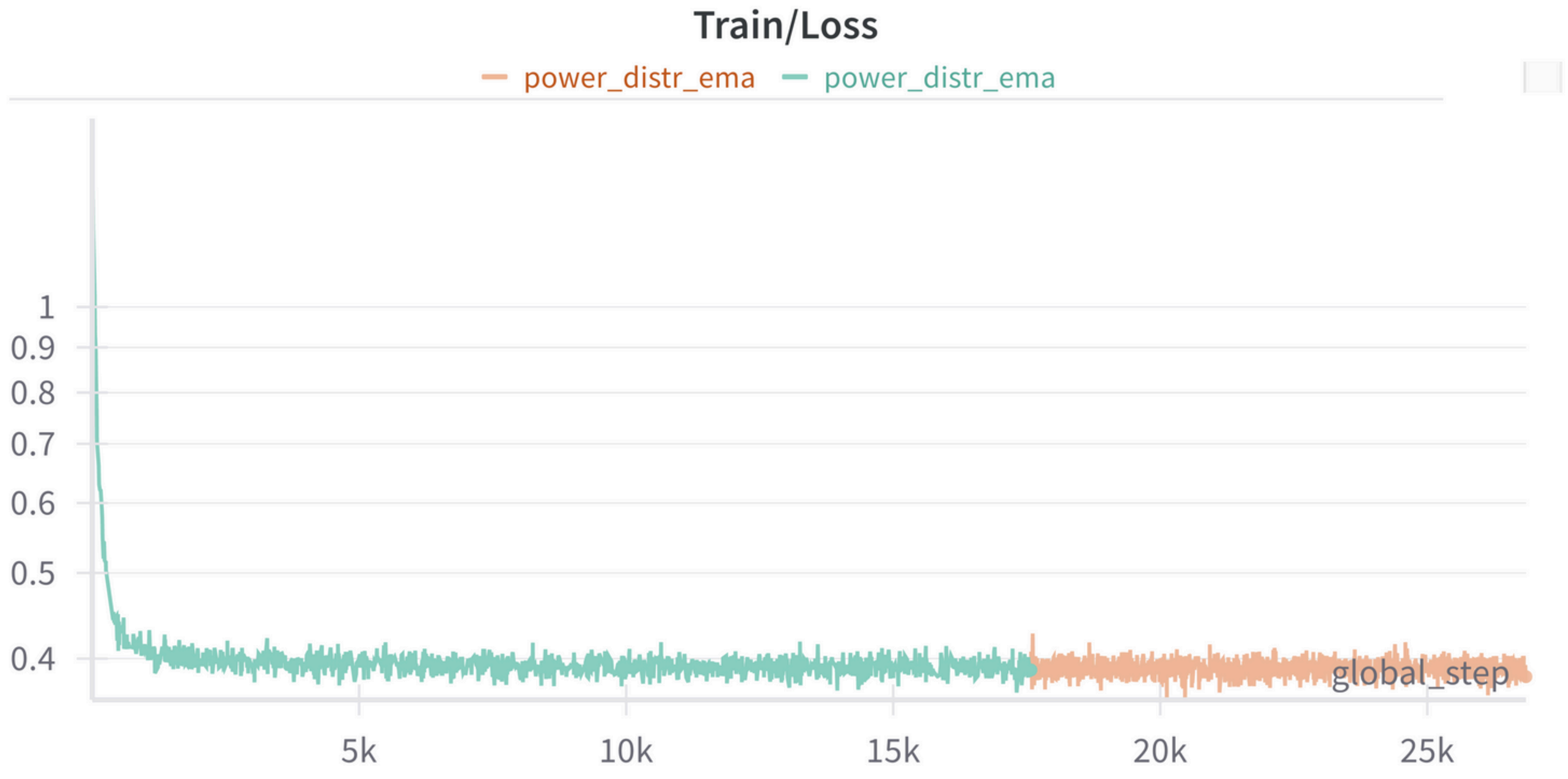
# Project outline

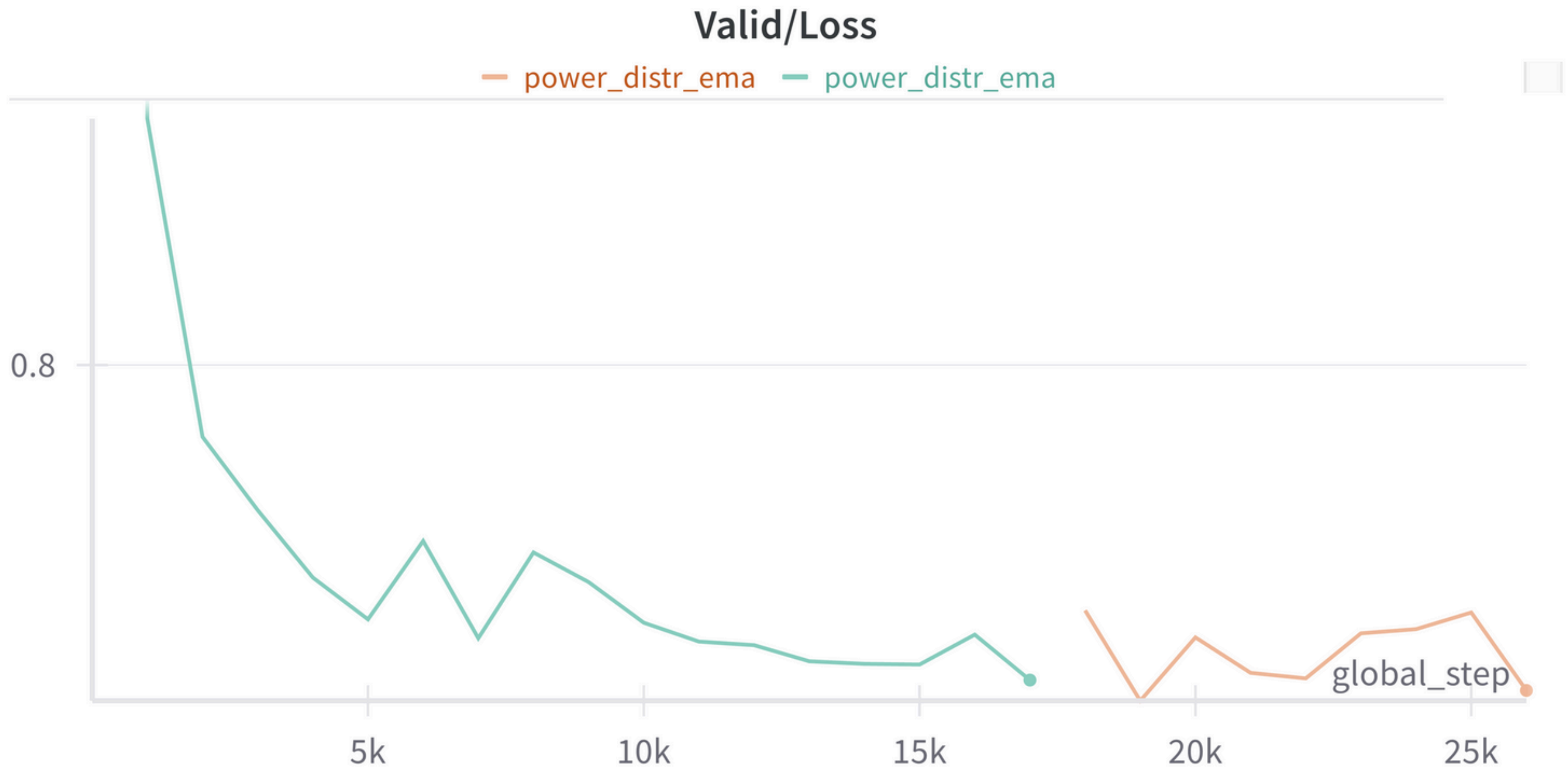
1. DDIM sampling (http://arxiv.org/abs/2010.02502)
2. ~~Progressive Distillation (https://arxiv.org/abs/2202.00512)~~
3. EDM (https://arxiv.org/abs/2206.00364) + ODE solvers (Heun's, DPM-Solver++ (https://arxiv.org/abs/2211.01095))
4. ~~Optional:~~ Consistency Distillation (https://arxiv.org/abs/2303.01469)

Code: https://gitlab.cern.ch/mpiorczy/diffusion4fastsim

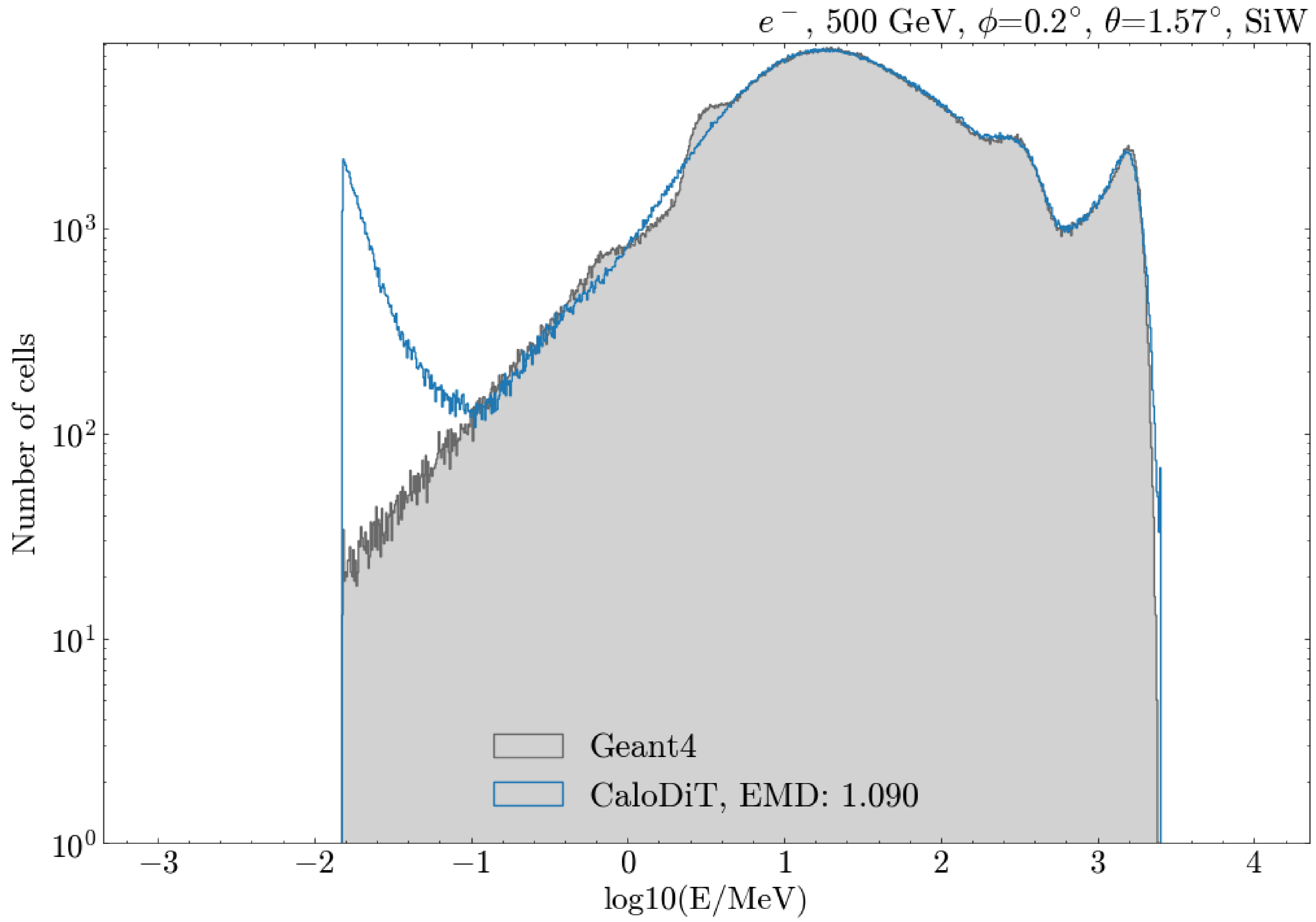
DDIM analysis: https://quiver-cornucopia-fdc.notion.site/DDIM-sampling-b35f60d08b5f4b61b2b075d32bfb97d8

# Updates: EDM w/ power energy distribution

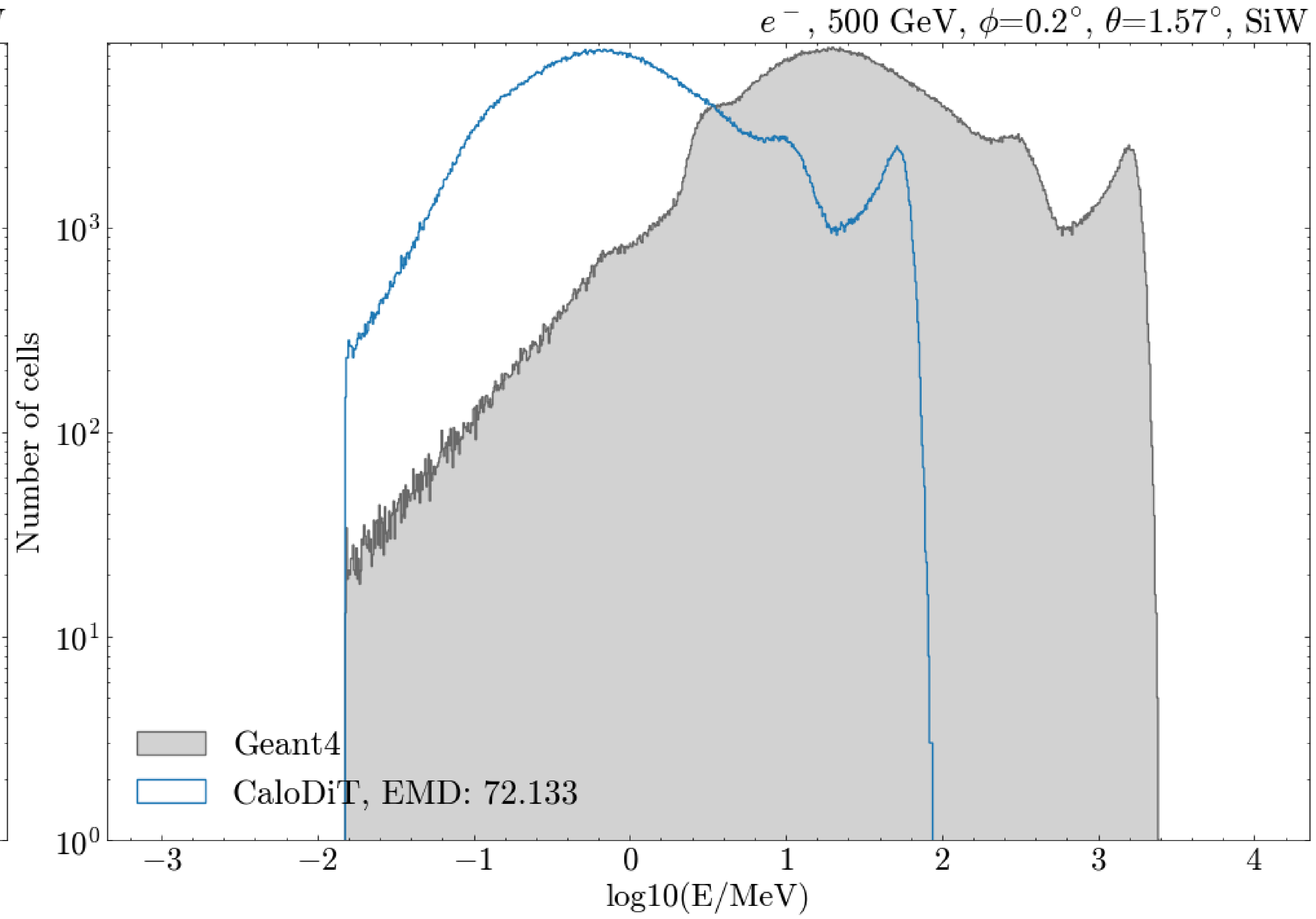


# Updates: EDM w/ uniform energy distribution

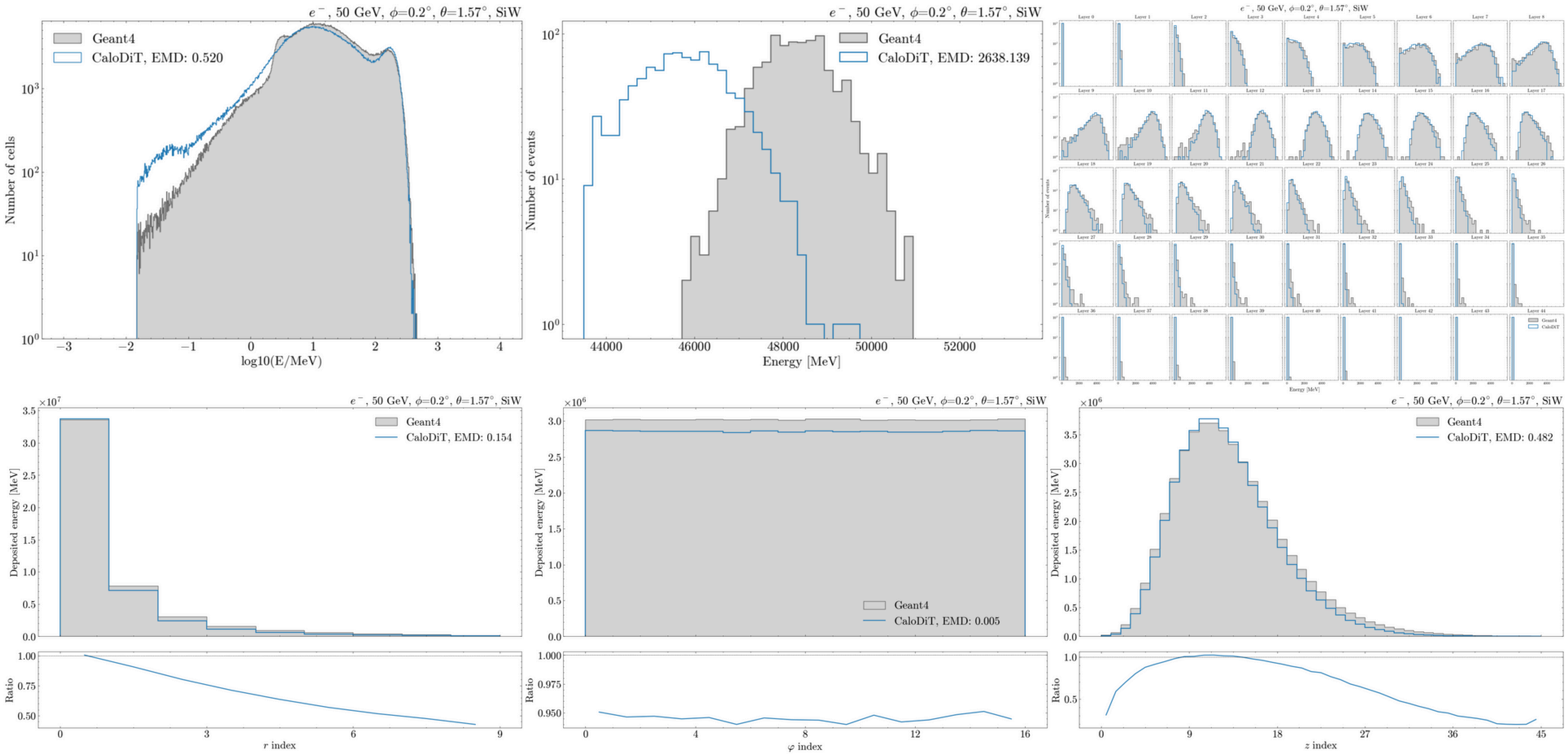



power spectrum

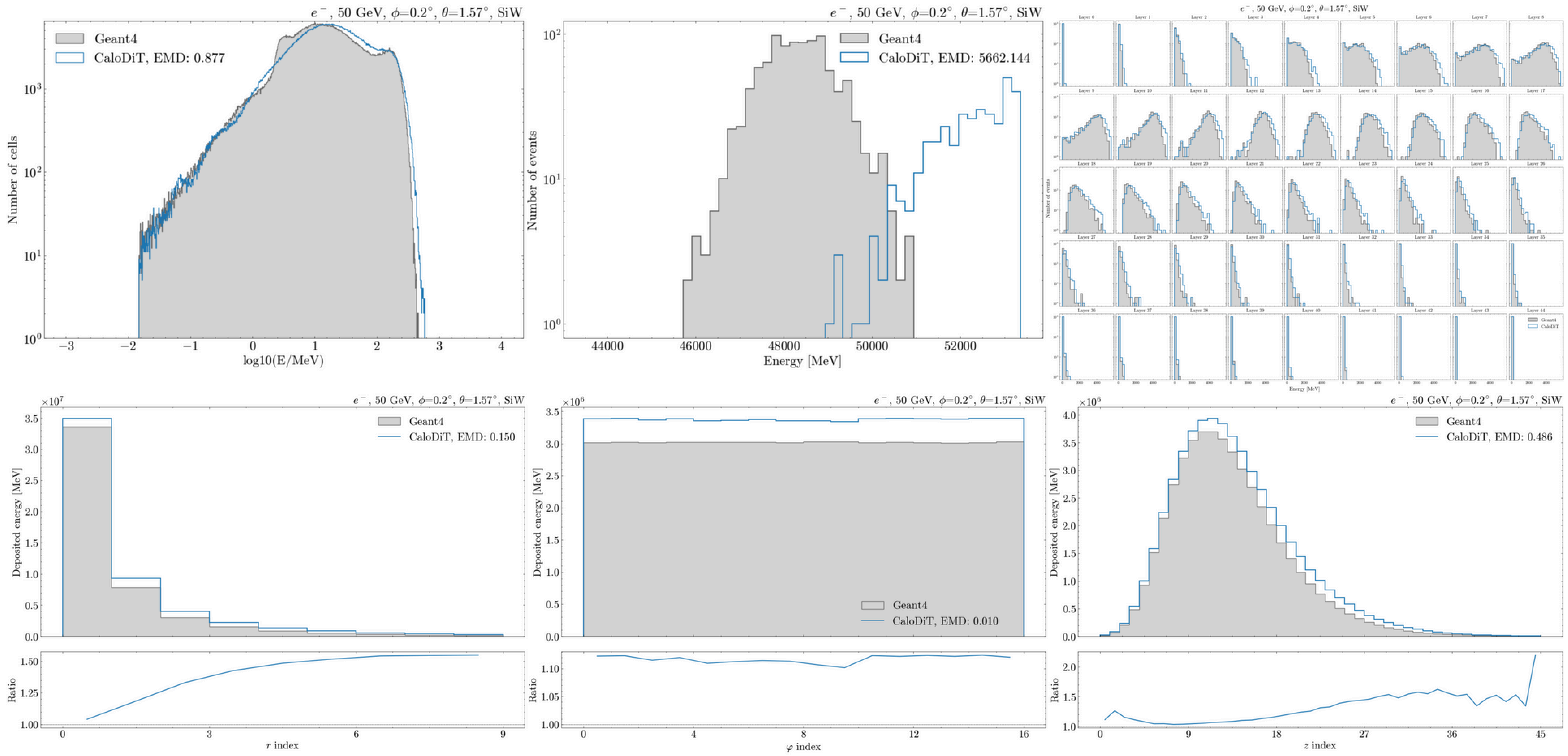



uniform spectrum

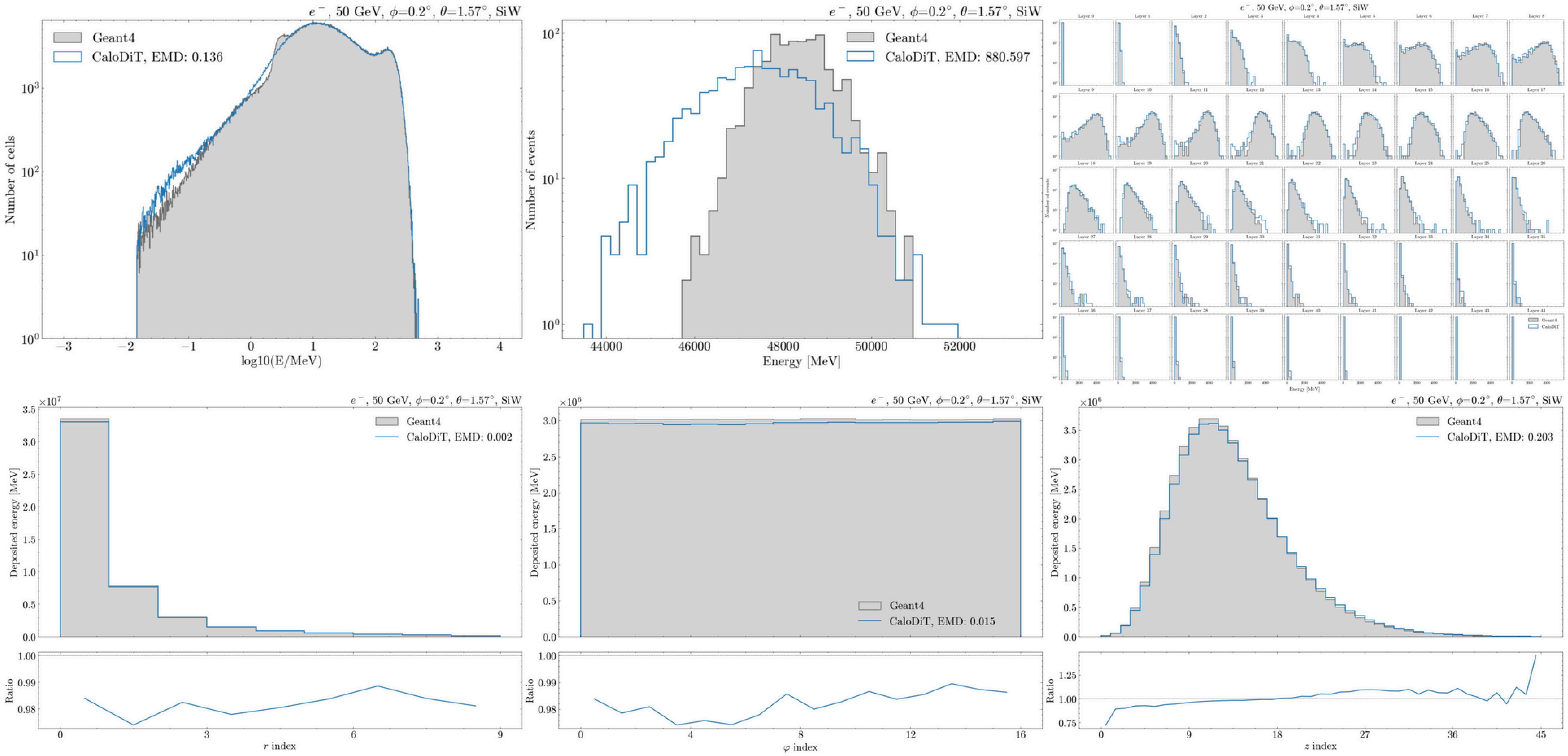
# Updates: Euler, 32 steps



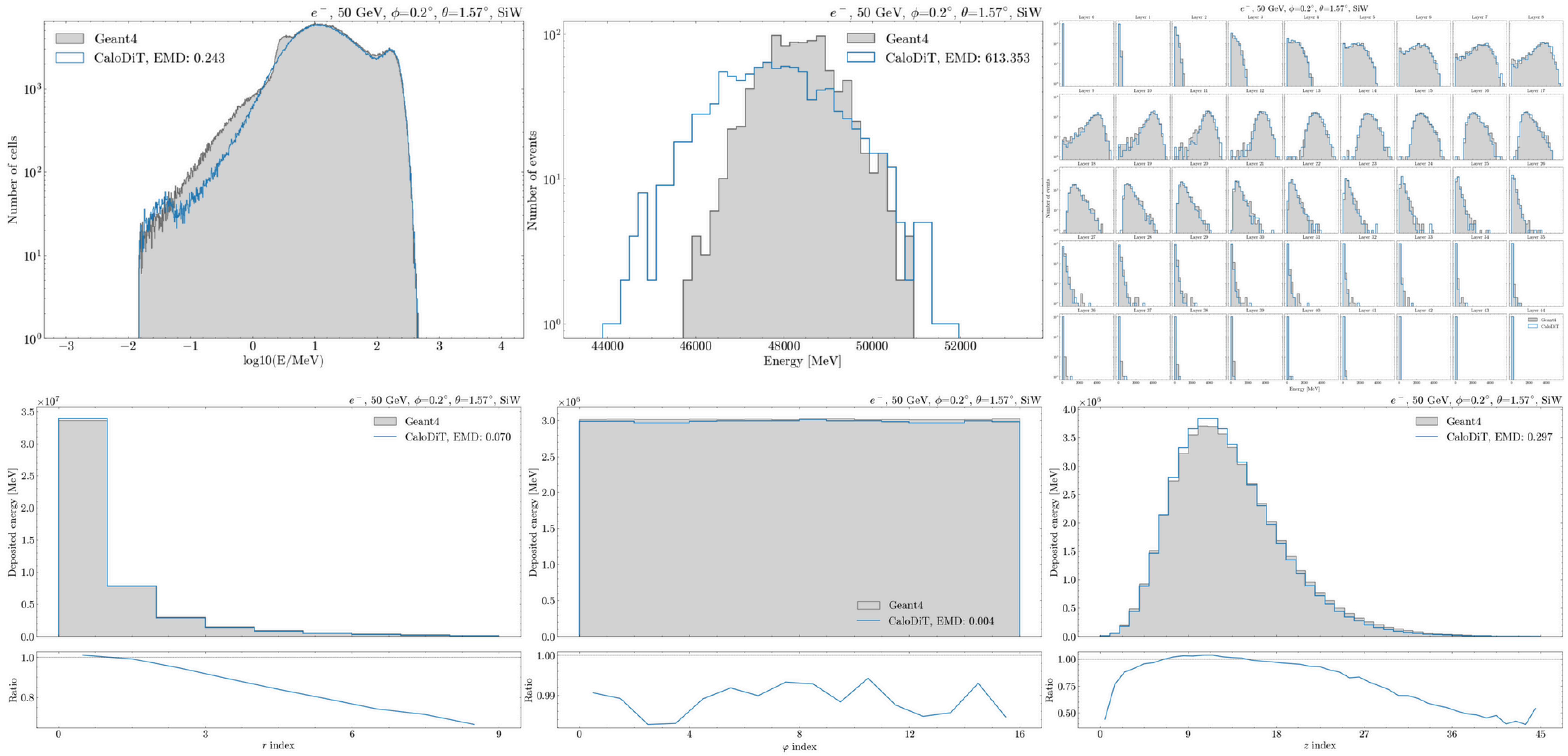
# Updates: Heun, 31 steps



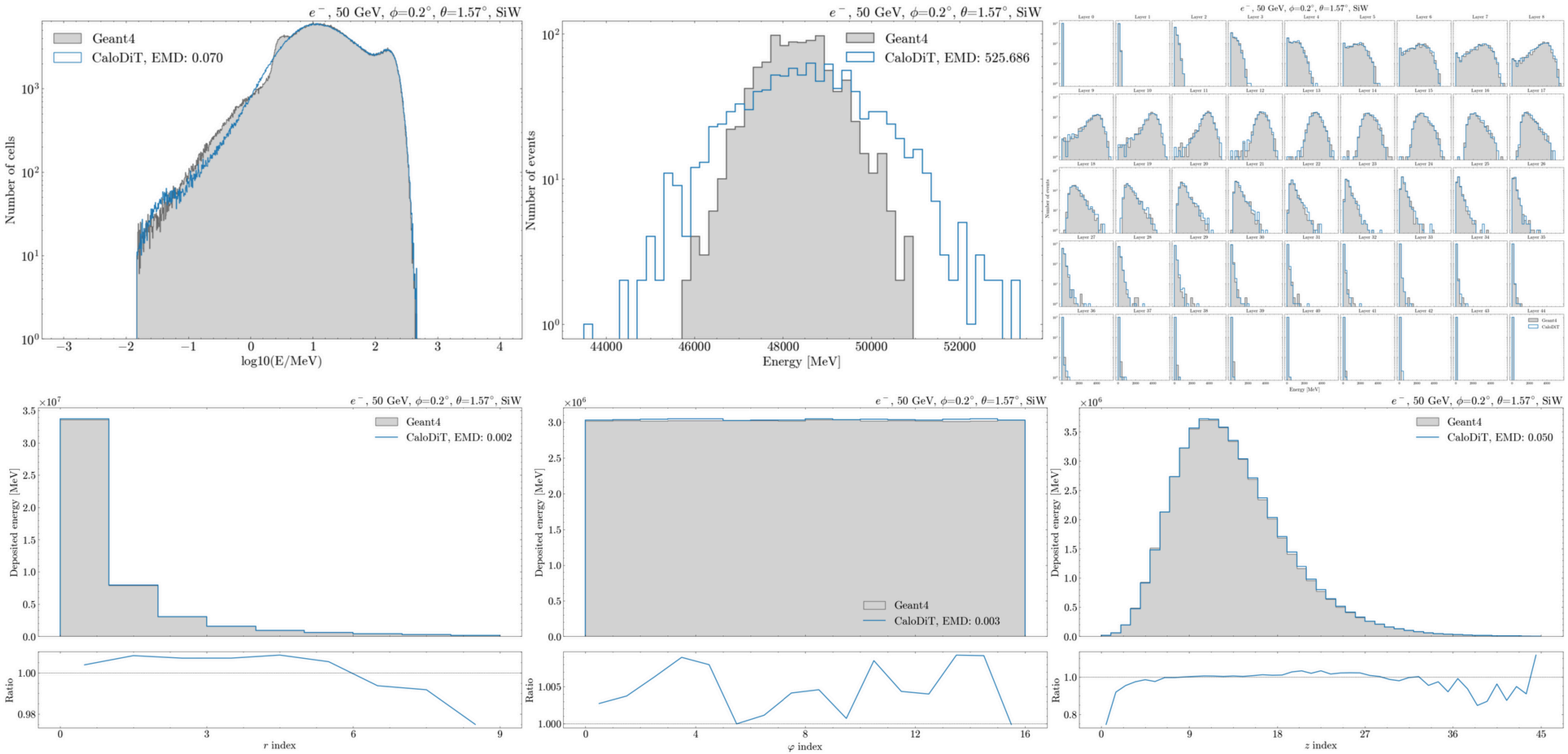
# Updates: LMS (Adams-Bashforth), 32 steps



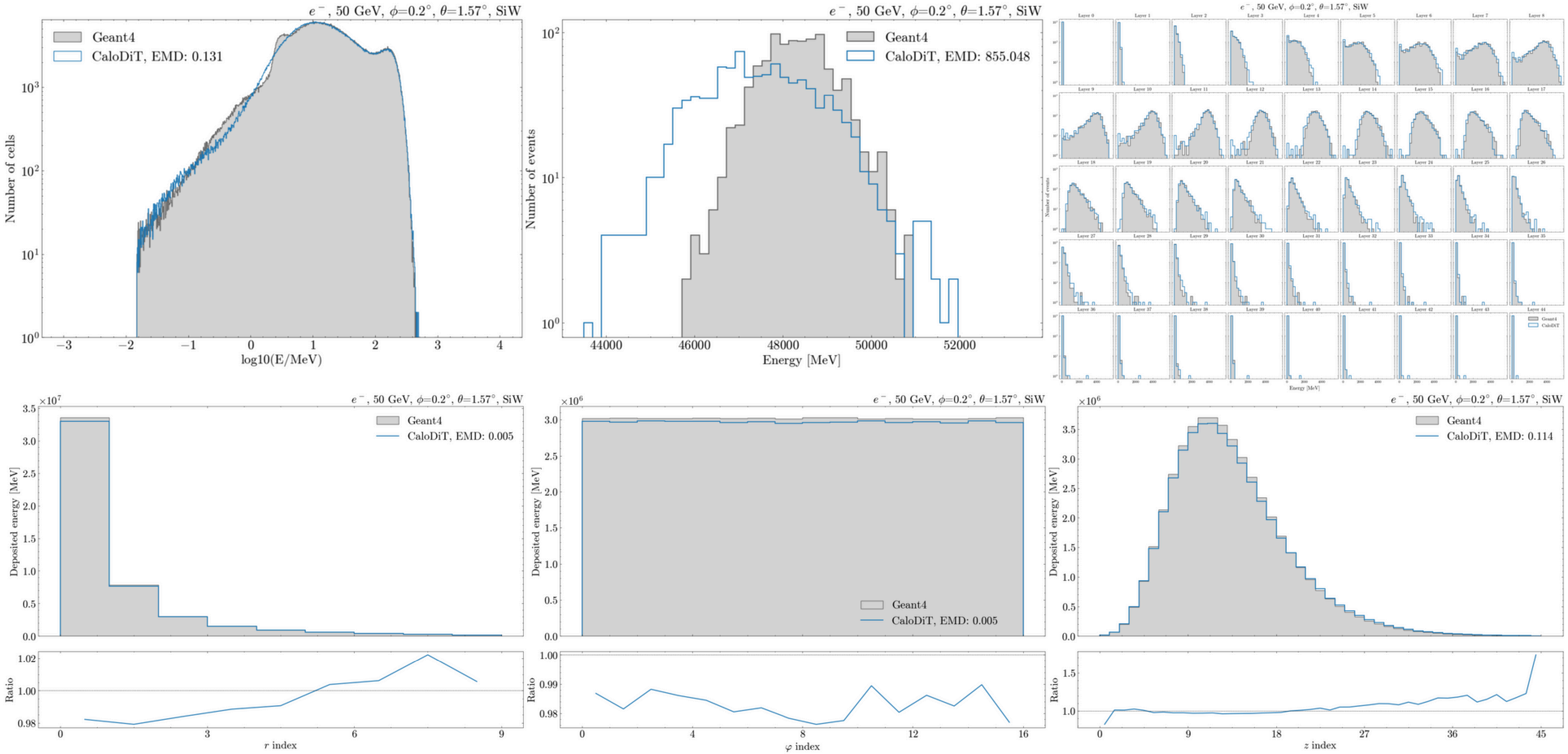
# Updates: Euler, 256 steps



# Updates: Heun, 255 steps



# Updates: LMS (Adams-Bashforth), 256 steps



# Next steps

1. Investigate why the model is underfitting (total event energy distribution, cell log-energy distribution)
2. Fix training with uniform energy distribution
3. Finnish the implementation of DPM-Solver++