(inductive) CaloFlow

Ian Pang

May 31, 2023

CaloChallenge Workshop 2023, Rome

RUTGERS
UNIVERSITY | NEW BRUNSWICK

[2210.14245] C. Krause, **IP**, D. Shih

[2305.11934] M. Buckley, C. Krause, **IP**, D. Shih

(inductive) CaloFlow for CaloChallenge

Dataset 1
$\mathcal{O}(10^2)$

Dataset 2
$\mathcal{O}(10^3)$

Dataset 3
$\mathcal{O}(10^4)$

(inductive) CaloFlow for CaloChallenge

Dataset 1
$\mathcal{O}(10^2)$

Dataset 2
$\mathcal{O}(10^3)$

Dataset 3
$\mathcal{O}(10^4)$

CaloFlow
[2210.14245]
C. Krause, **IP**, D. Shih

(inductive) CaloFlow for CaloChallenge

Dataset 1
$\mathcal{O}(10^2)$

Dataset 2
$\mathcal{O}(10^3)$

Dataset 3
$\mathcal{O}(10^4)$

CaloFlow
[2210.14245]
C. Krause, **IP**, D. Shih

Original CaloFlow
Krause, Shih [2106.05285, 2110.11377]
developed on CaloGAN dataset
Paganini, de Oliveira, Nachman [1712.10321, PRD]

(inductive) CaloFlow for CaloChallenge

Dataset 1
$\mathcal{O}(10^2)$

Dataset 2
$\mathcal{O}(10^3)$

Dataset 3
$\mathcal{O}(10^4)$

CaloFlow
[2210.14245]
C. Krause, **IP**, D. Shih

(inductive) CaloFlow for CaloChallenge

Dataset 1
$\mathcal{O}(10^2)$

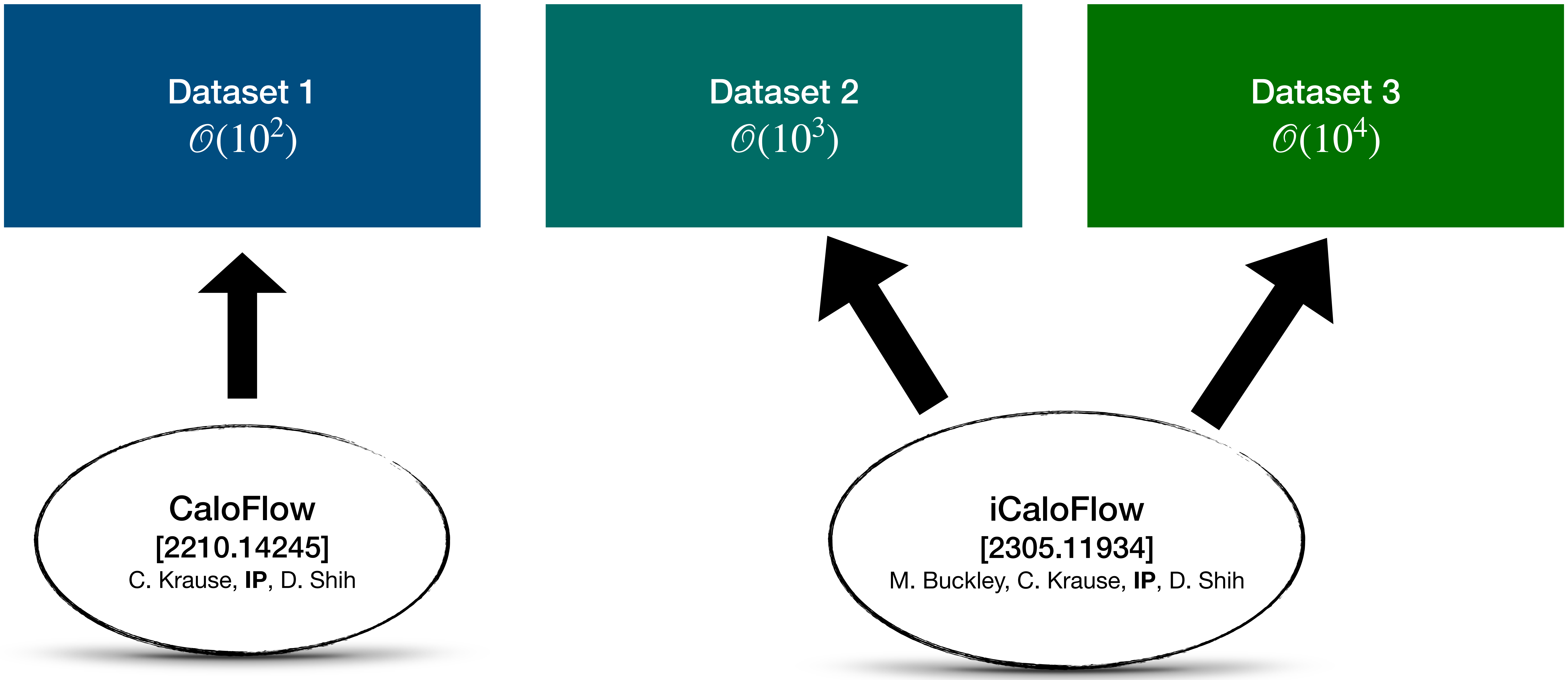
Dataset 2
$\mathcal{O}(10^3)$

Dataset 3
$\mathcal{O}(10^4)$

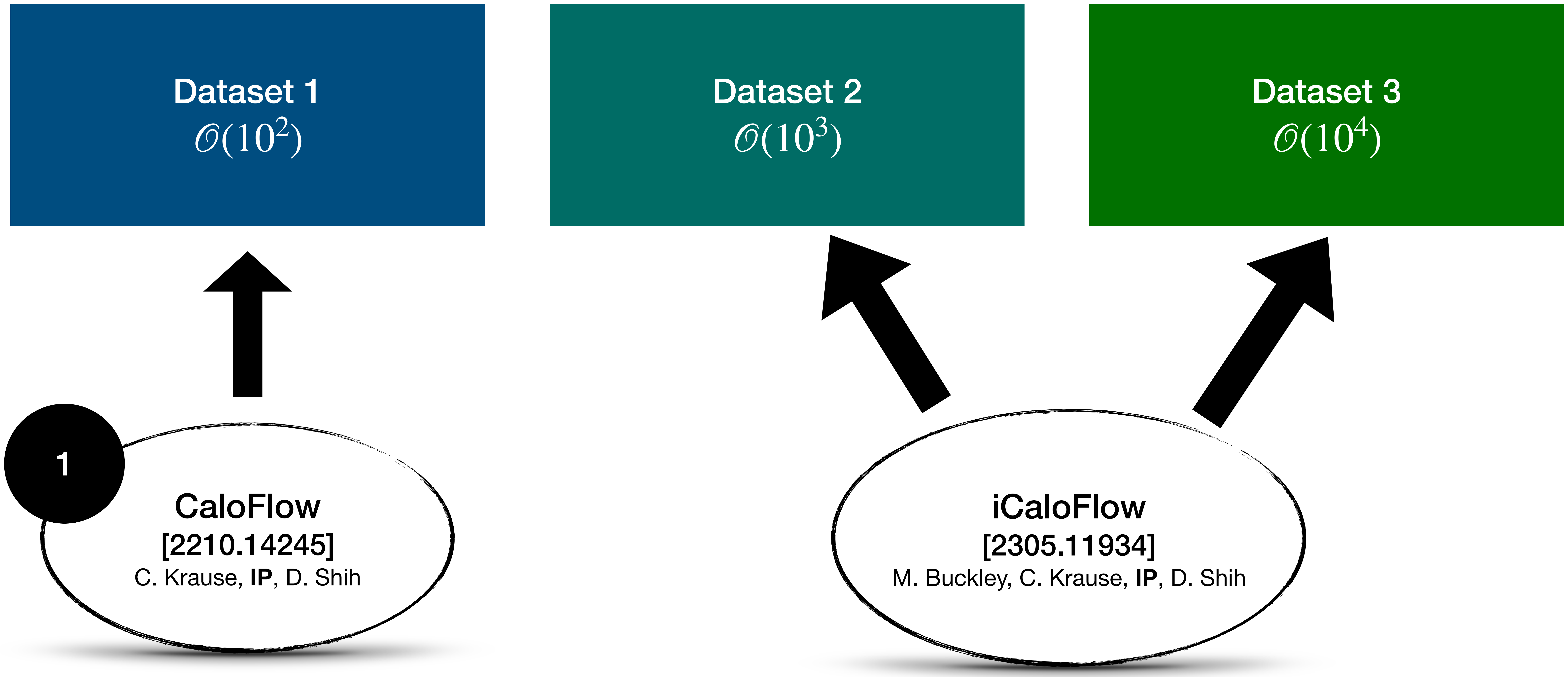
Training time doesn't scale well with N_{voxels}

CaloFlow
[2210.14245]
C. Krause, **IP**, D. Shih

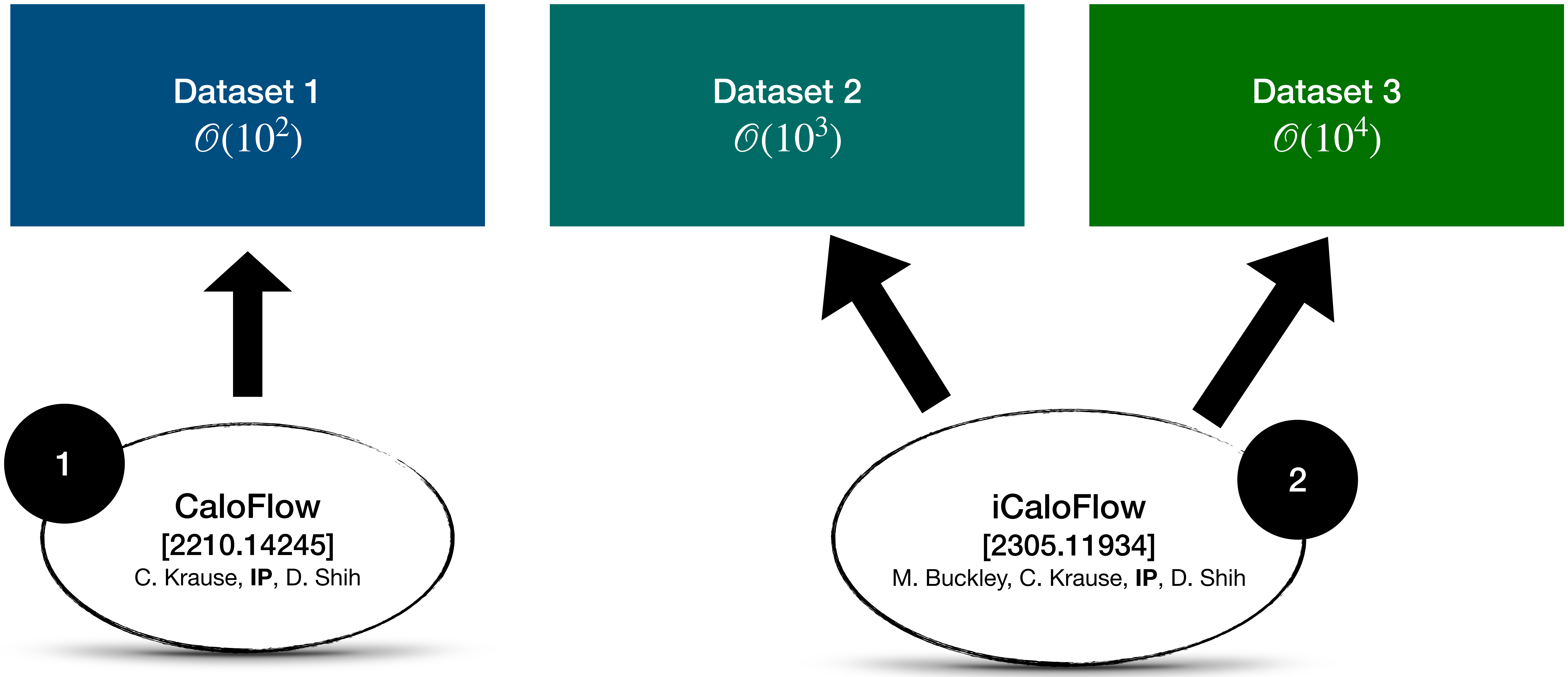
(inductive) CaloFlow for CaloChallenge



(inductive) CaloFlow for CaloChallenge

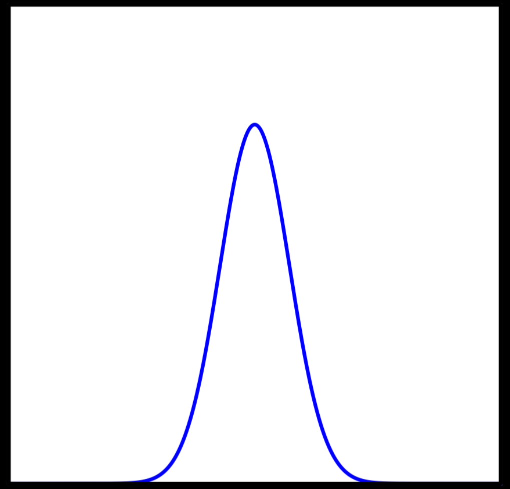


(inductive) CaloFlow for CaloChallenge



Normalizing Flows (Overview)

Latent space
with simple base distribution
z ~ Z

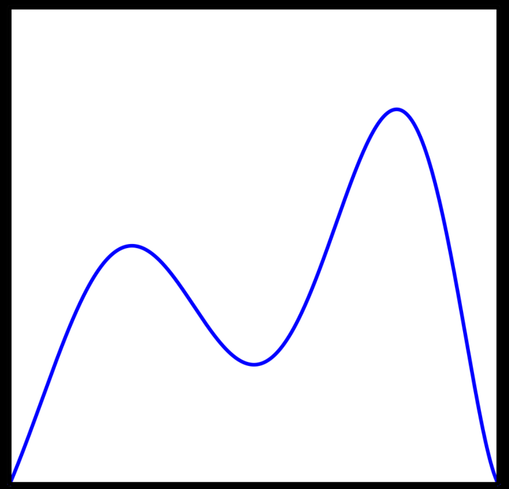


Bijective transformation f

$$p(x) = \pi(f(x))\left|\det\frac{\partial f(x)}{\partial x}\right|$$

where $f(x) = z$

Data space
with more complex distribution
x ~ X

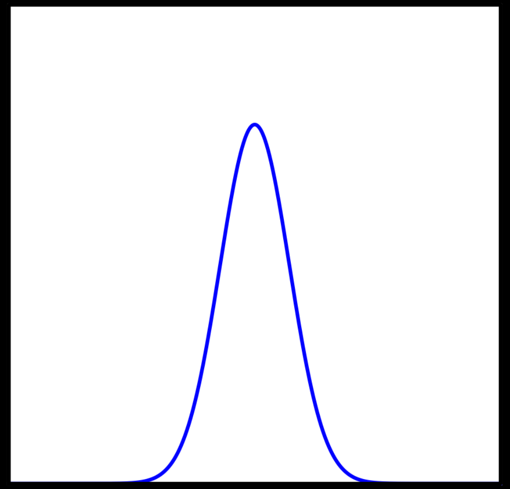


Density estimation, $p(x)$

Sample generation

Normalizing Flows (Overview)

Latent space
with simple base distribution
z ~ Z

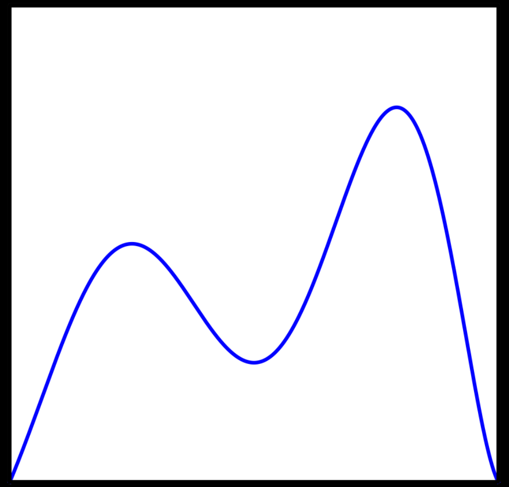


Bijective transformation f

$$p(x) = \pi(f(x)) \left| \det \frac{\partial f(x)}{\partial x} \right|$$

where $f(x) = z$

Data space
with more complex distribution
x ~ X



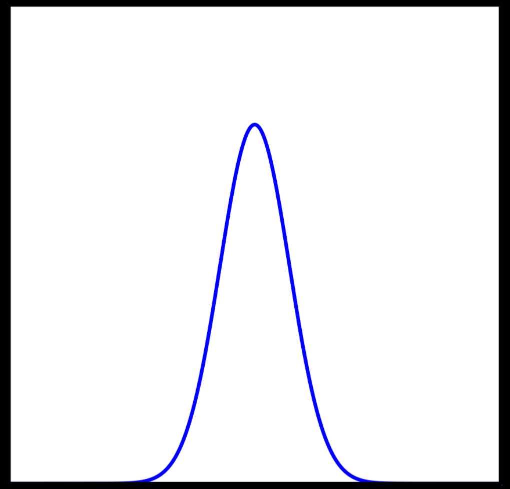
Needs to be tractable!

Density estimation, $p(x)$

Sample generation

Normalizing Flows (Overview)

Latent space
with simple base distribution
z ~ Z

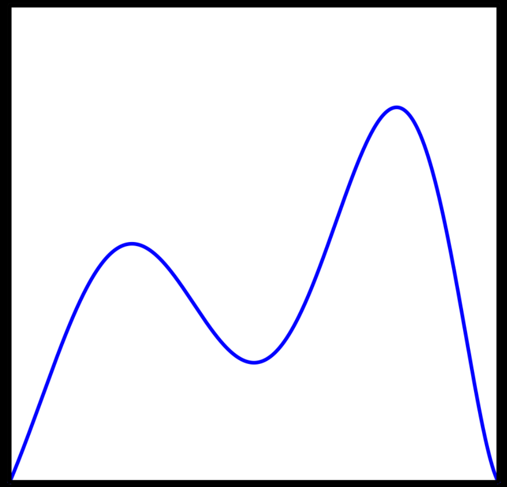


Bijective transformation f

$$p(x) = \pi(f(x)) \left| \det \frac{\partial f(x)}{\partial x} \right|$$

where $f(x) = z$

Data space
with more complex distribution
x ~ X



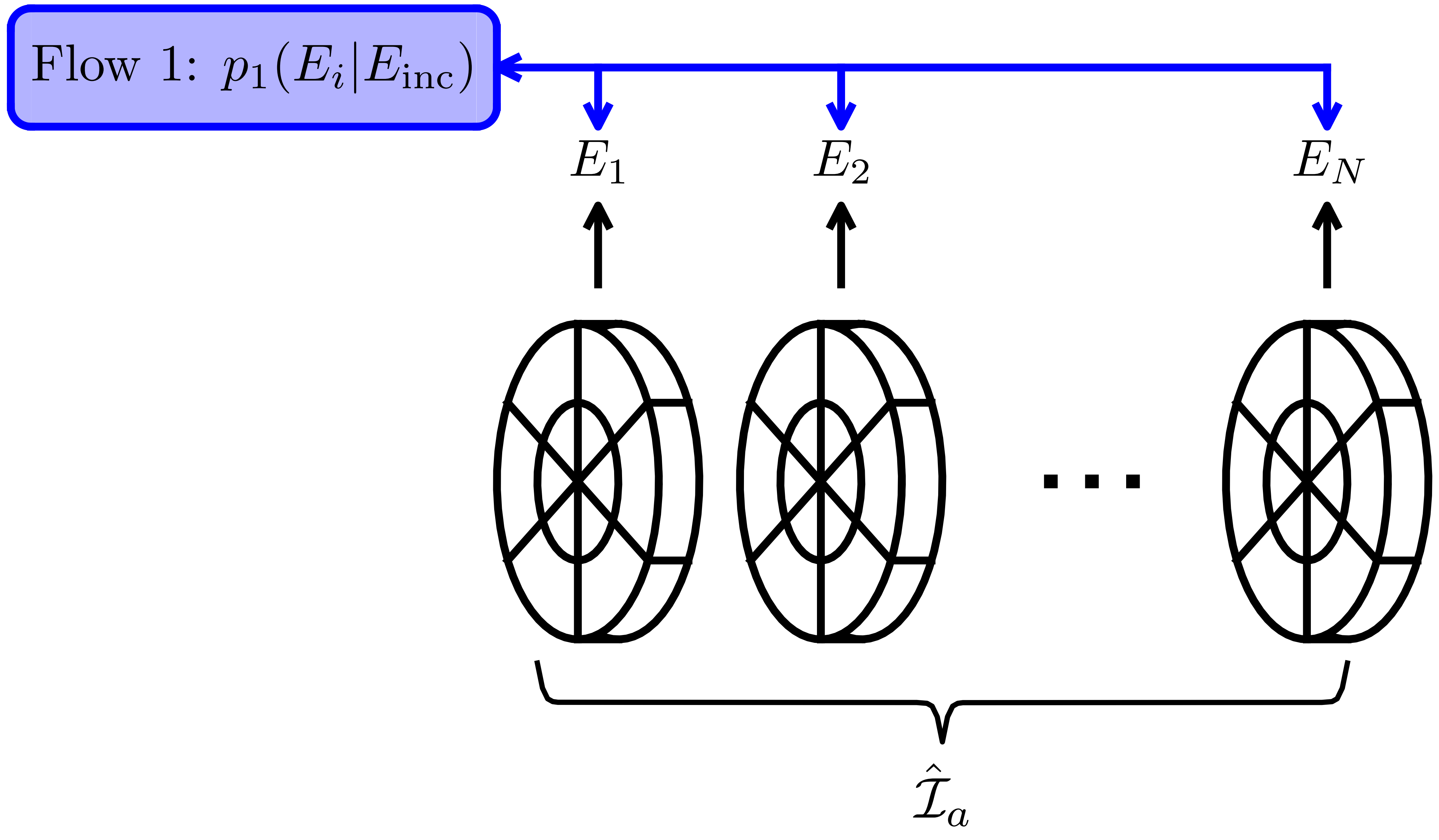
Density estimation, $p(x)$

Sample generation

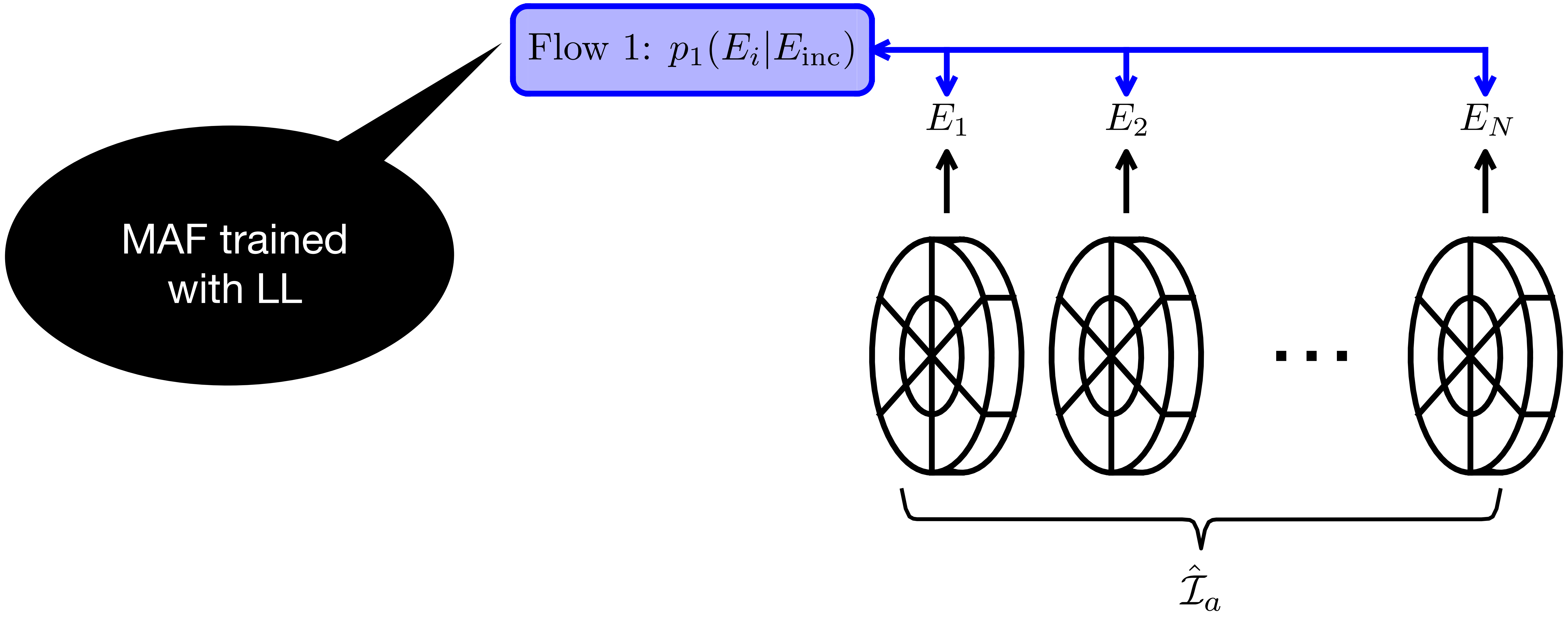
CaloFlow (Dataset 1)

- Rational Quadratic Splines chosen as transformations
  Durkan et al. [arXiv:1906.04032], Gregory/Delbourgo [IMA J. of Num. An., '82]
- Parameters θ of a composition of RQS are NN known as MADE blocks Germain et al. [1502.03509]
- Goal: Learn parameters θ

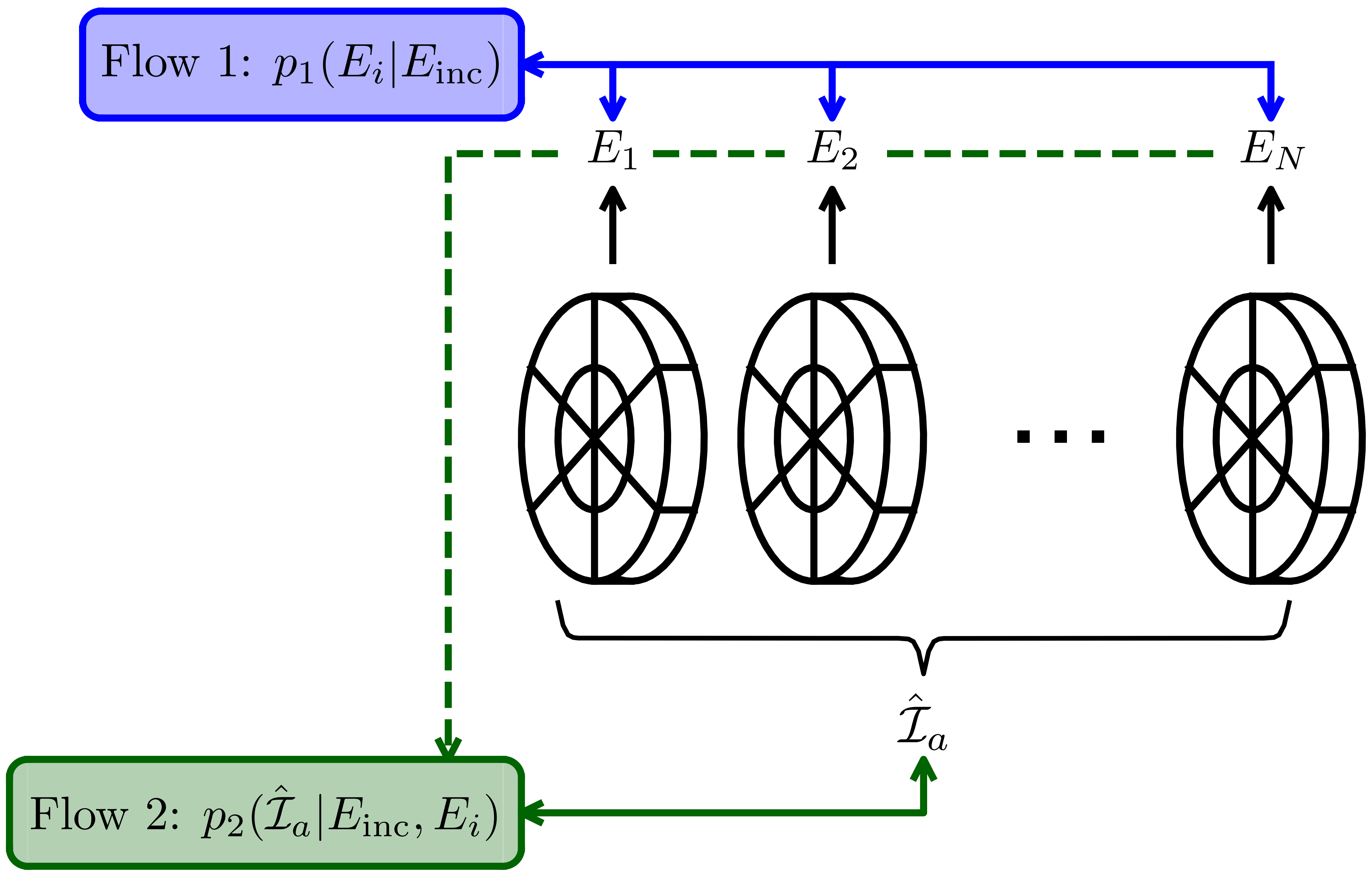
CaloFlow (Dataset 1)



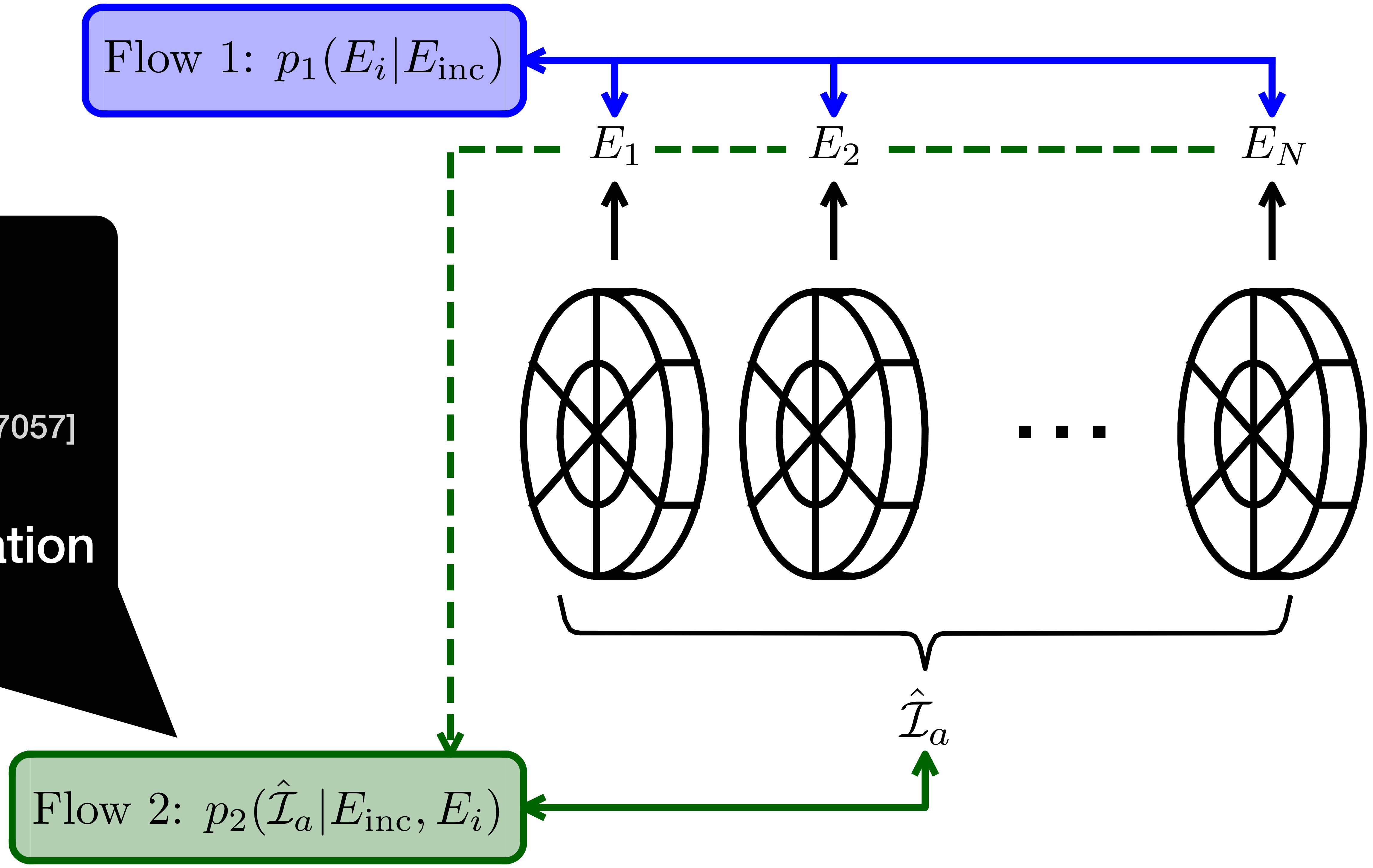
CaloFlow (Dataset 1)



CaloFlow (Dataset 1)



CaloFlow (Dataset 1)

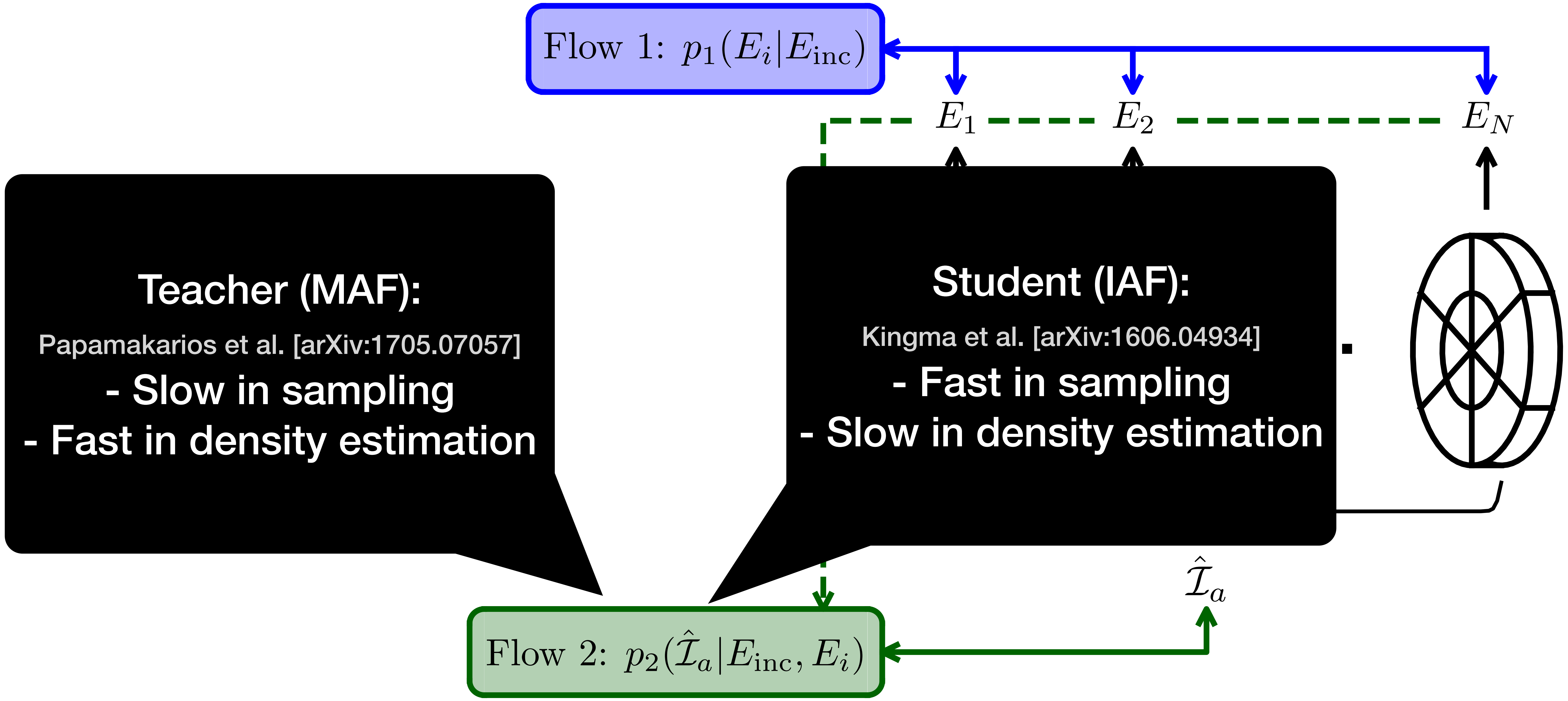


Teacher (MAF):

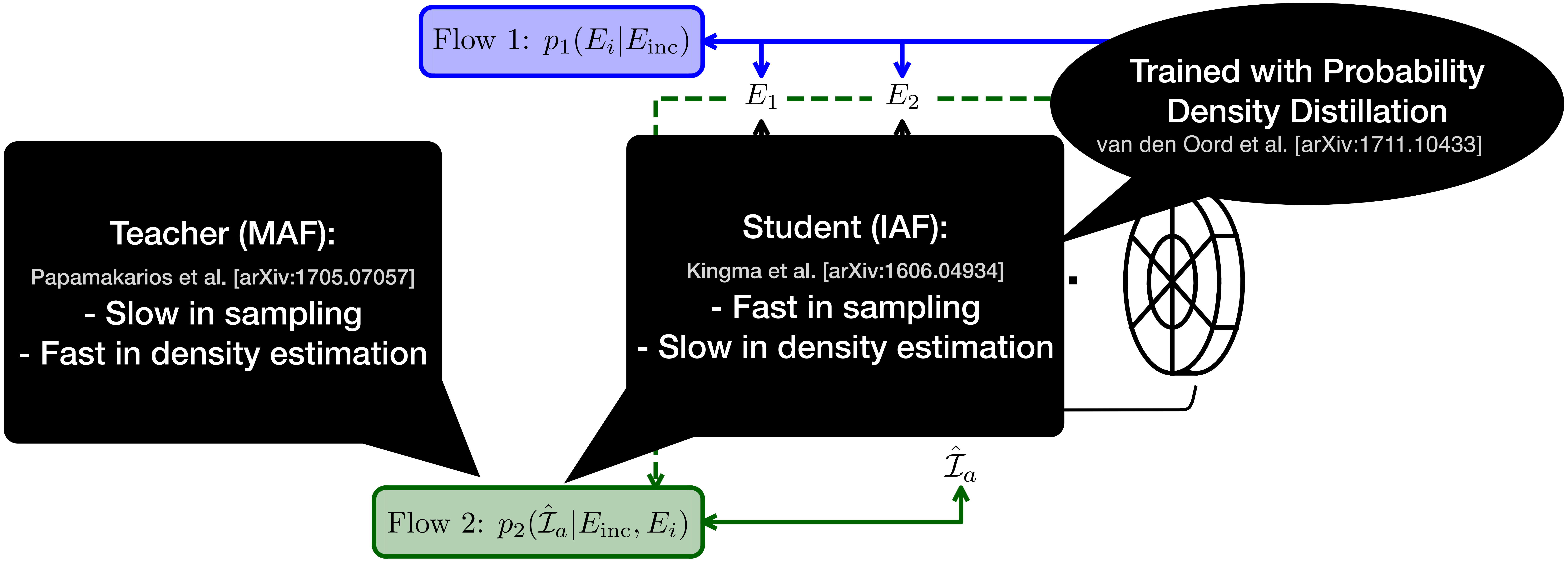
Papamakarios et al. [arXiv:1705.07057]

- Slow in sampling
- Fast in density estimation

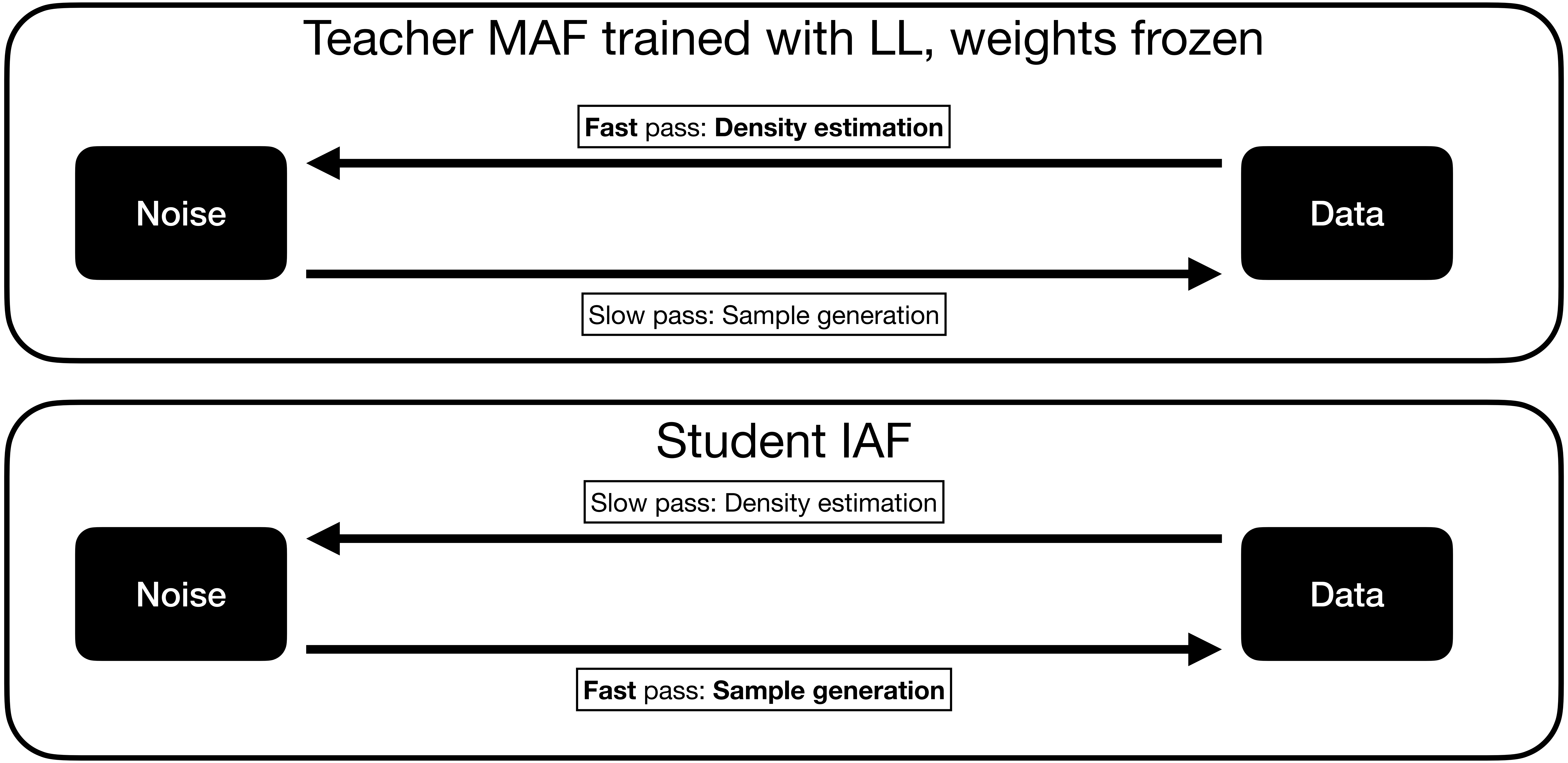
CaloFlow (Dataset 1)



CaloFlow (Dataset 1)

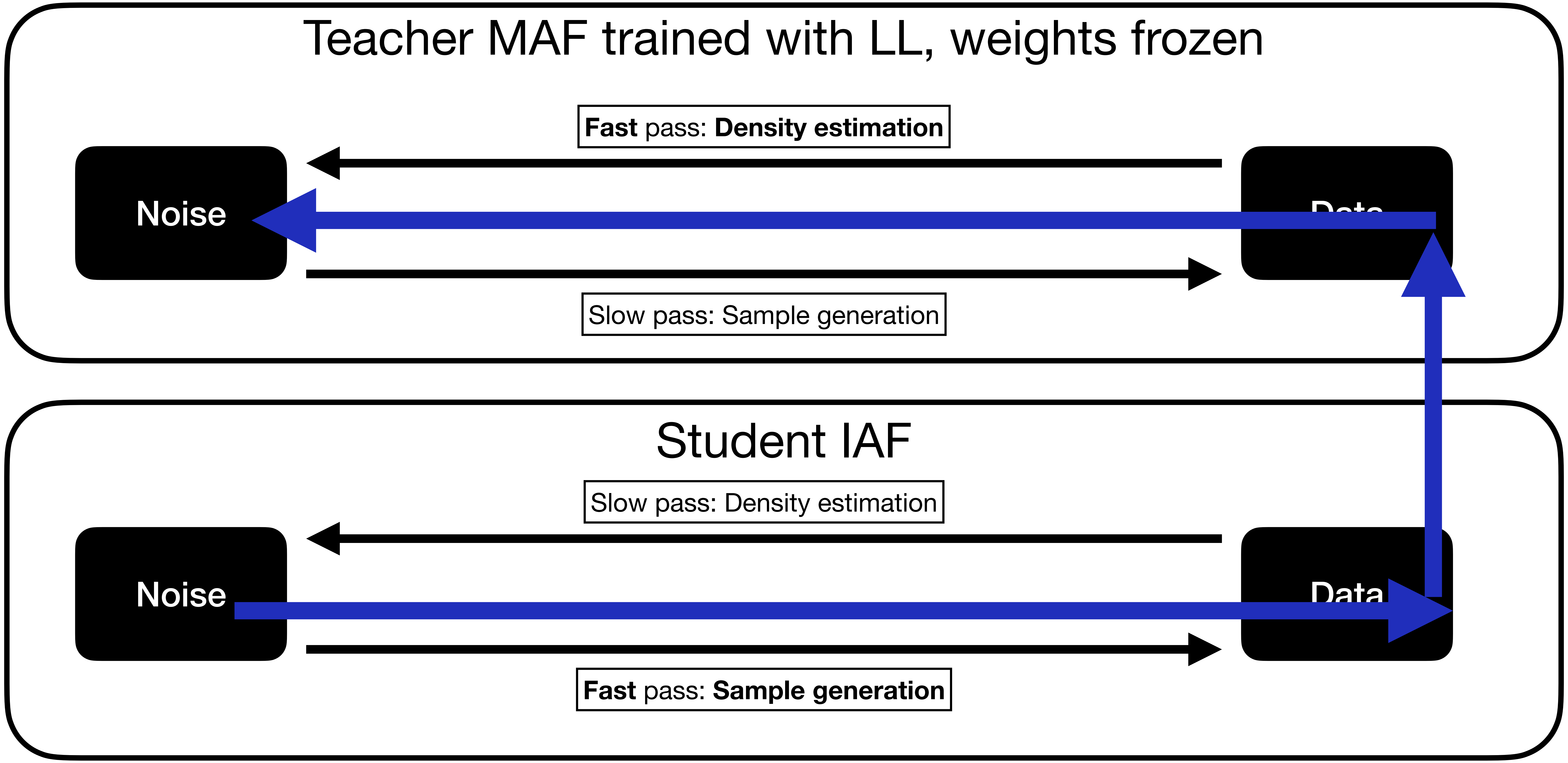


Probability Density Distillation



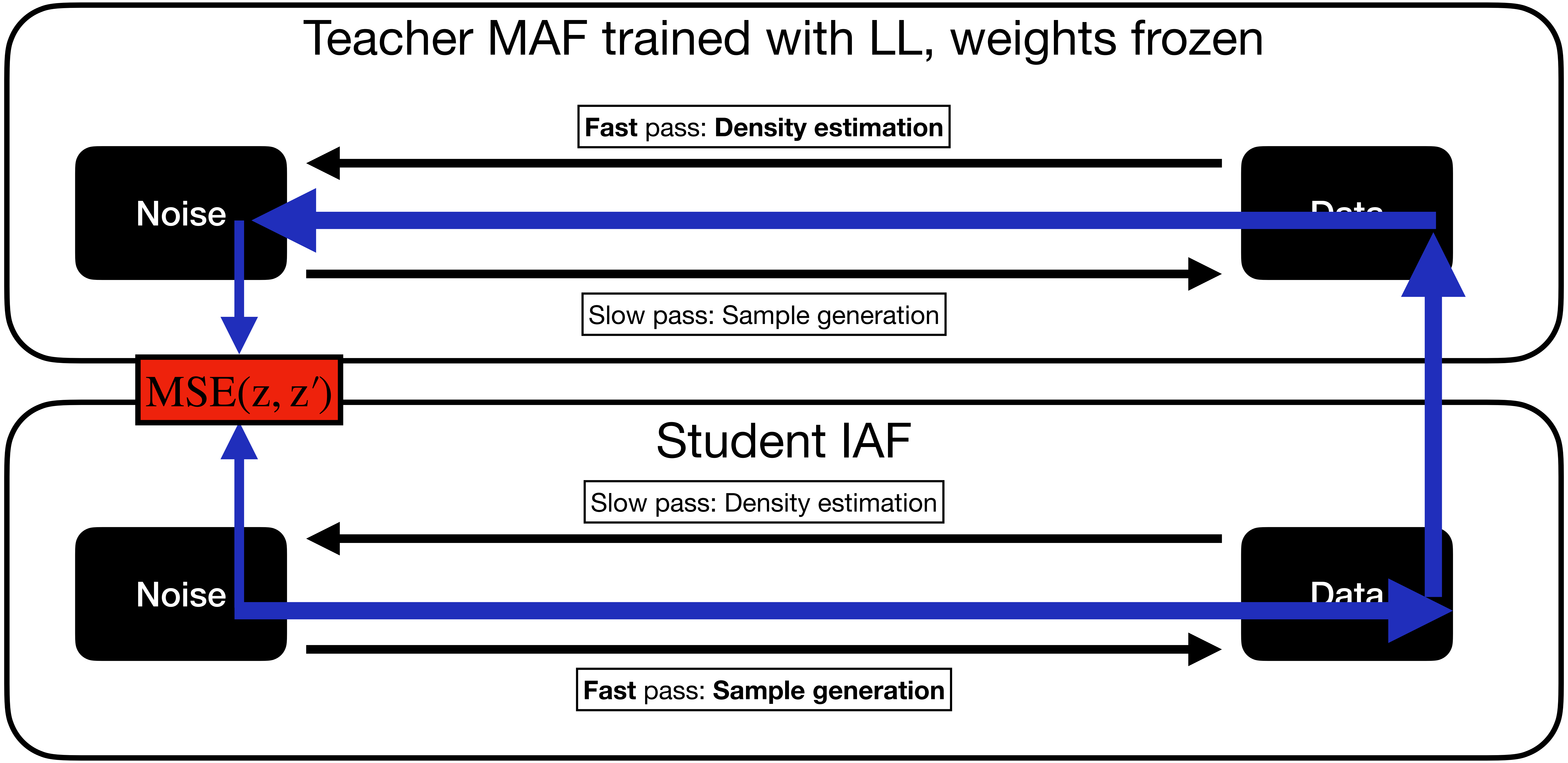
$$\mathrm{Loss} = \mathrm{MSE}(z, z') + \mathrm{MSE}(x, x') + \mathrm{MSE}(z_i, z_i')$$
$$+\mathrm{MSE}(x_i, x_i') + \mathrm{MSE}(p_z, p_z') + \mathrm{MSE}(p_x, p_x')$$

Probability Density Distillation



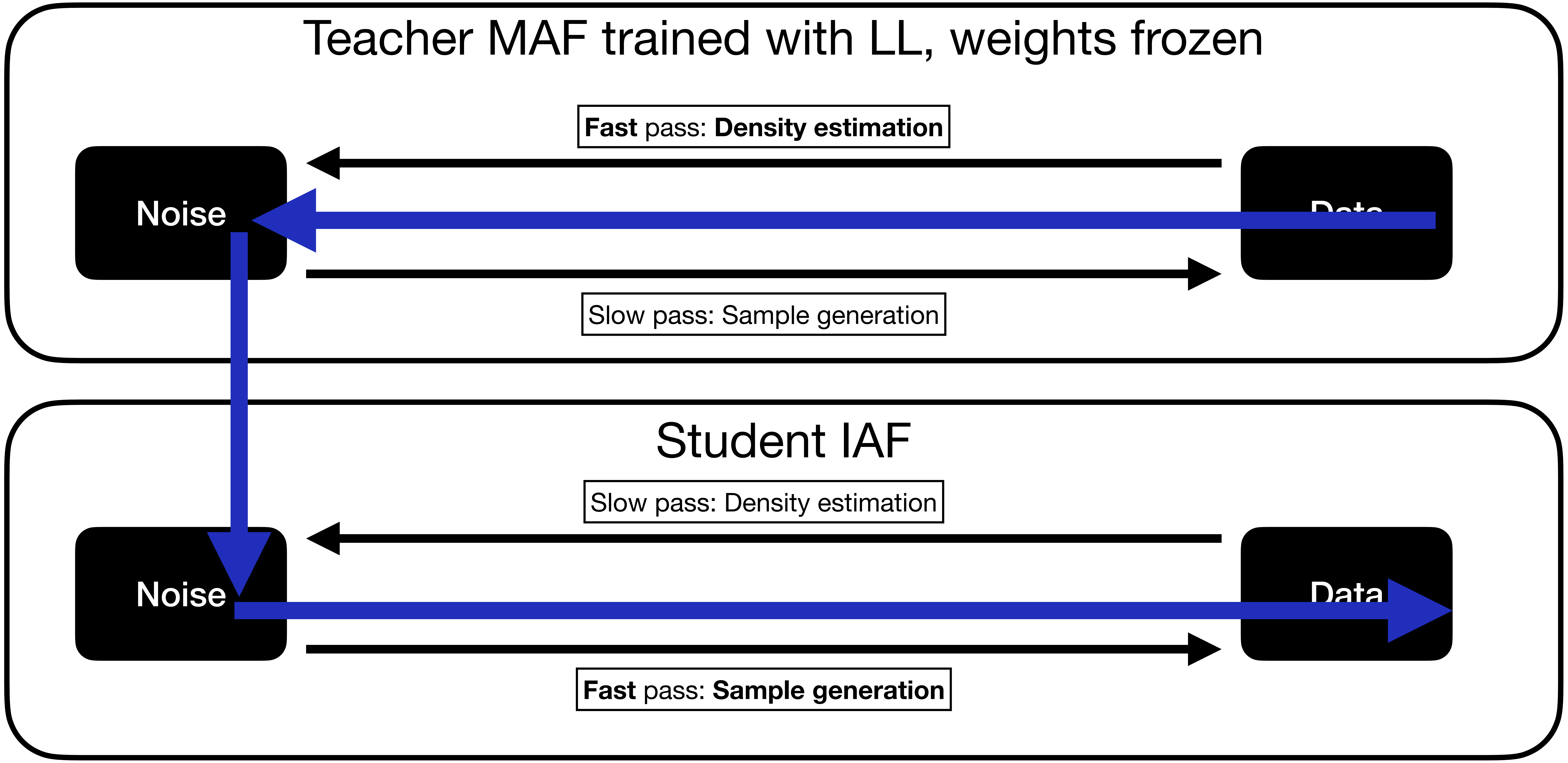
$$\text{Loss} = \text{MSE}(z, z') + \text{MSE}(x, x') + \text{MSE}(z_i, z_i')$$
$$+\text{MSE}(x_i, x_i') + \text{MSE}(p_z, p_z') + \text{MSE}(p_x, p_x')$$

Probability Density Distillation



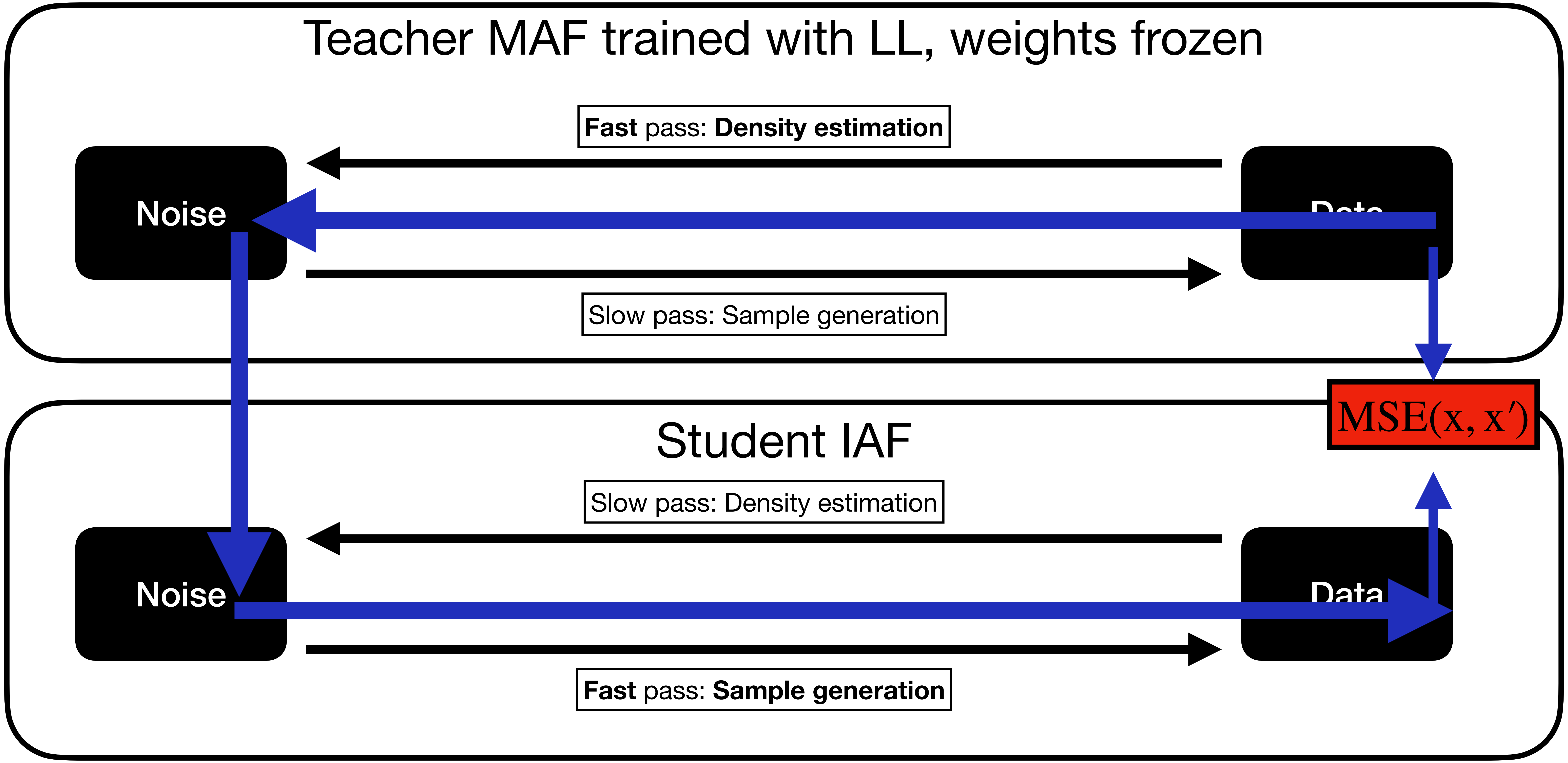
$$\mathrm{Loss} = \boxed{\mathrm{MSE}(z, z')} + \mathrm{MSE}(x, x') + \mathrm{MSE}(z_i, z_i') + \mathrm{MSE}(x_i, x_i') + \mathrm{MSE}(p_z, p_z') + \mathrm{MSE}(p_x, p_x')$$

Probability Density Distillation



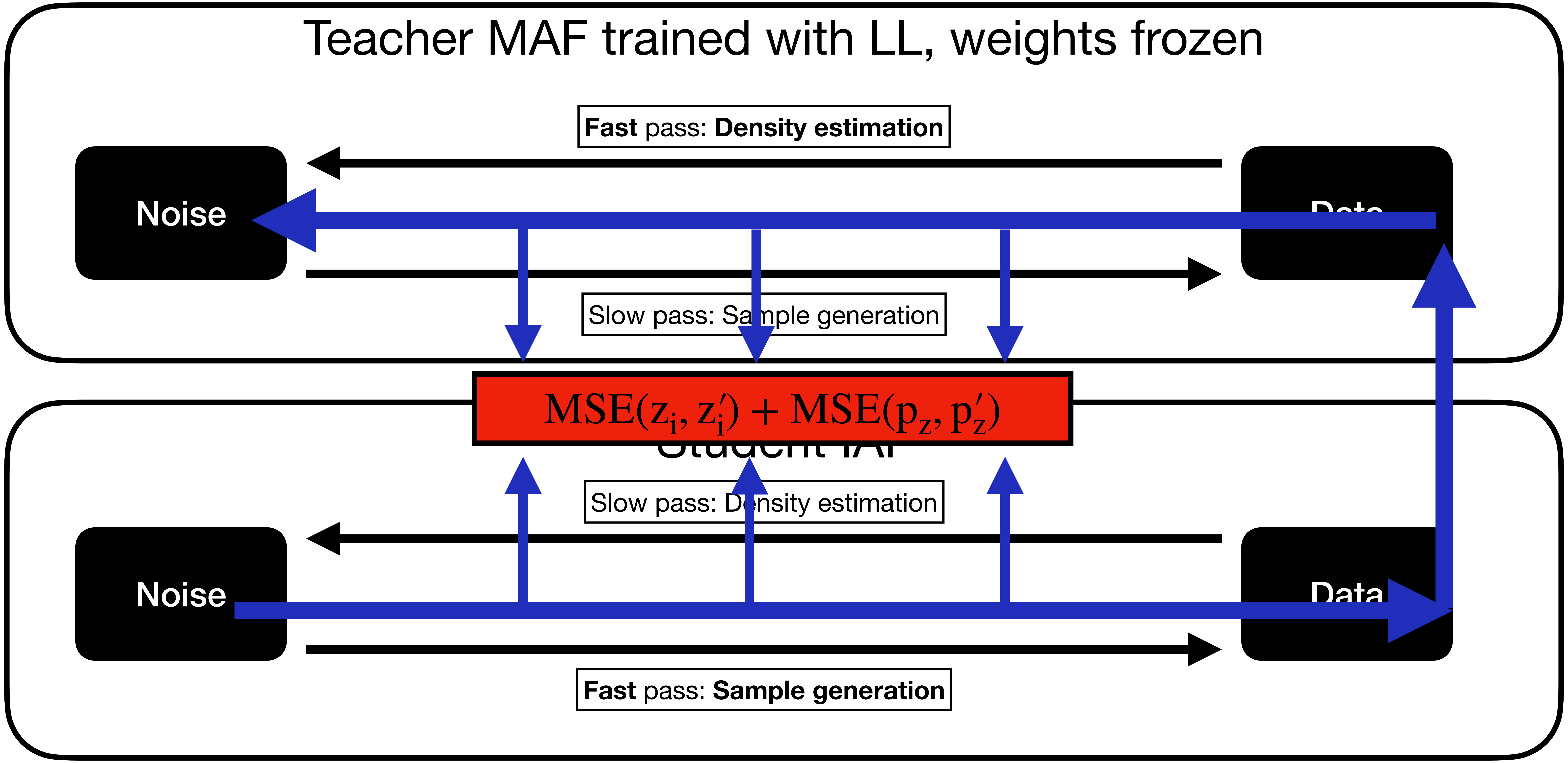
$$\mathrm{Loss} = \mathrm{MSE}(z, z') + \mathrm{MSE}(x, x') + \mathrm{MSE}(z_i, z_i')$$
$$+\mathrm{MSE}(x_i, x_i') + \mathrm{MSE}(p_z, p_z') + \mathrm{MSE}(p_x, p_x')$$

Probability Density Distillation



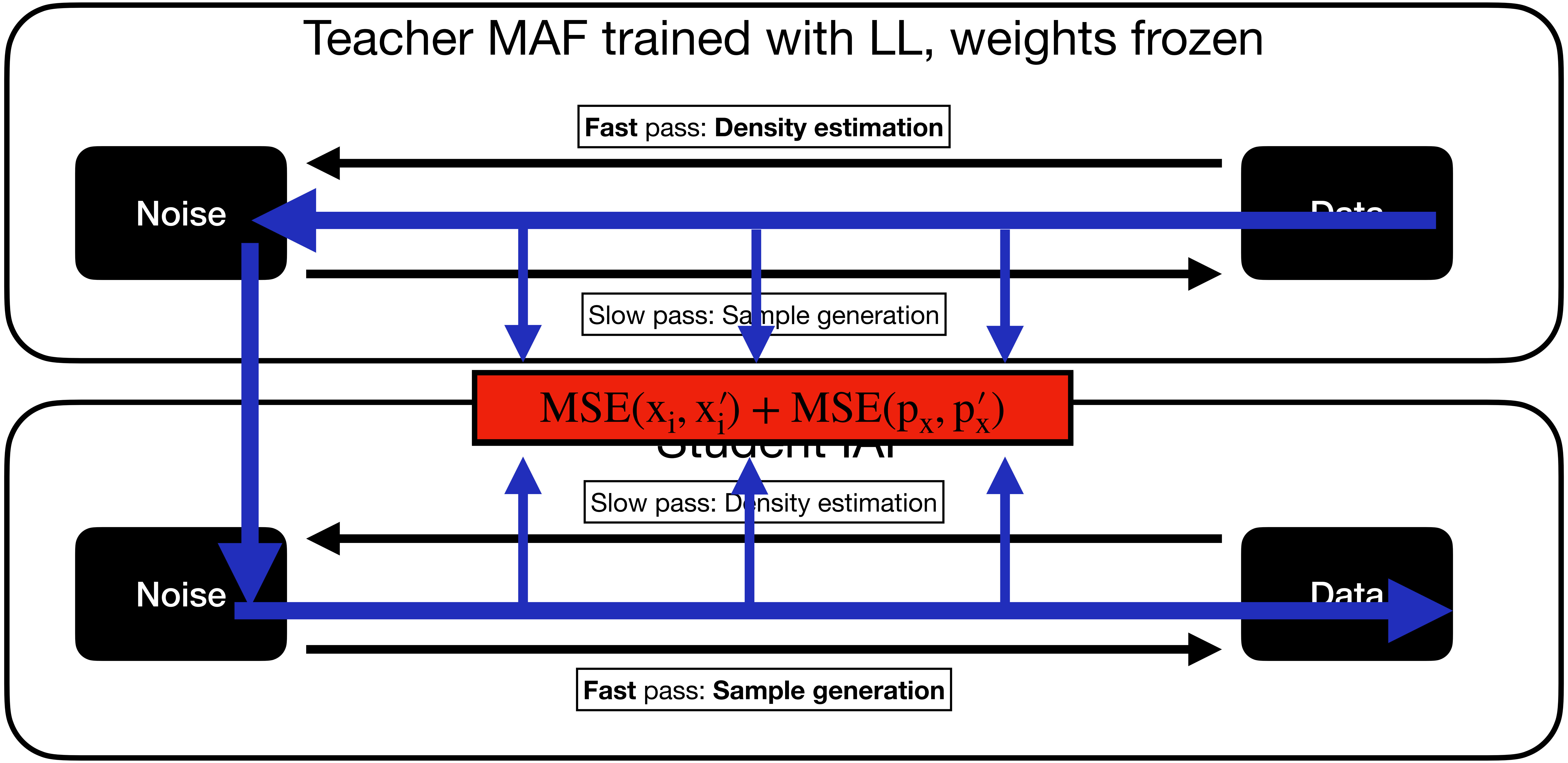
$$\mathrm{Loss} = \mathrm{MSE}(z, z') + \boxed{\mathrm{MSE}(x, x')} + \mathrm{MSE}(z_i, z_i') + \mathrm{MSE}(x_i, x_i') + \mathrm{MSE}(p_z, p_z') + \mathrm{MSE}(p_x, p_x')$$

Probability Density Distillation



$$\text{Loss} = \text{MSE}(z, z') + \text{MSE}(x, x') + \boxed{\text{MSE}(z_i, z_i')}$$
$$+\text{MSE}(x_i, x_i') + \boxed{\text{MSE}(p_z, p_z')} + \text{MSE}(p_x, p_x')$$

Probability Density Distillation



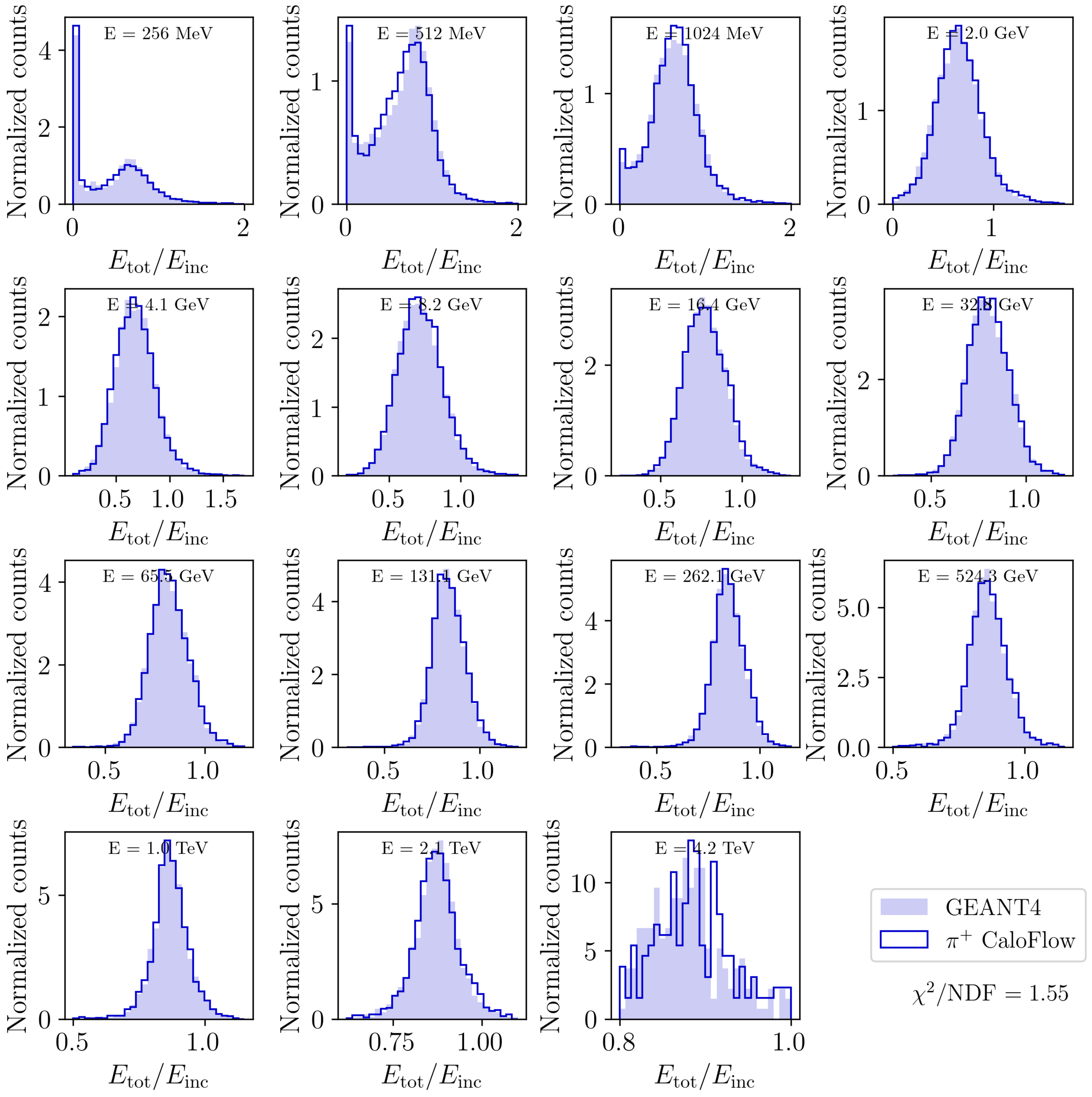
$$\mathrm{Loss} = \mathrm{MSE}(z, z') + \mathrm{MSE}(x, x') + \mathrm{MSE}(z_i, z_i') + \boxed{\mathrm{MSE}(x_i, x_i')} + \mathrm{MSE}(p_z, p_z') + \boxed{\mathrm{MSE}(p_x, p_x')}$$

Classifier scores (DS 1)

AUC / JSD		DNN based classifier	
		GEANT4 vs. CALOFLOW (teacher)	GEANT4 vs. CALOFLOW (student)
γ	low-level	0.701(3) / 0.092(3)	0.739(3) / 0.131(4)
	high-level	0.551(3) / 0.013(2)	0.556(3) / 0.015(2)
π^+	low-level	0.779(1) / 0.185(2)	0.854(3) / 0.313(6)
	high-level	0.698(2) / 0.104(3)	0.726(3) / 0.128(3)

All AUC and JSD much below 1 $\Longrightarrow$ High fidelity

π^+ E_{tot}/E_{inc}



E = 256 MeV
E = 512 MeV
E = 1024 MeV
E = 2.0 GeV
E = 4.1 GeV
E = 8.2 GeV
E = 16.4 GeV
E = 32.8 GeV
E = 65.5 GeV
E = 131.1 GeV
E = 262.1 GeV
E = 524.3 GeV
E = 1.0 TeV
E = 2.1 TeV
E = 4.2 TeV
Normalized counts
$E_{\rm tot}/E_{\rm inc}$
GEANT4
π^+ CaloFlow
χ^2/NDF = 1.55

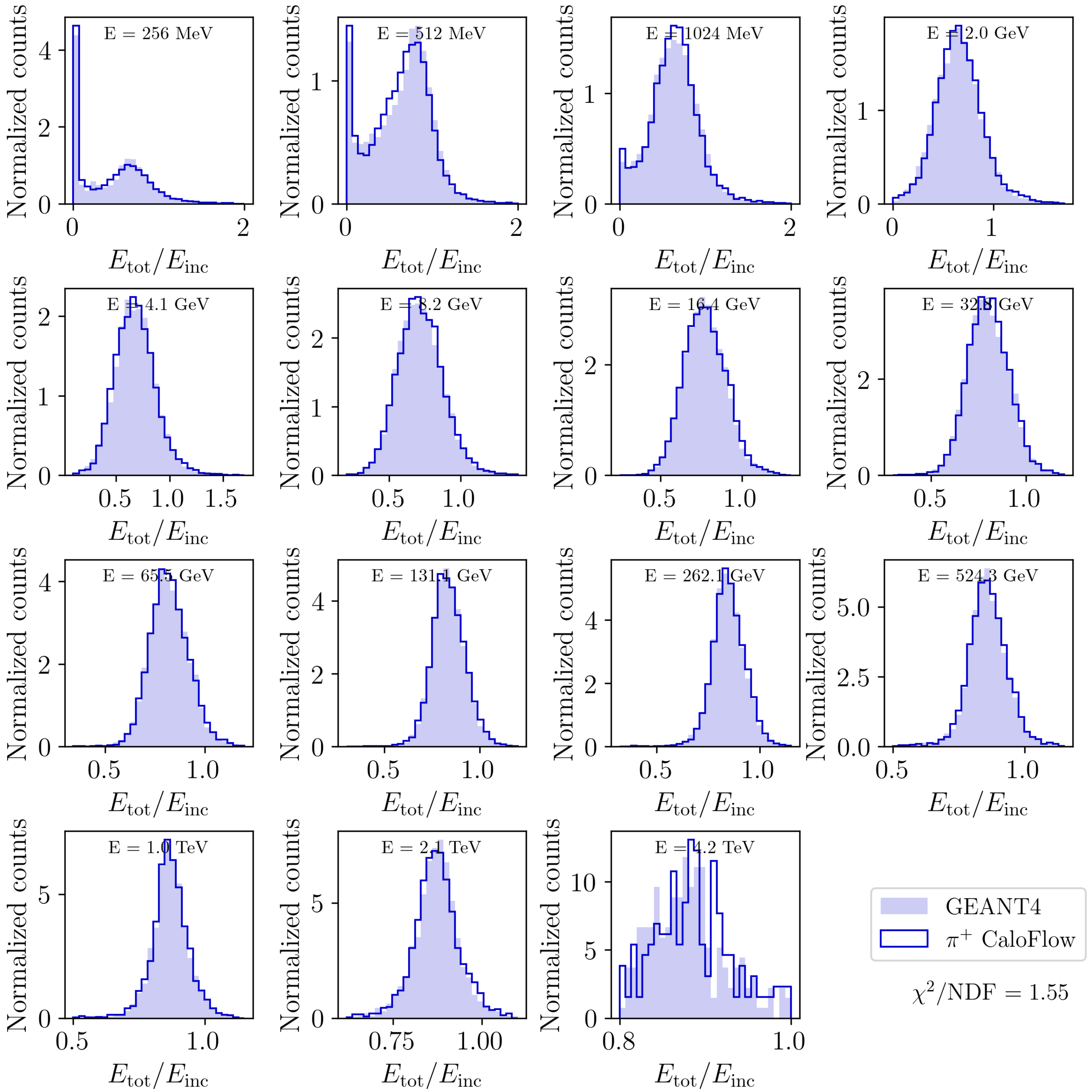
π^+ E_{tot}/E_{inc}

π^+ flow-1:

Uniform random of noise **[0, 0.1] keV** added to voxels

Other CF models:

Uniform random of noise **[0, 1] keV** added to voxels



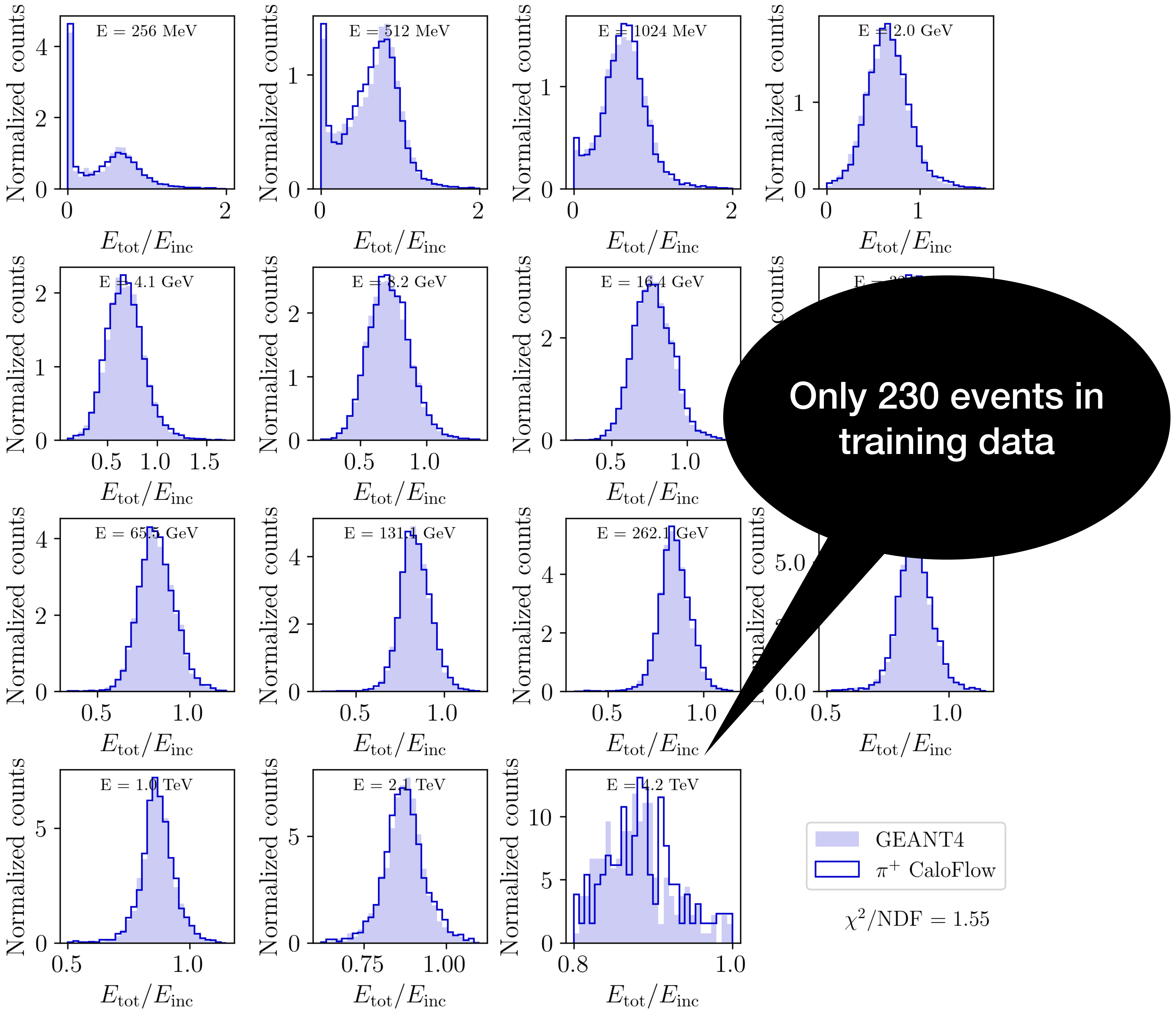
π^+ E_{tot}/E_{inc}

π^+ flow-1:

Uniform random of noise **[0, 0.1] keV** added to voxels

Other CF models:

Uniform random of noise **[0, 1] keV** added to voxels



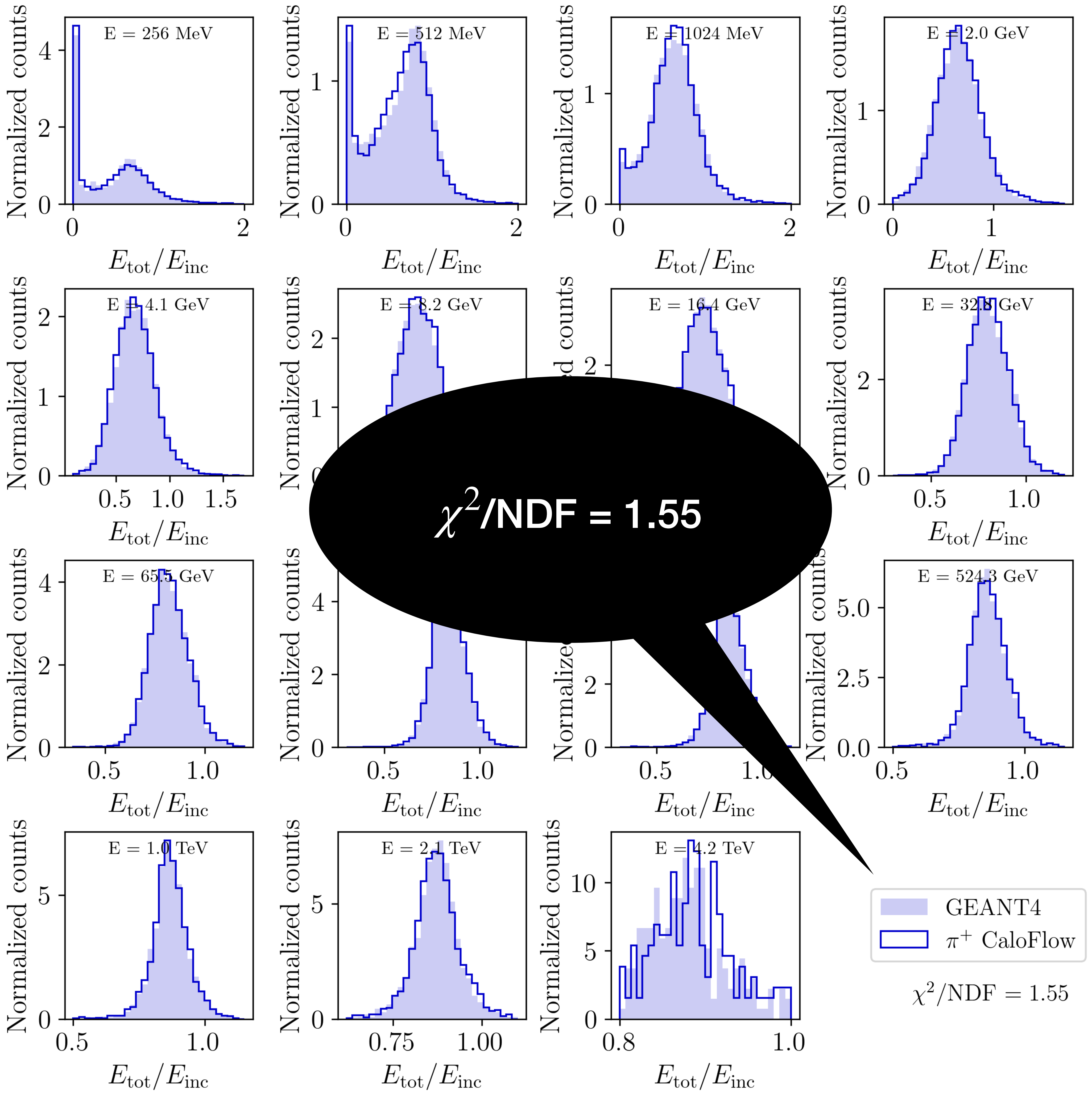
π^+ E_{tot}/E_{inc}

π^+ flow-1:

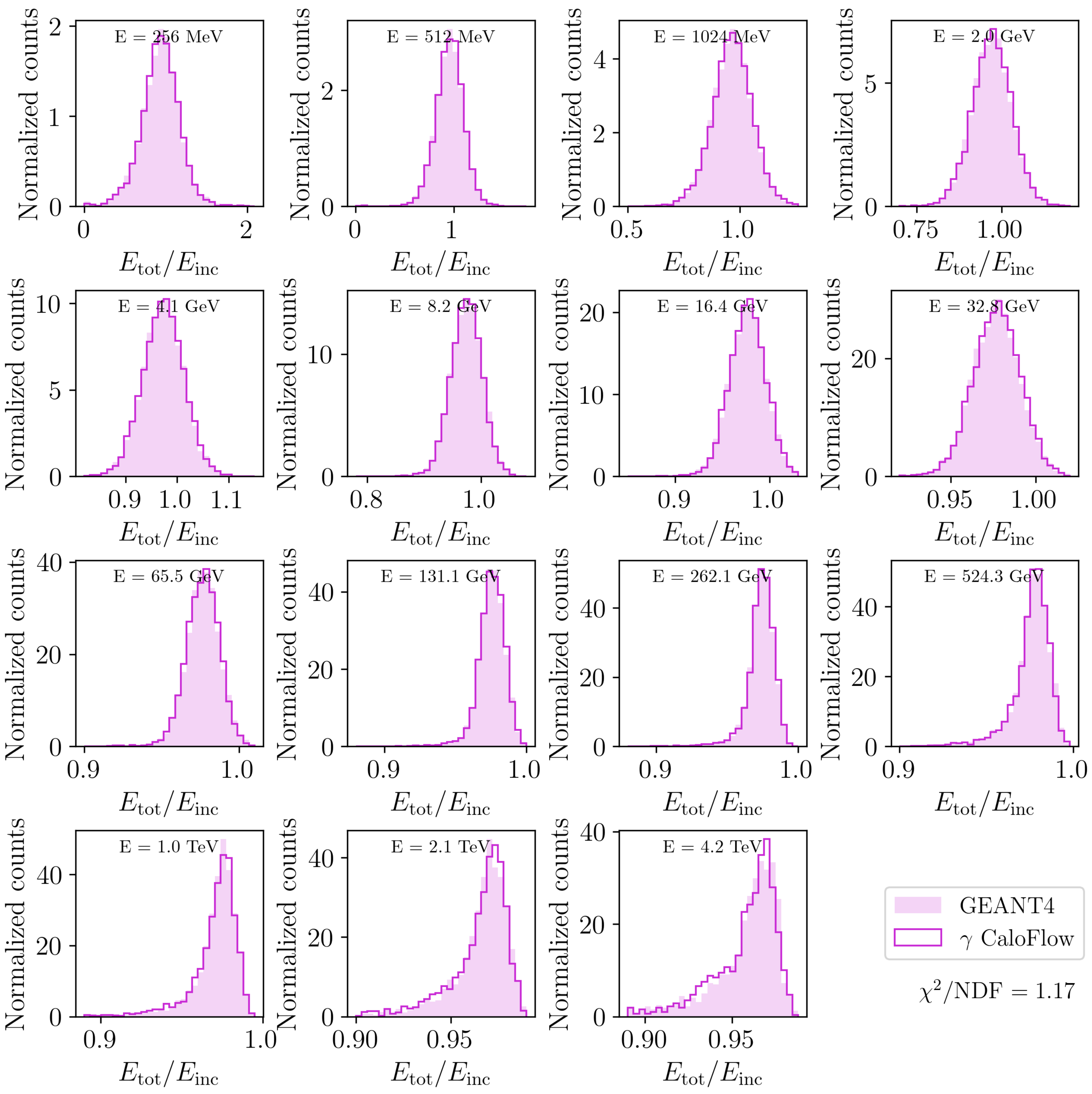
Uniform random of noise **[0, 0.1] keV** added to voxels

Other CF models:

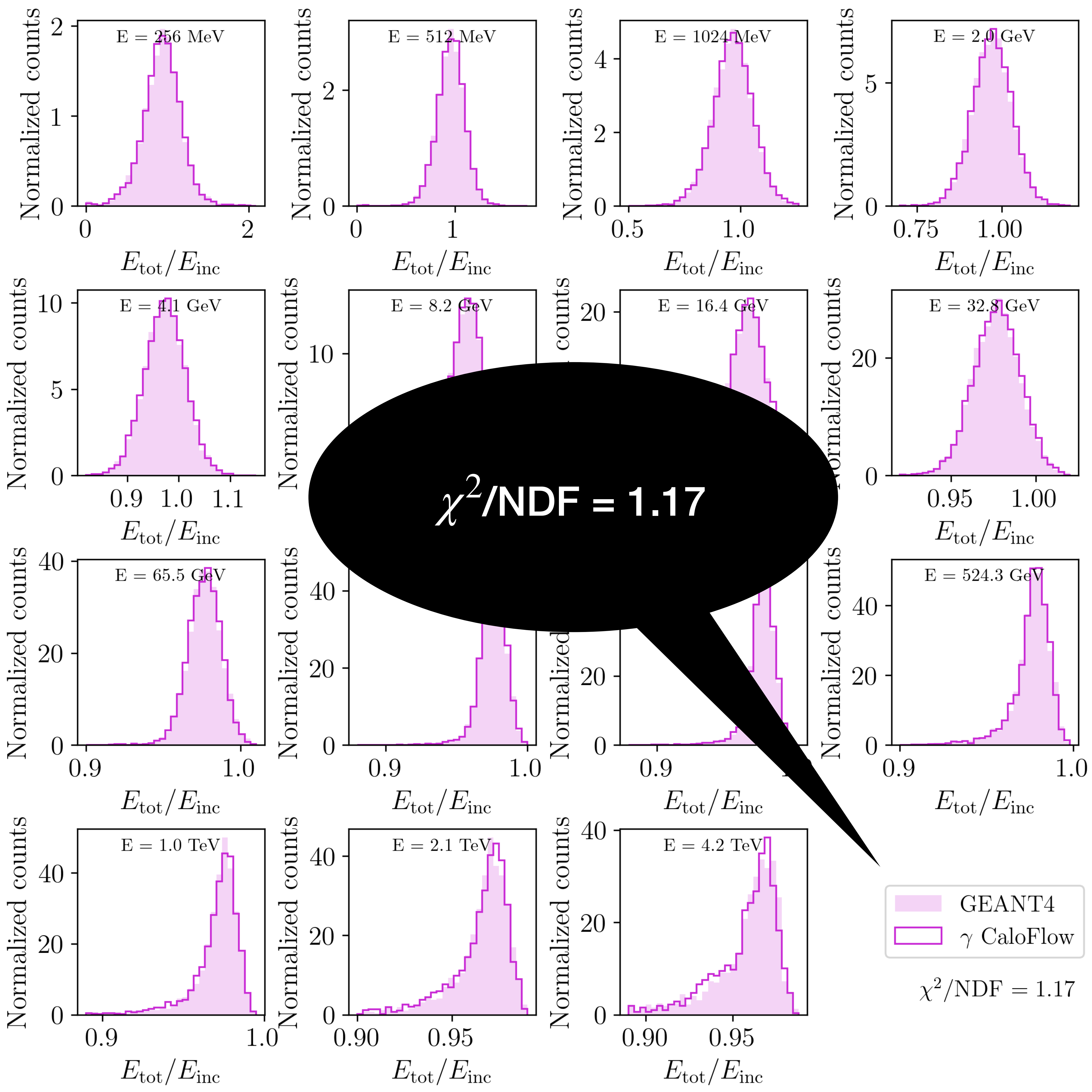
Uniform random of noise **[0, 1] keV** added to voxels



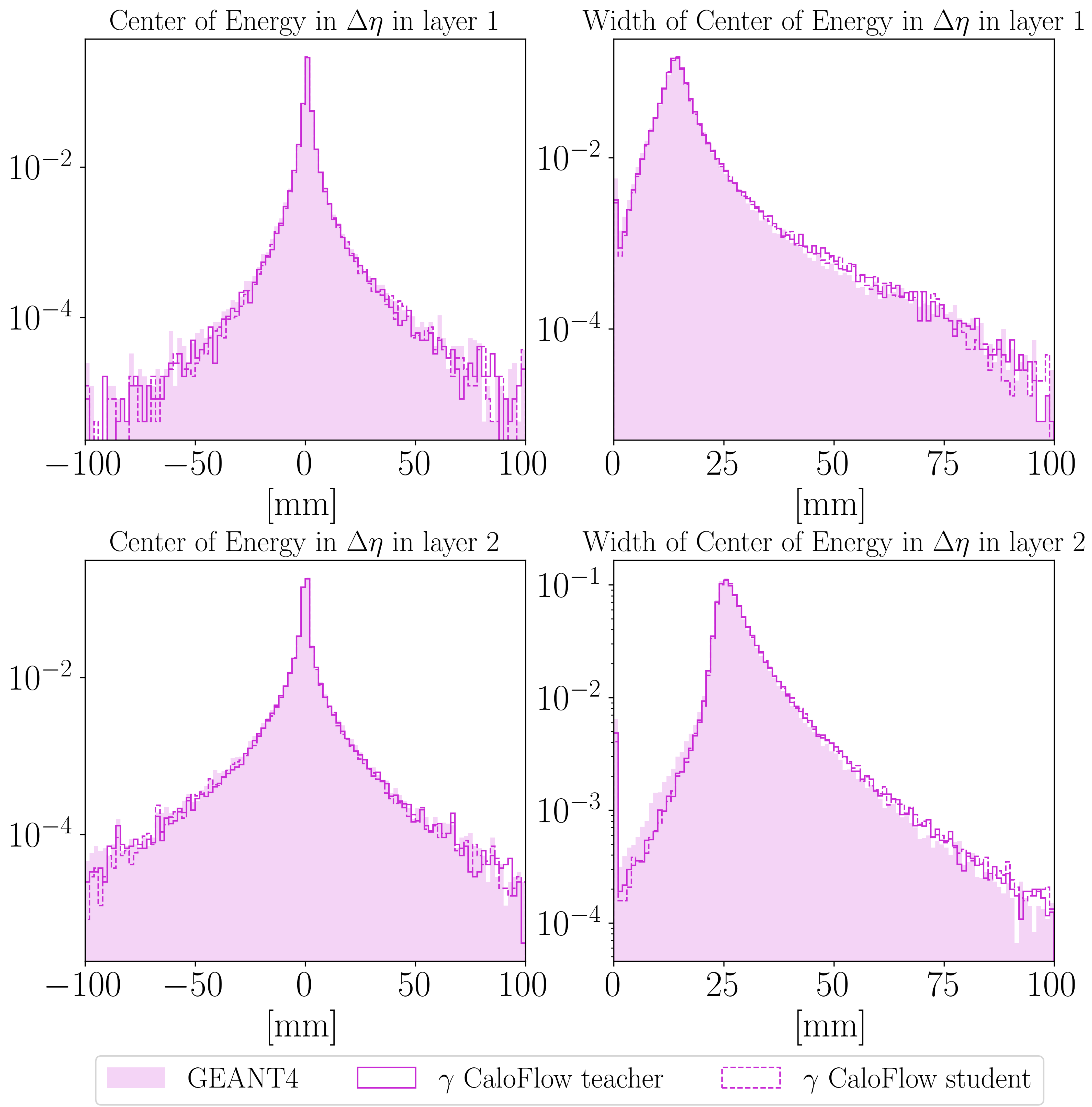
γ E_{tot}/E_{inc}



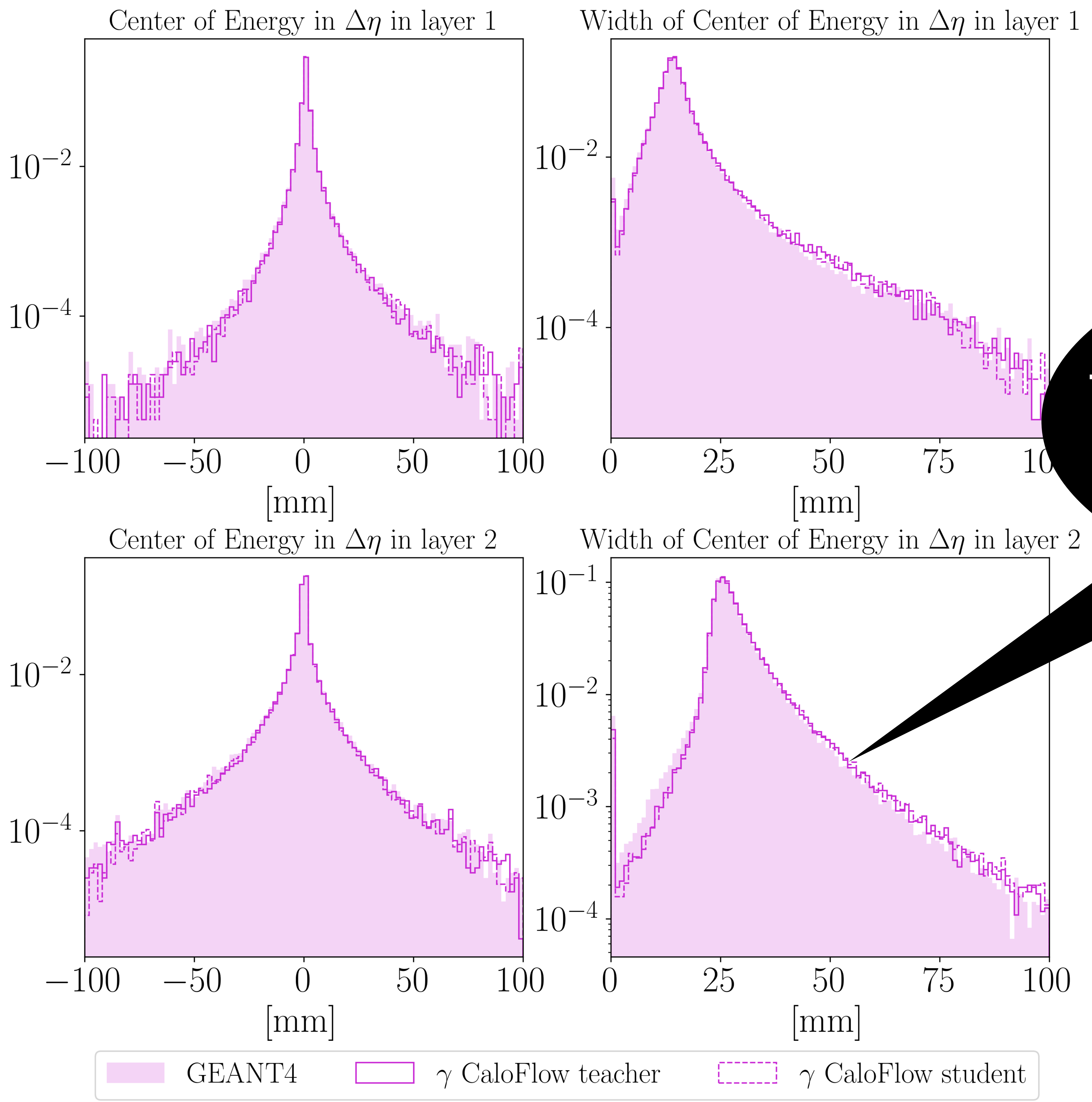
γ E_{tot}/E_{inc}



γ shower shape

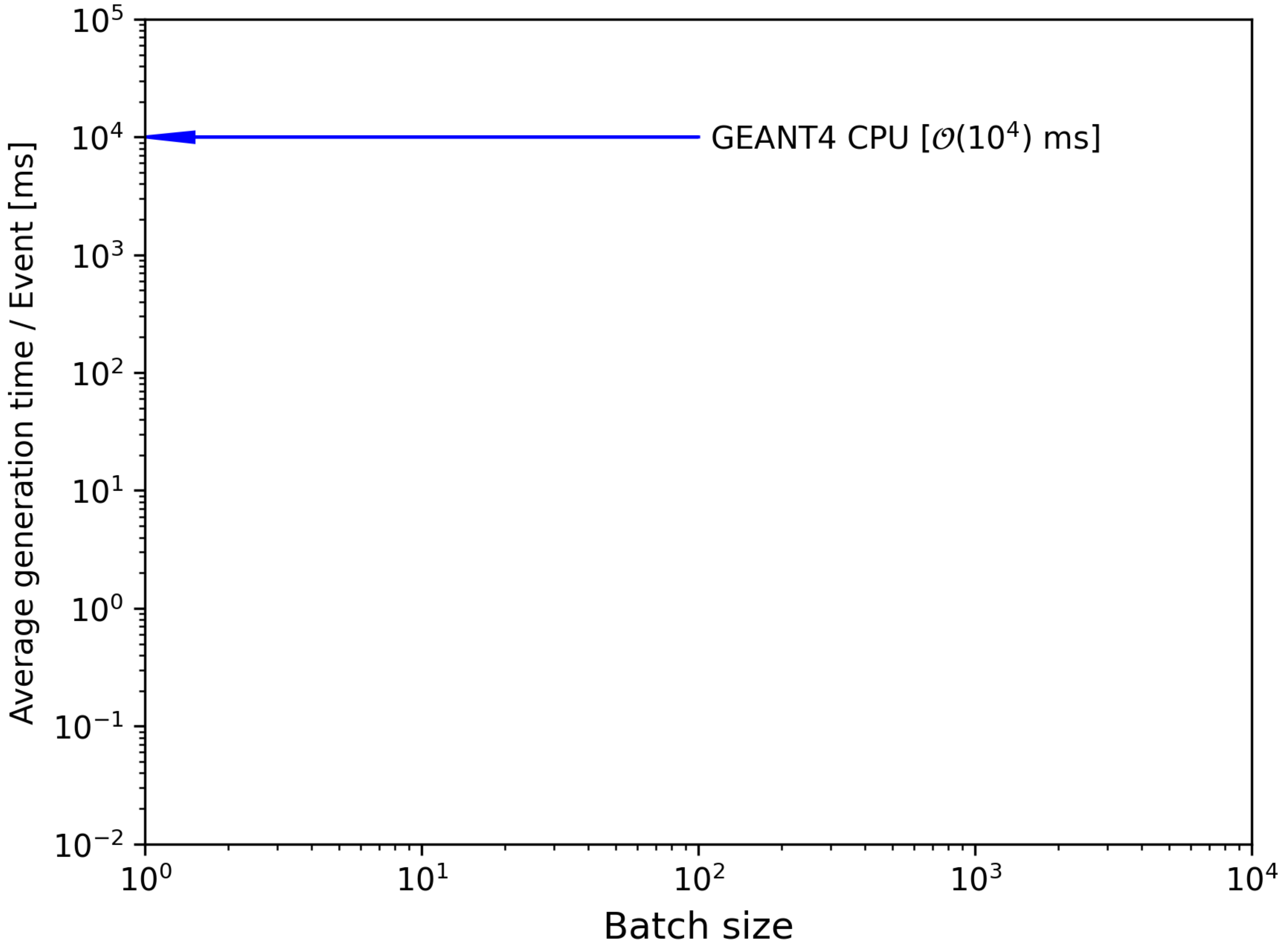


γ shower shape

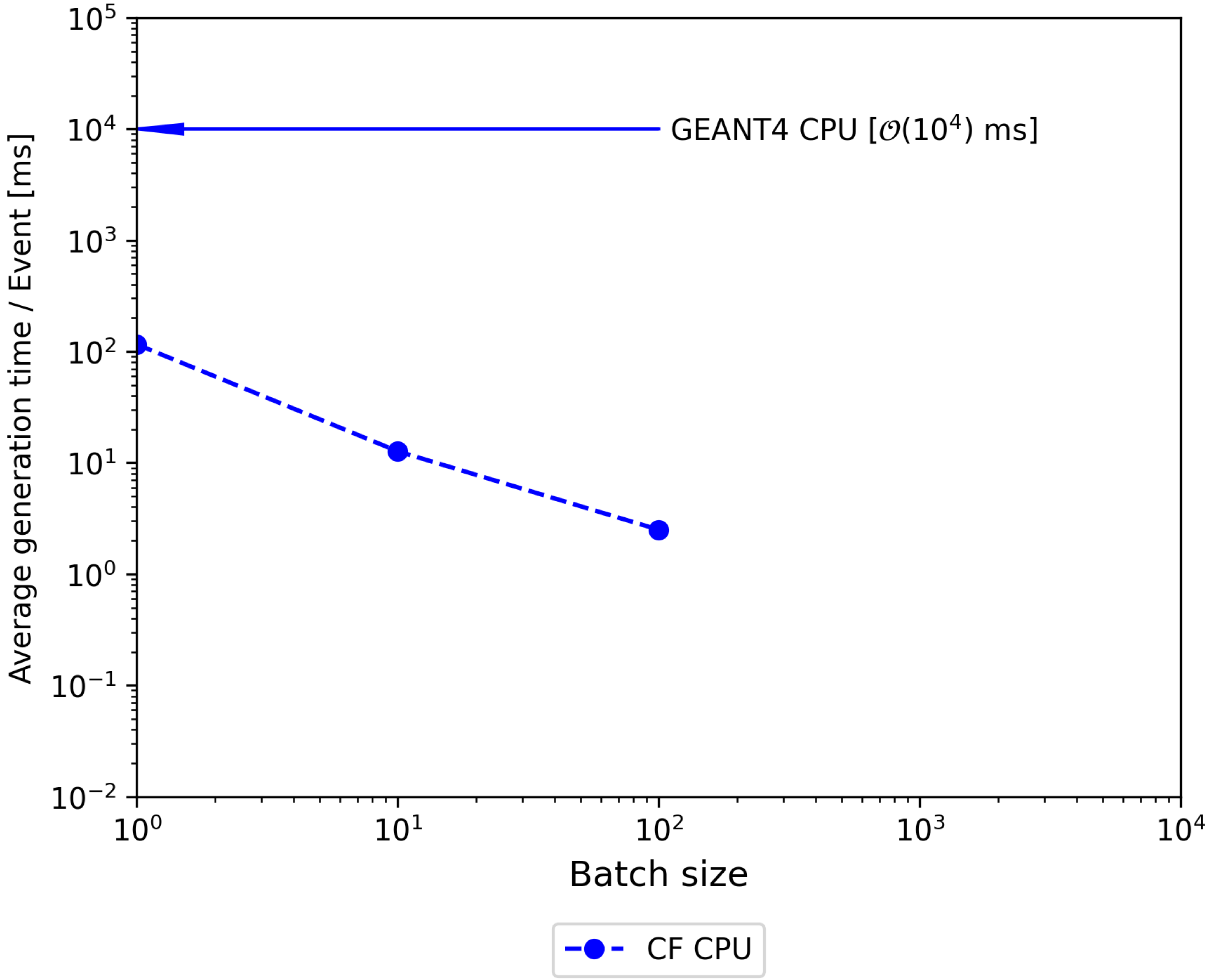


Teacher and student overlapped
$\Longrightarrow$ Flow 2 well-trained

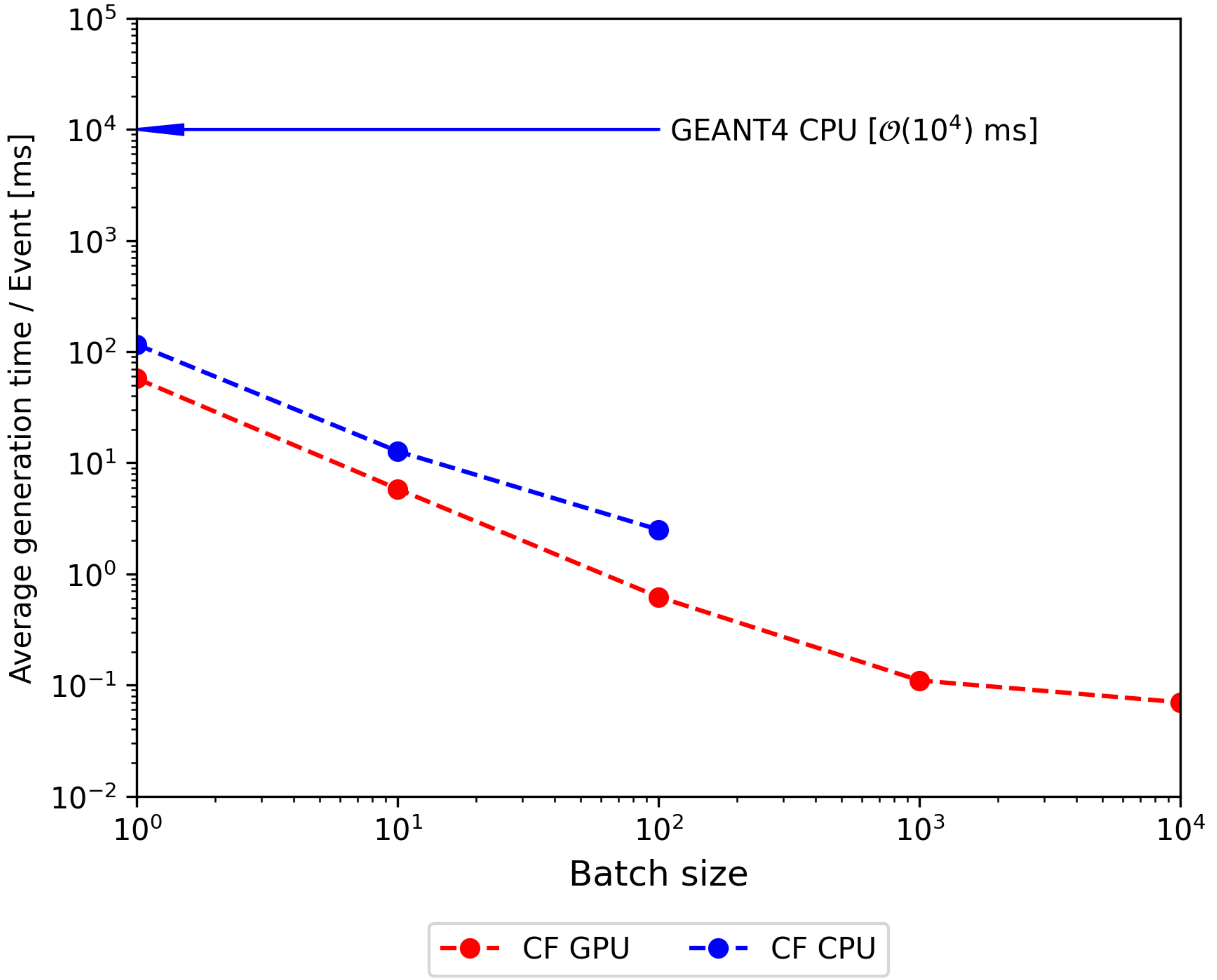
Generation timing



Generation timing



Generation timing

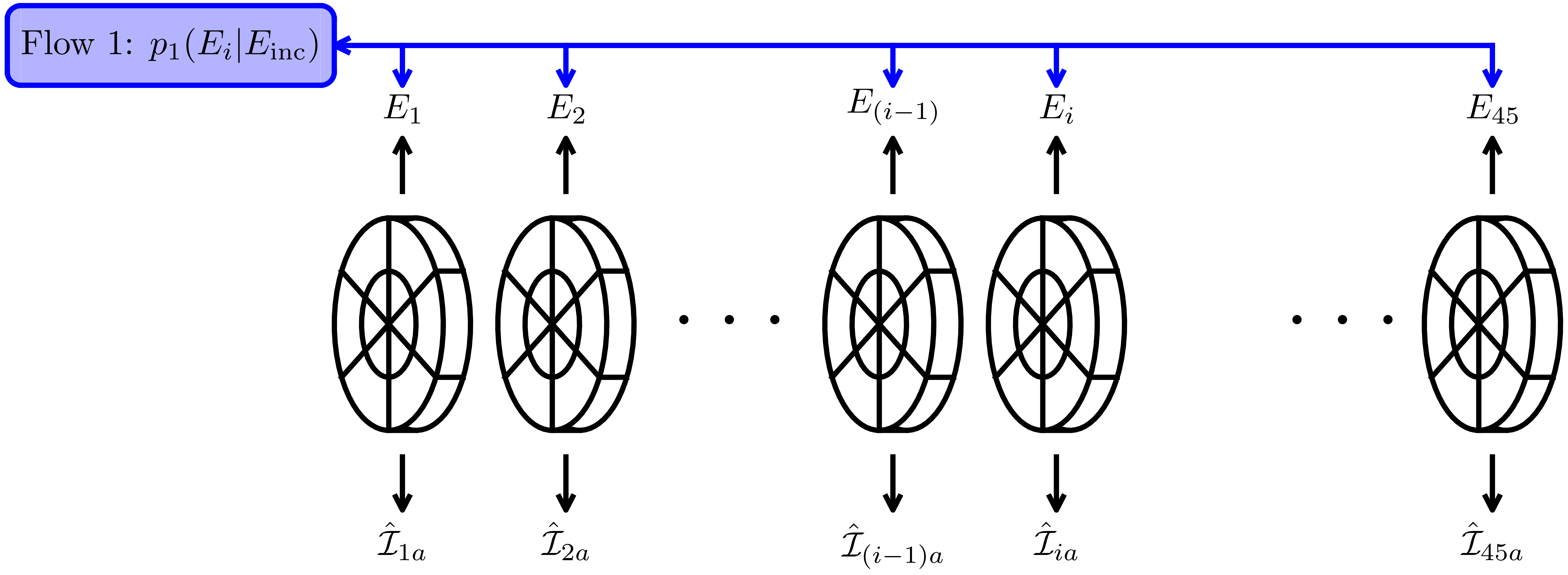


iCaloFlow (Datasets 2 & 3)

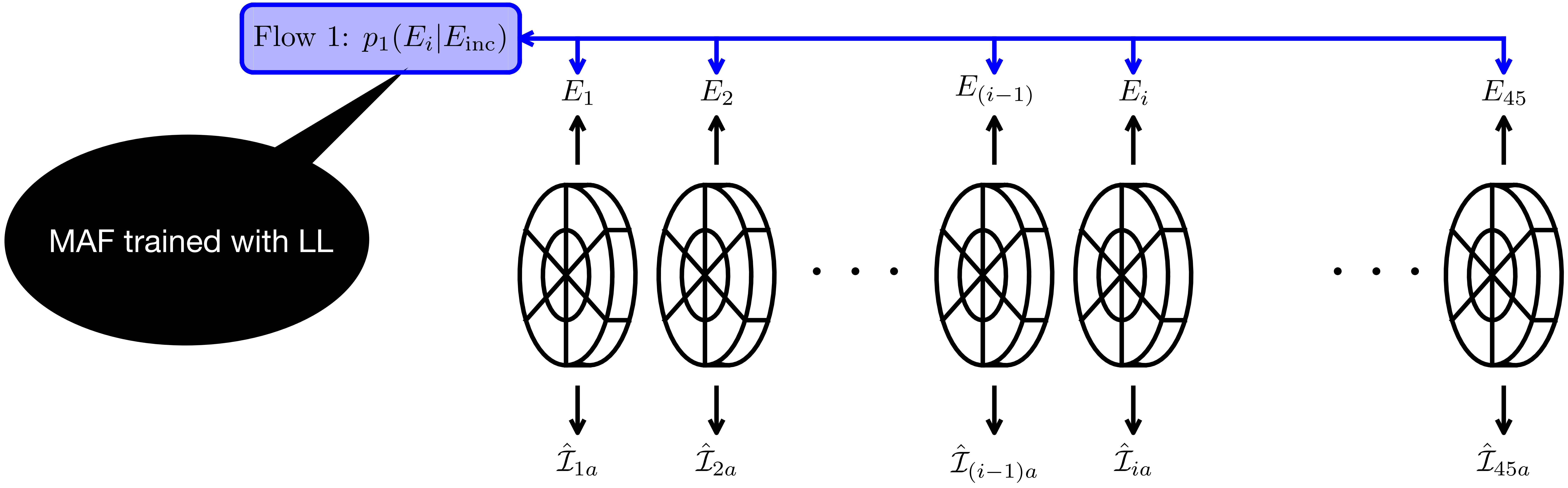
- Physical intuition
  - Particles propagate through calorimeter primarily in one direction
  - Pattern of energy deposition in given layer should be largely dependent on that in the previous layer

- Main idea
  - Have **ONE** flow to learn each layer's voxel energy distribution inductively!
  - Condition on previous layer's voxel energy distribution
  - Note: L2LFlows is an earlier work based on a similar idea Diefenbacher et al. [2302.11594]
    - Used one flow per layer
    - Conditioned on 5 previous layers

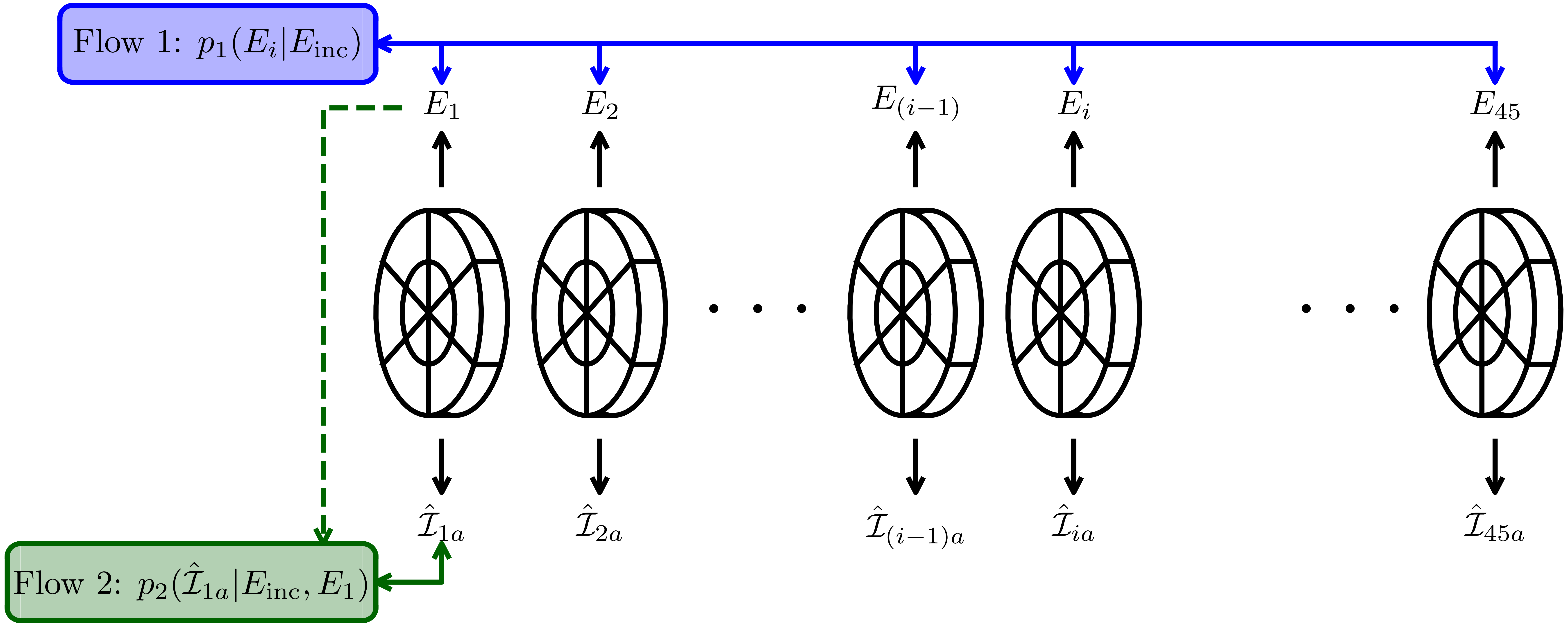
iCaloFlow (Datasets 2 & 3)



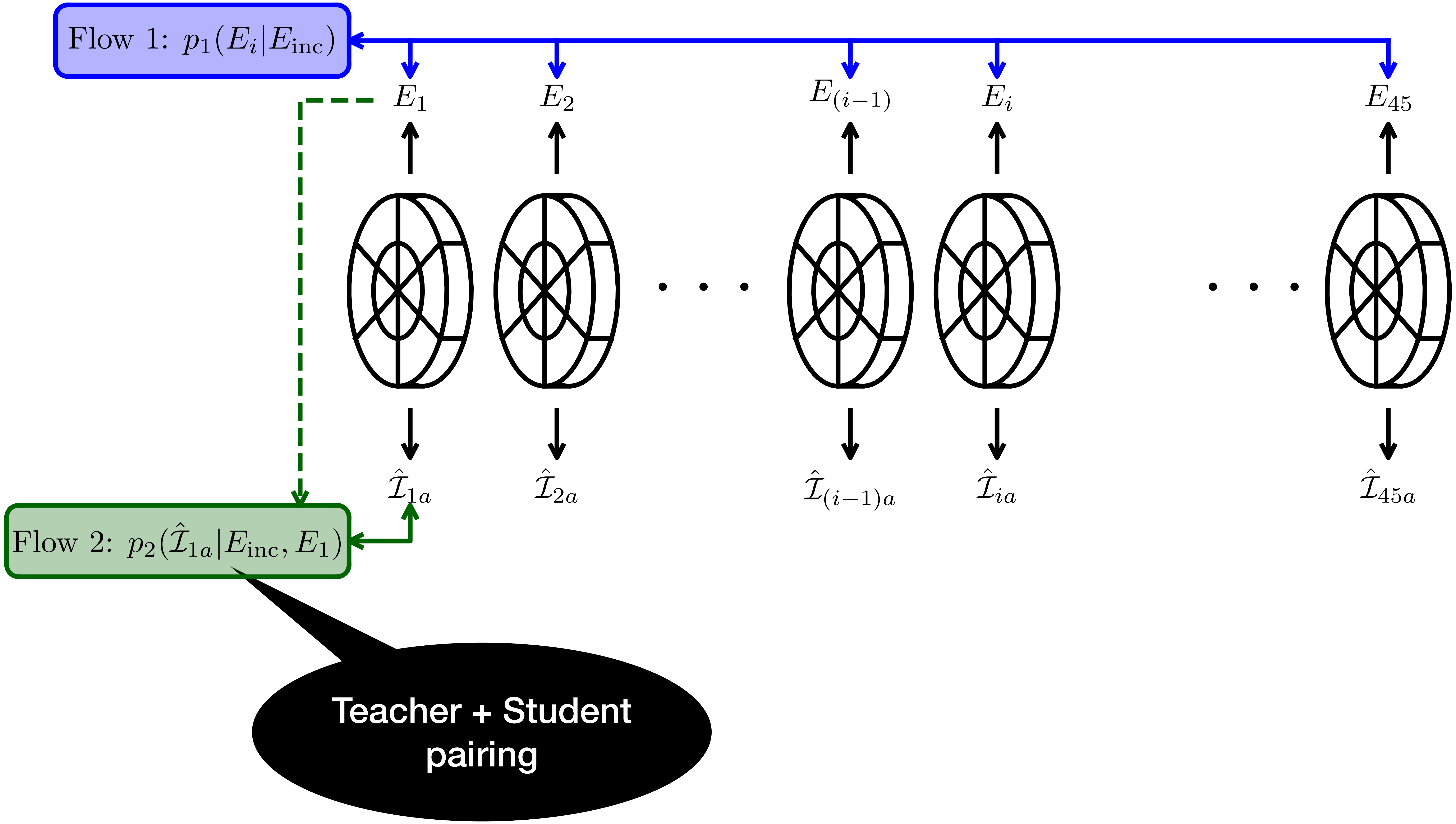
iCaloFlow (Datasets 2 & 3)



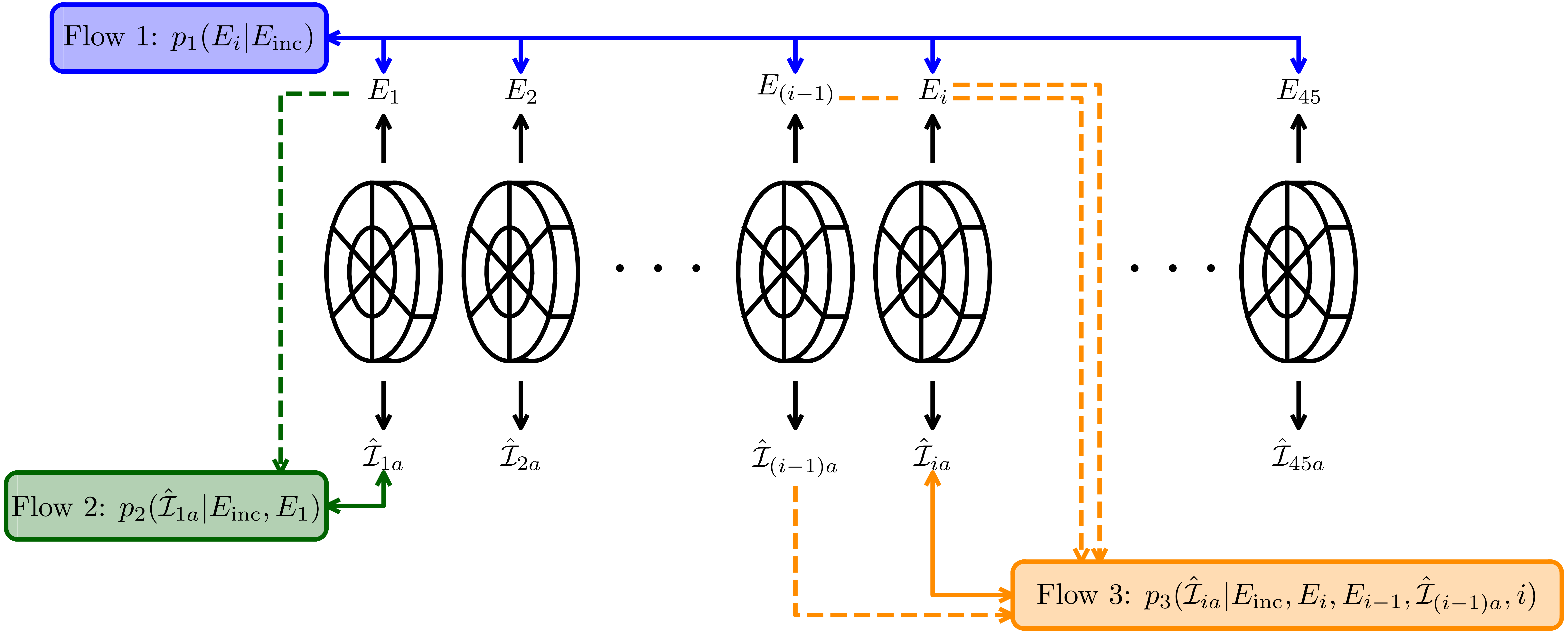
iCaloFlow (Datasets 2 & 3)



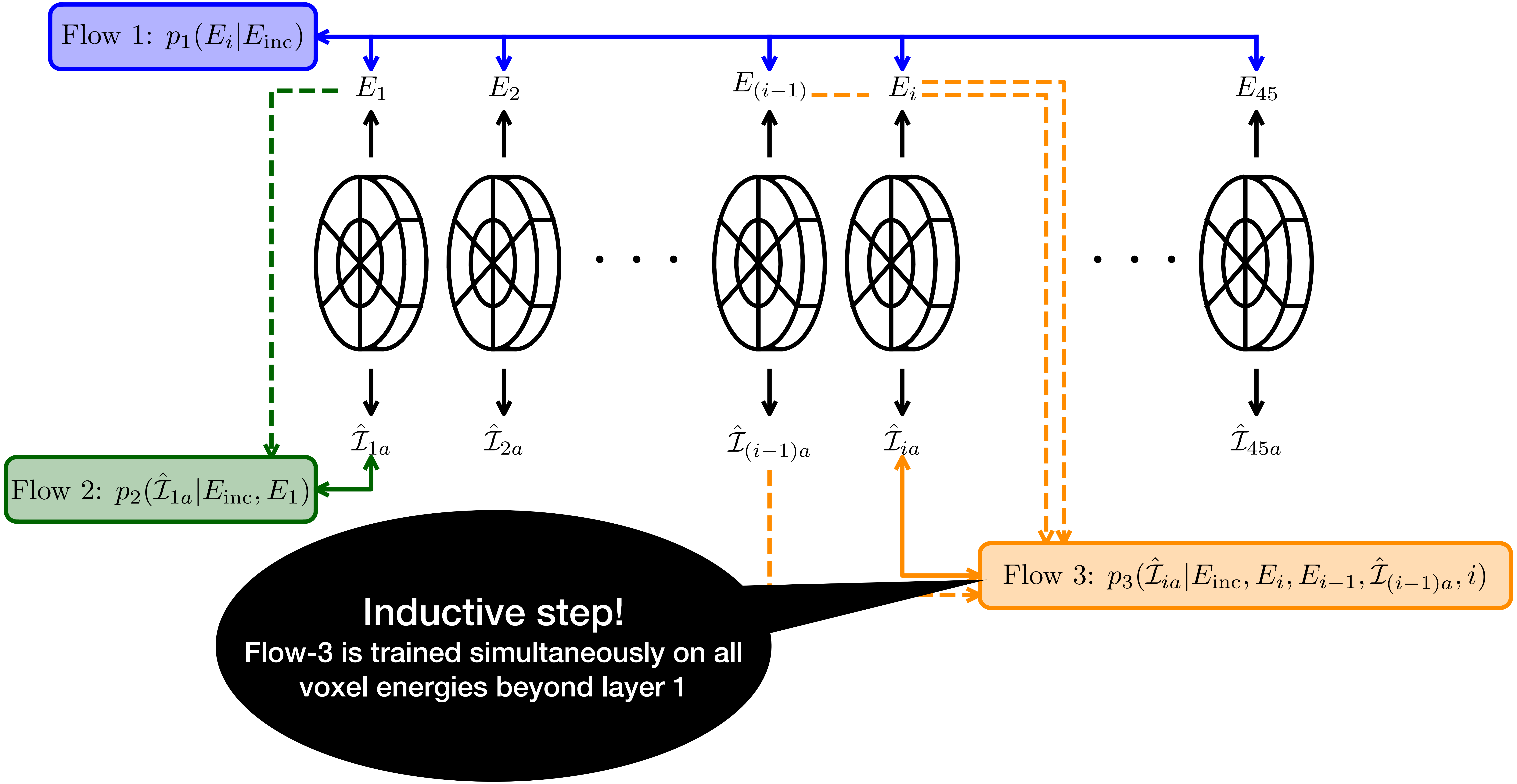
iCaloFlow (Datasets 2 & 3)



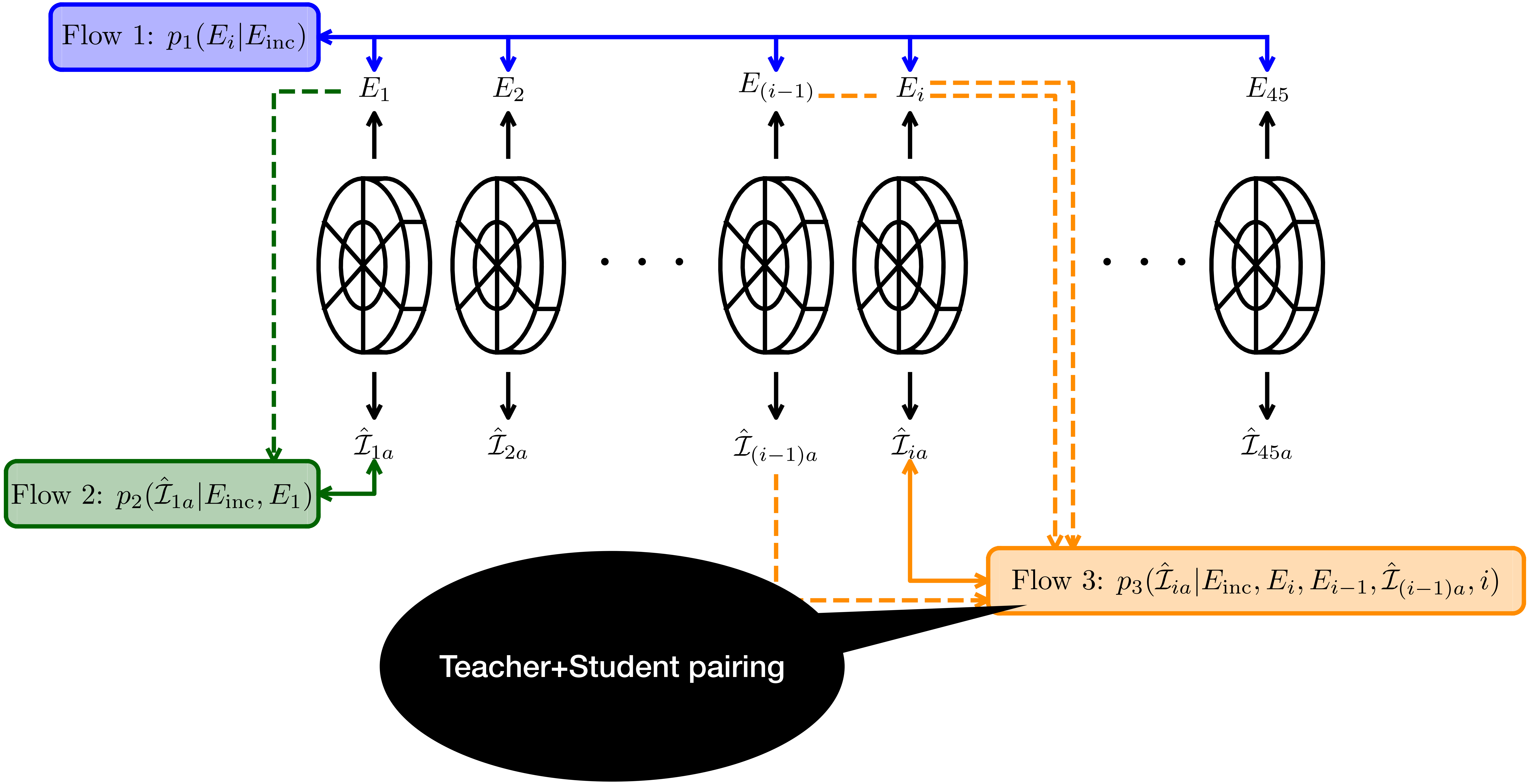
iCaloFlow (Datasets 2 & 3)



iCaloFlow (Datasets 2 & 3)



iCaloFlow (Datasets 2 & 3)



Classifier scores (DS 2 & 3)

	low-level features		high-level features	
	AUC	JSD	AUC	JSD
DS2 teacher	0.797(5)	0.210(7)	0.798(3)	0.214(5)
DS2 student	0.840(3)	0.286(5)	0.838(2)	0.283(4)
DS3 teacher	0.911(3)	0.465(6)	0.941(1)	0.561(3)
DS3 student	0.910(8)	0.462(18)	0.951(1)	0.601(5)

Classifier scores (DS 2 & 3)

LLF classifier score
better than HLF score?!

Perhaps LLF classifier has
insufficient capacity

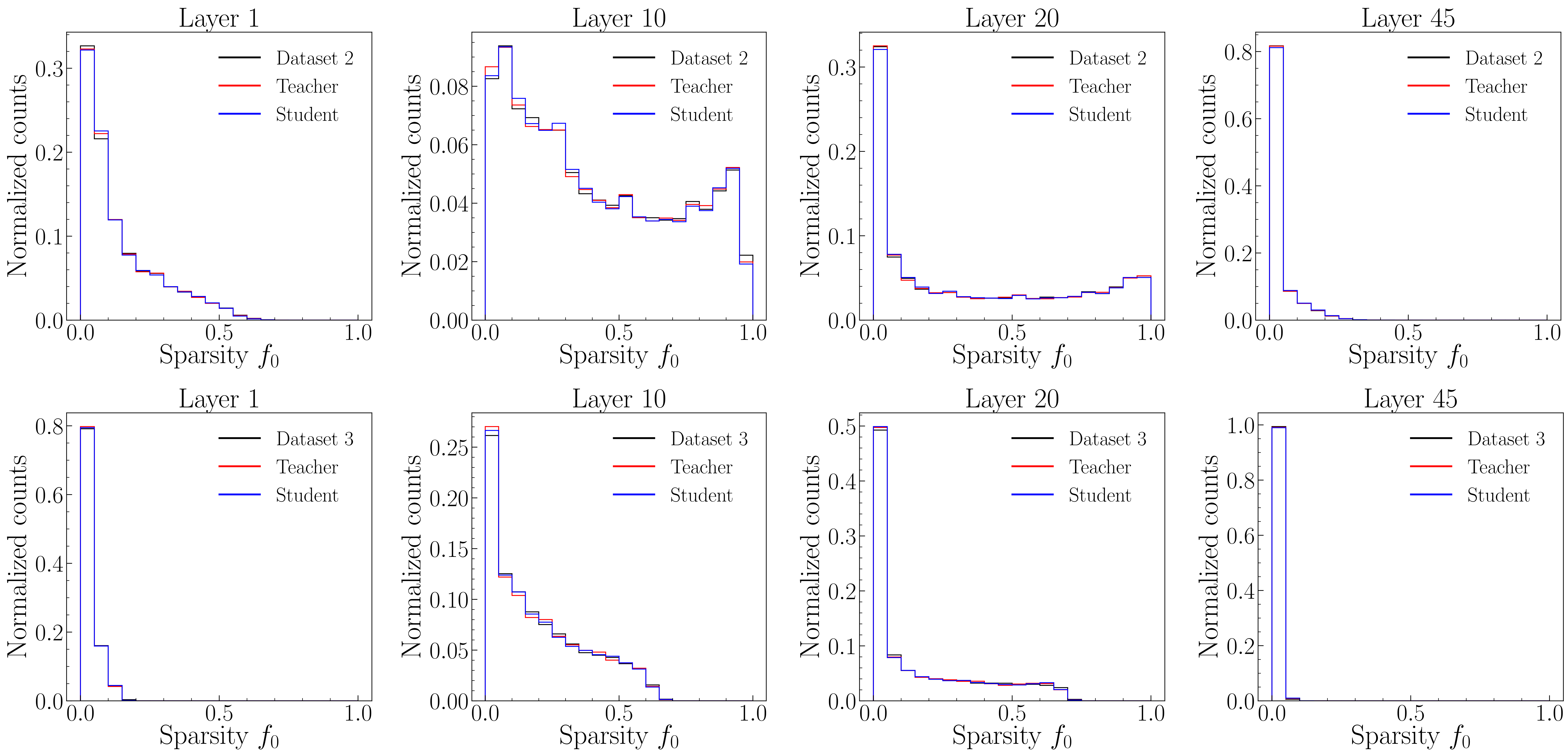
	low-level features		high-level features	
	UC	JSD	AUC	JSD
[illegible]	97(5)	0.210(7)	0.798(3)	0.214(5)
DS2 student	840(3)	0.286(5)	0.838(2)	0.283(4)
DS3 teacher	0.911(3)	0.465(6)	0.941(1)	0.561(3)
DS3 student	0.910(8)	0.462(18)	0.951(1)	0.601(5)

Sparsity

- Large number of zero voxels in Datasets 2 & 3
  - 75% for Dataset 2
  - 90% for Dataset 3
  - cf. 30% for photons and 60% for pions (Dataset 1)

- Adding [0, 5] keV uniform noise to voxels ensure that flow does not only learn zero voxels

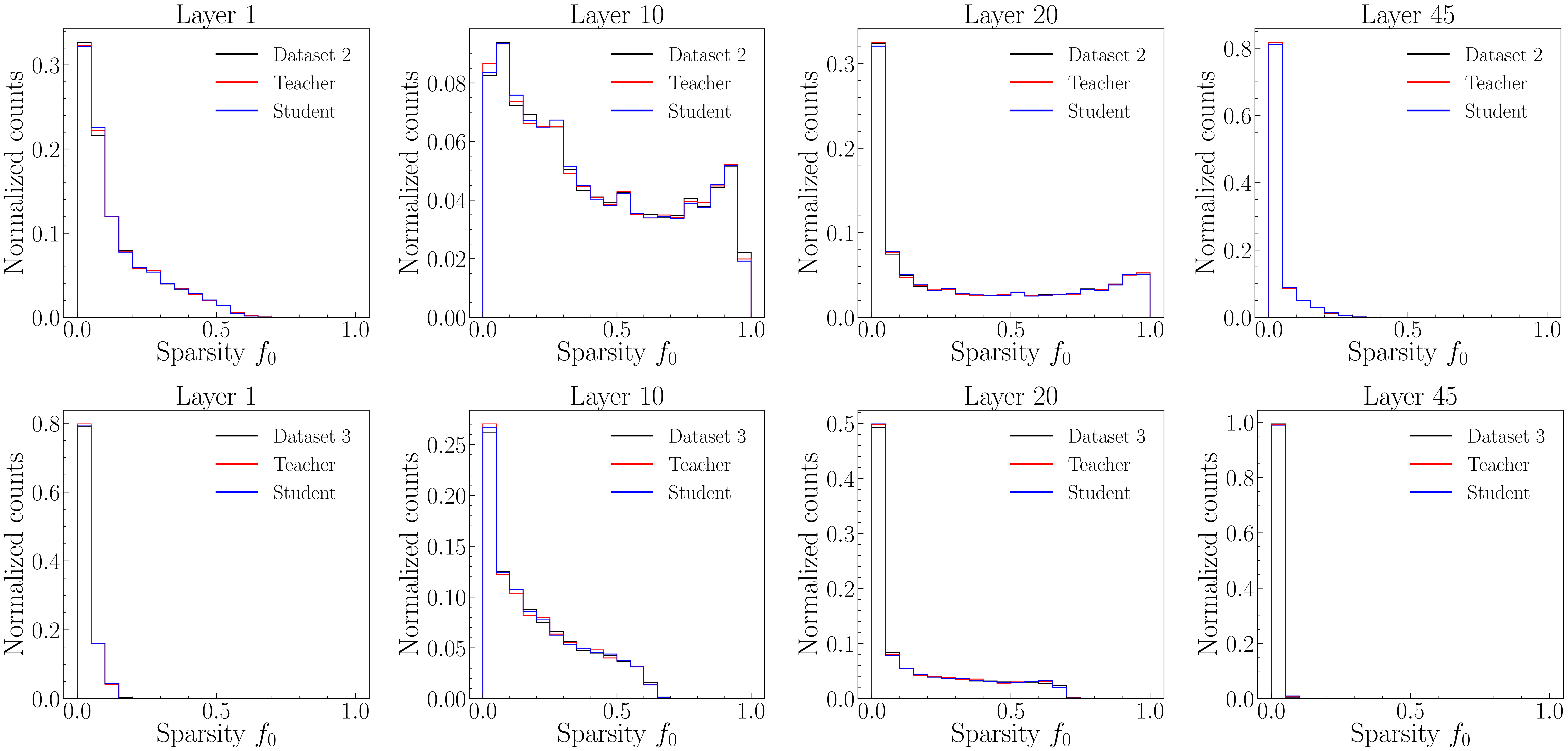
Sparsity



Layer 1
Layer 10
Layer 20
Layer 45
Dataset 2
Dataset 3
Teacher
Student
Normalized counts
Sparsity f_0

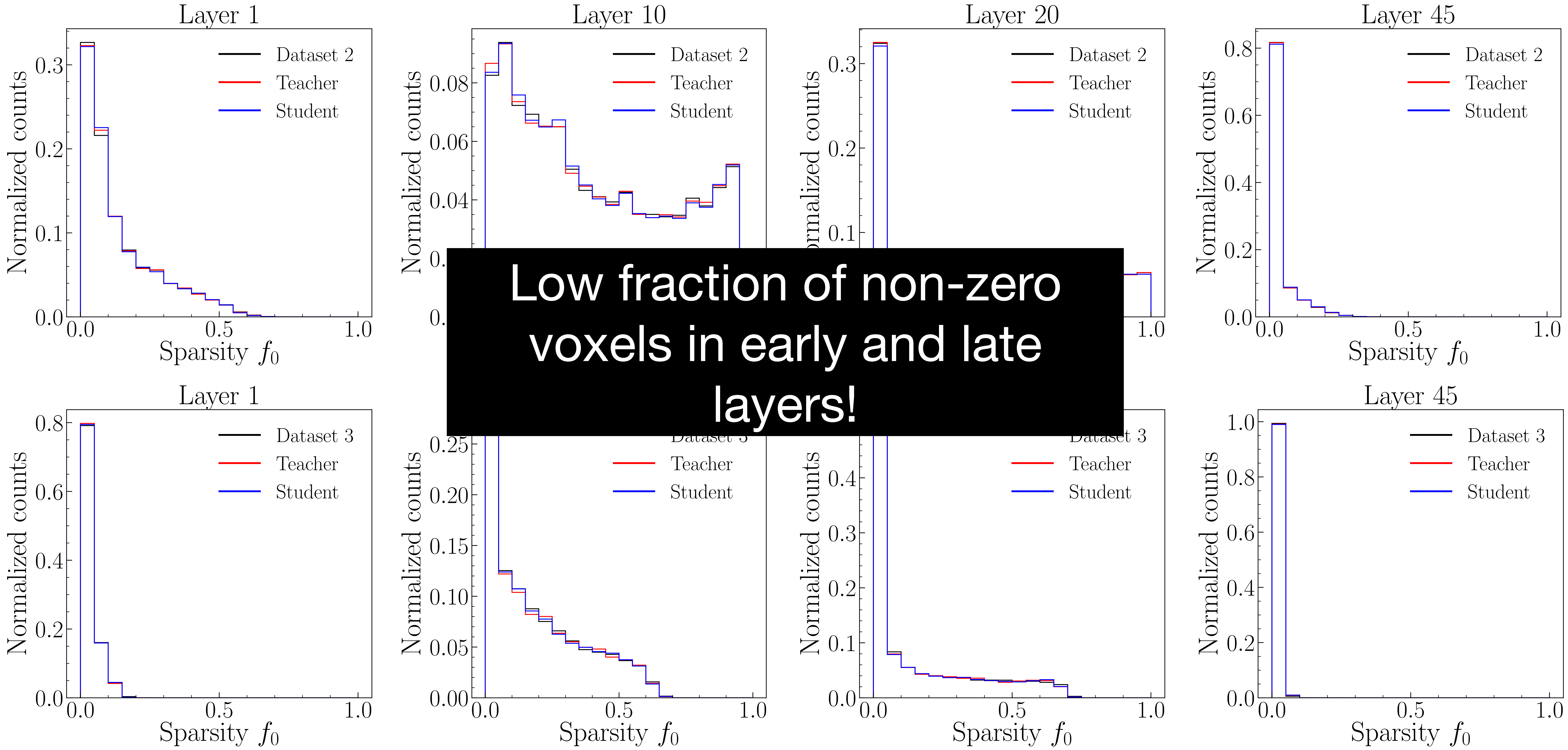
Sparsity

f_0 defined as avg fraction of non-zero voxels in layer



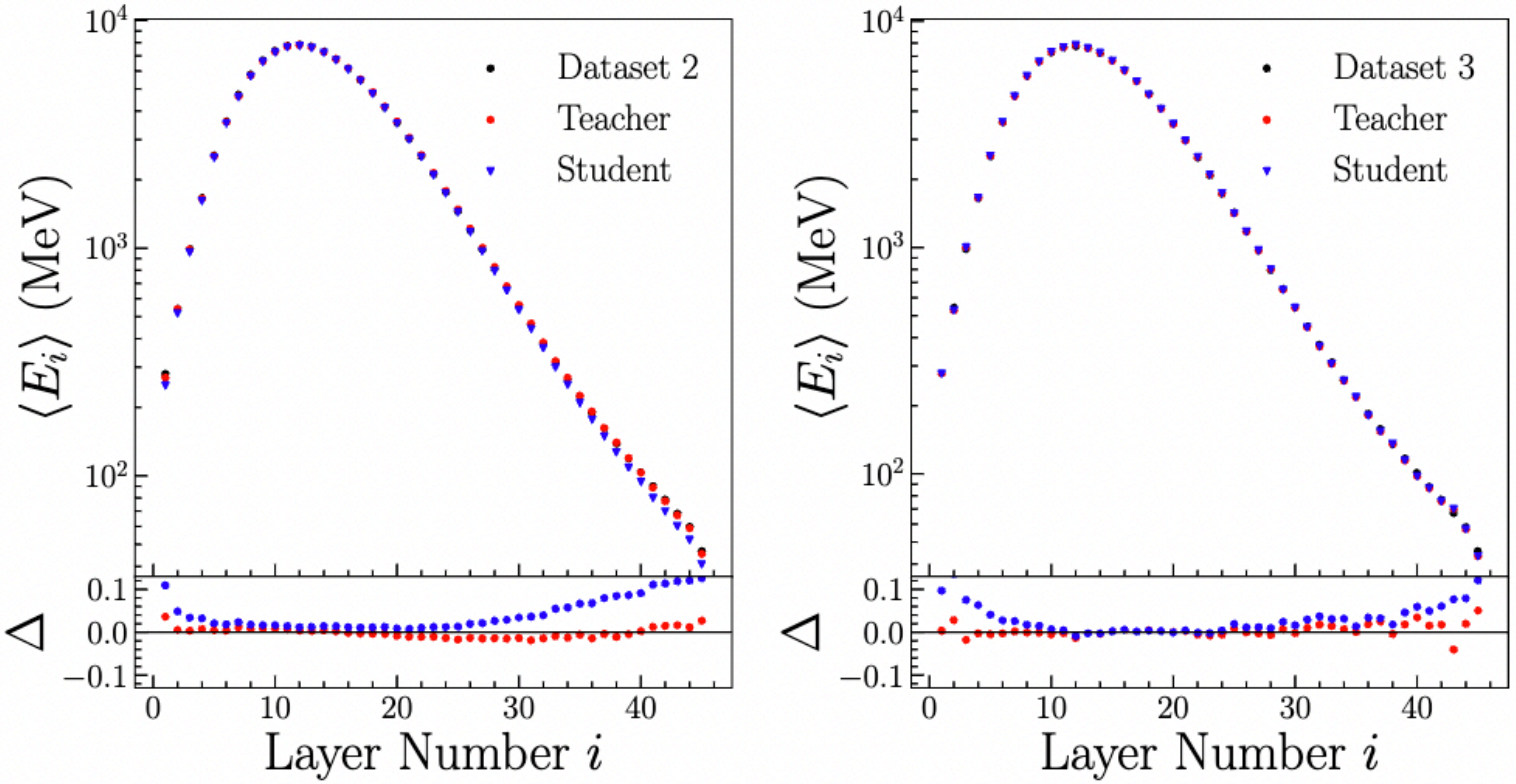
Sparsity

f_0 defined as avg fraction of non-zero voxels in layer

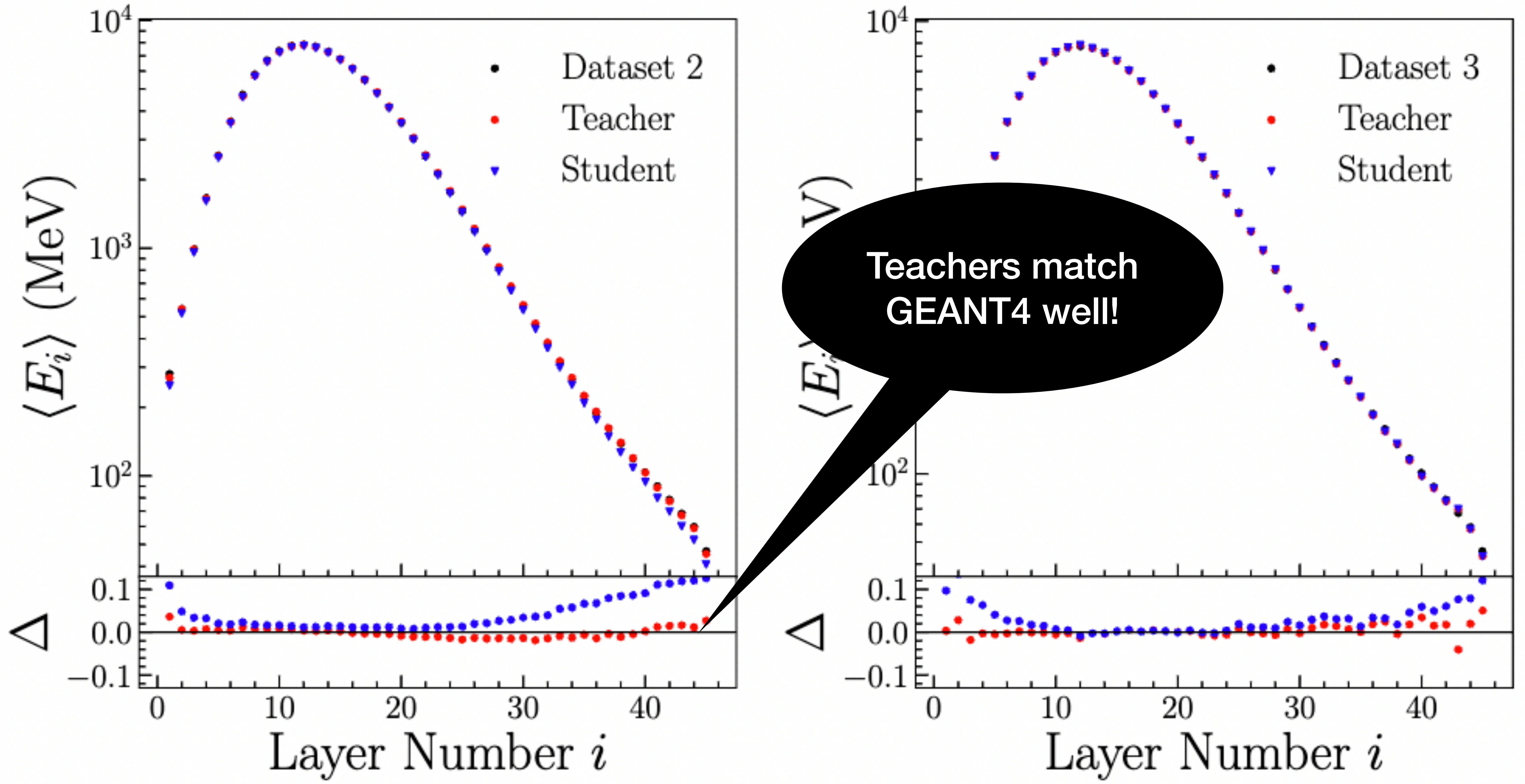


Low fraction of non-zero voxels in early and late layers!

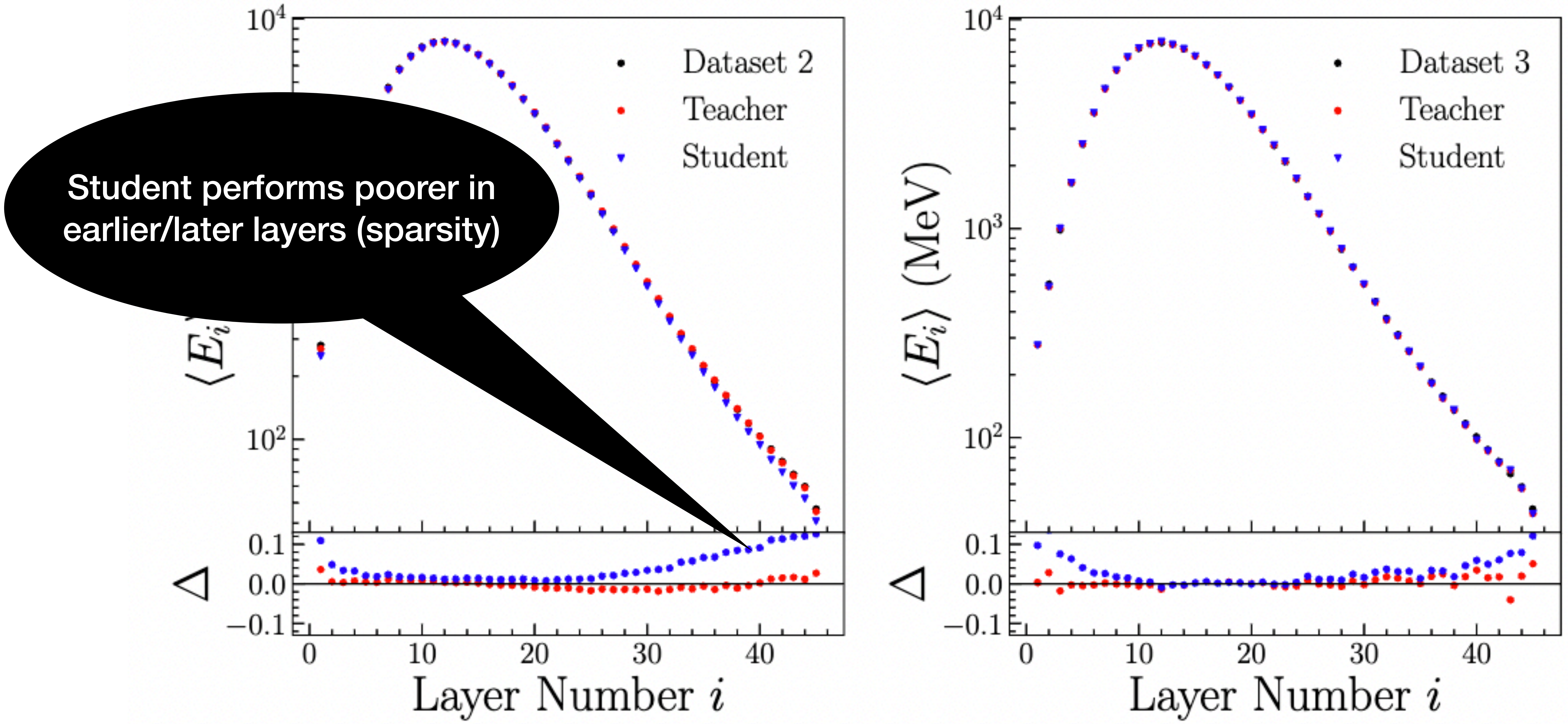
Average layer energy deposition



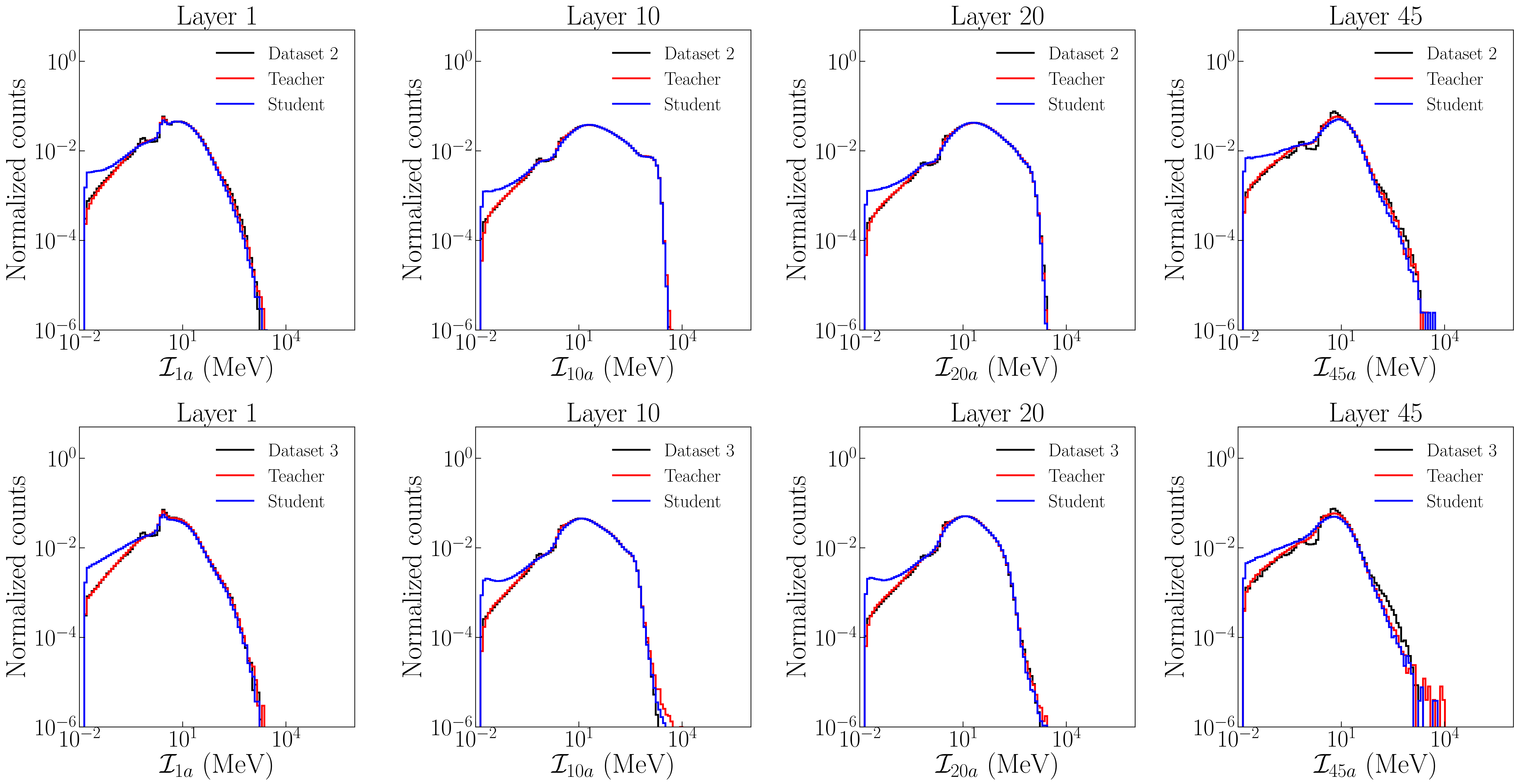
Average layer energy deposition



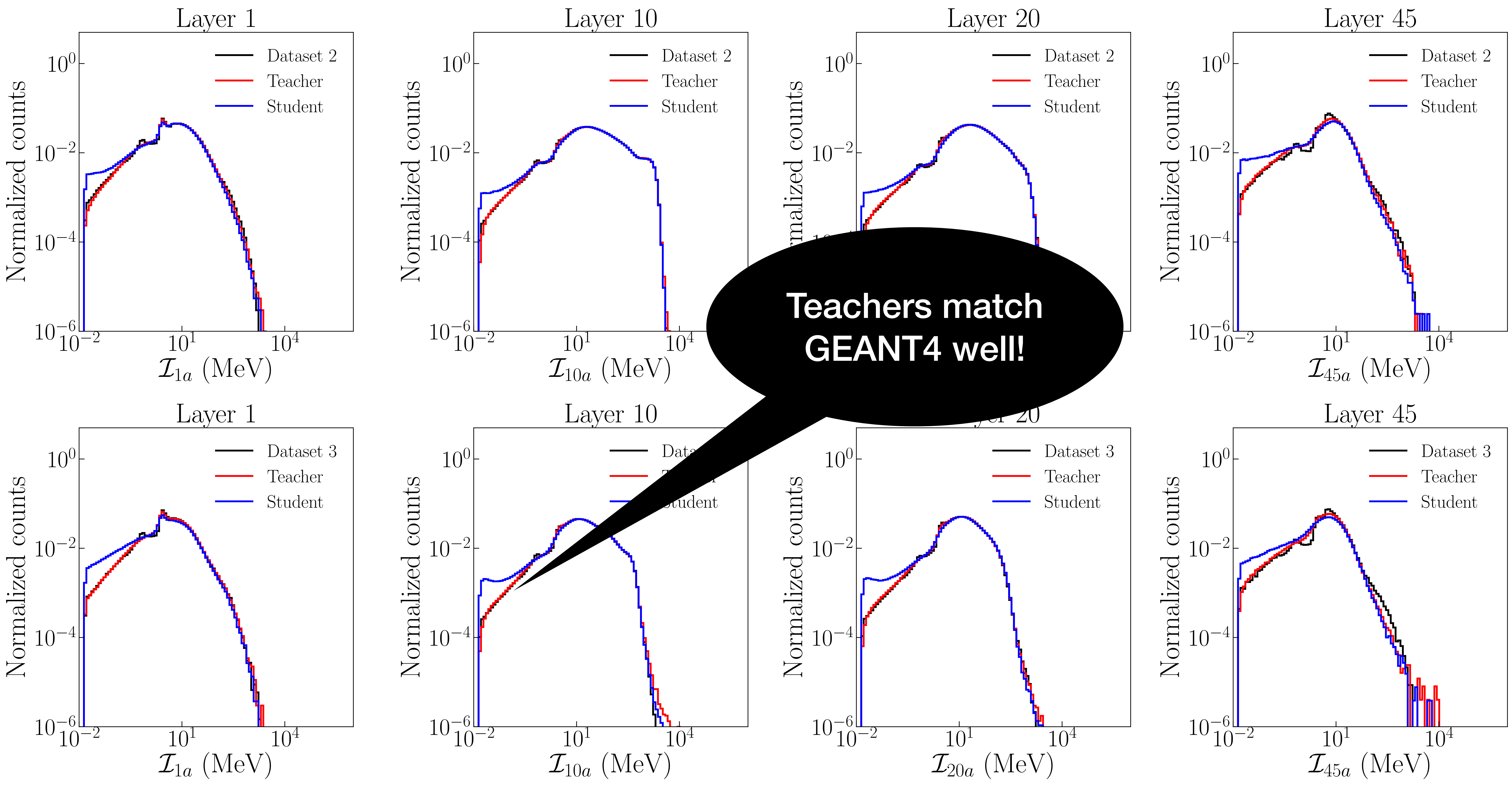
Average layer energy deposition



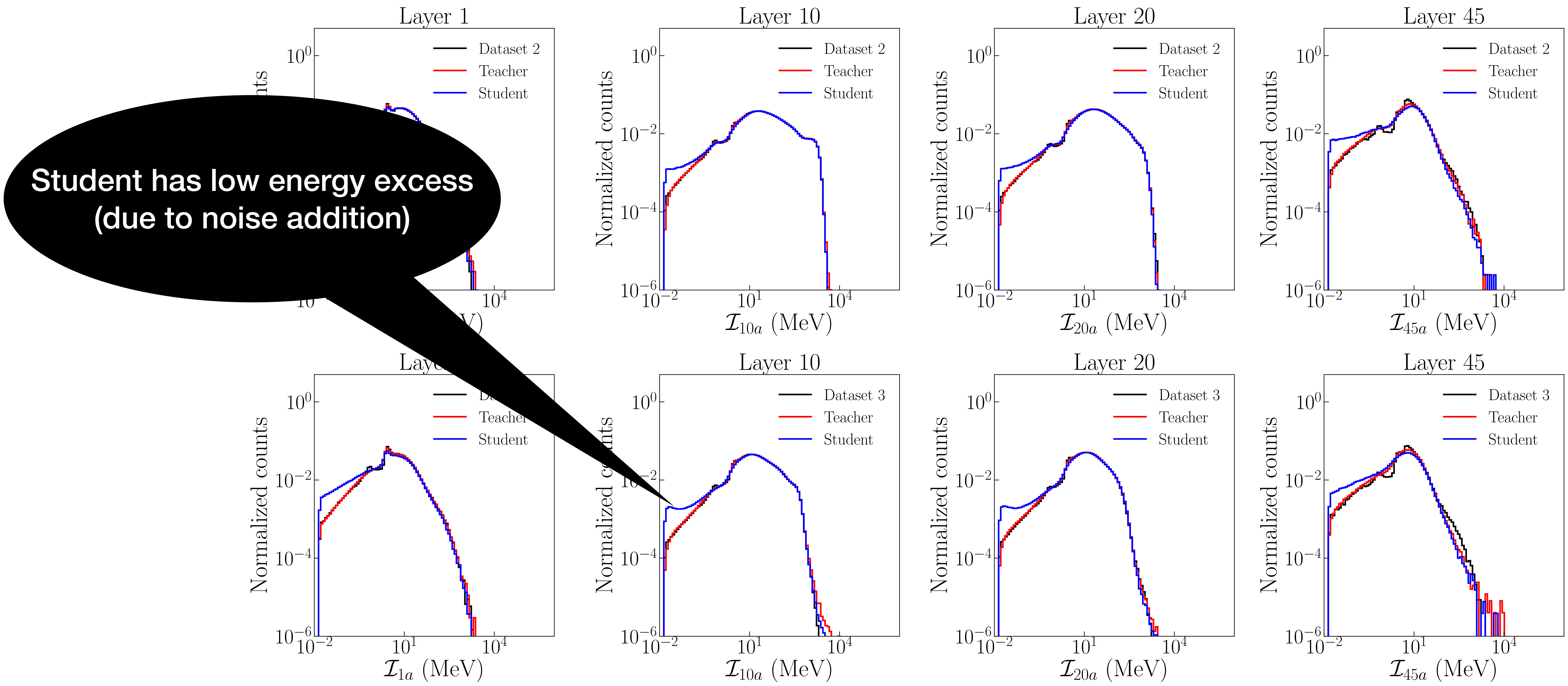
Voxel energies



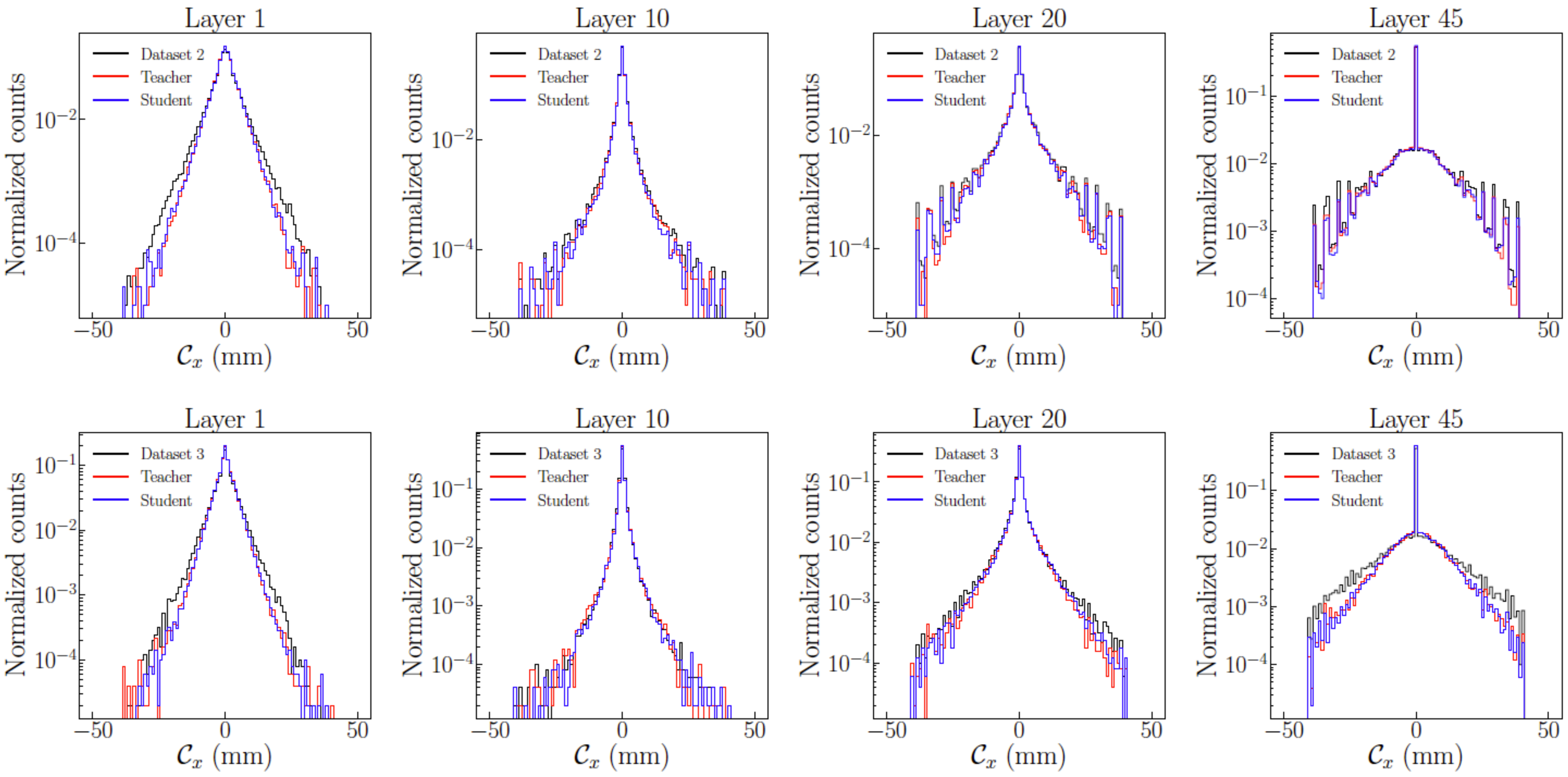
Voxel energies



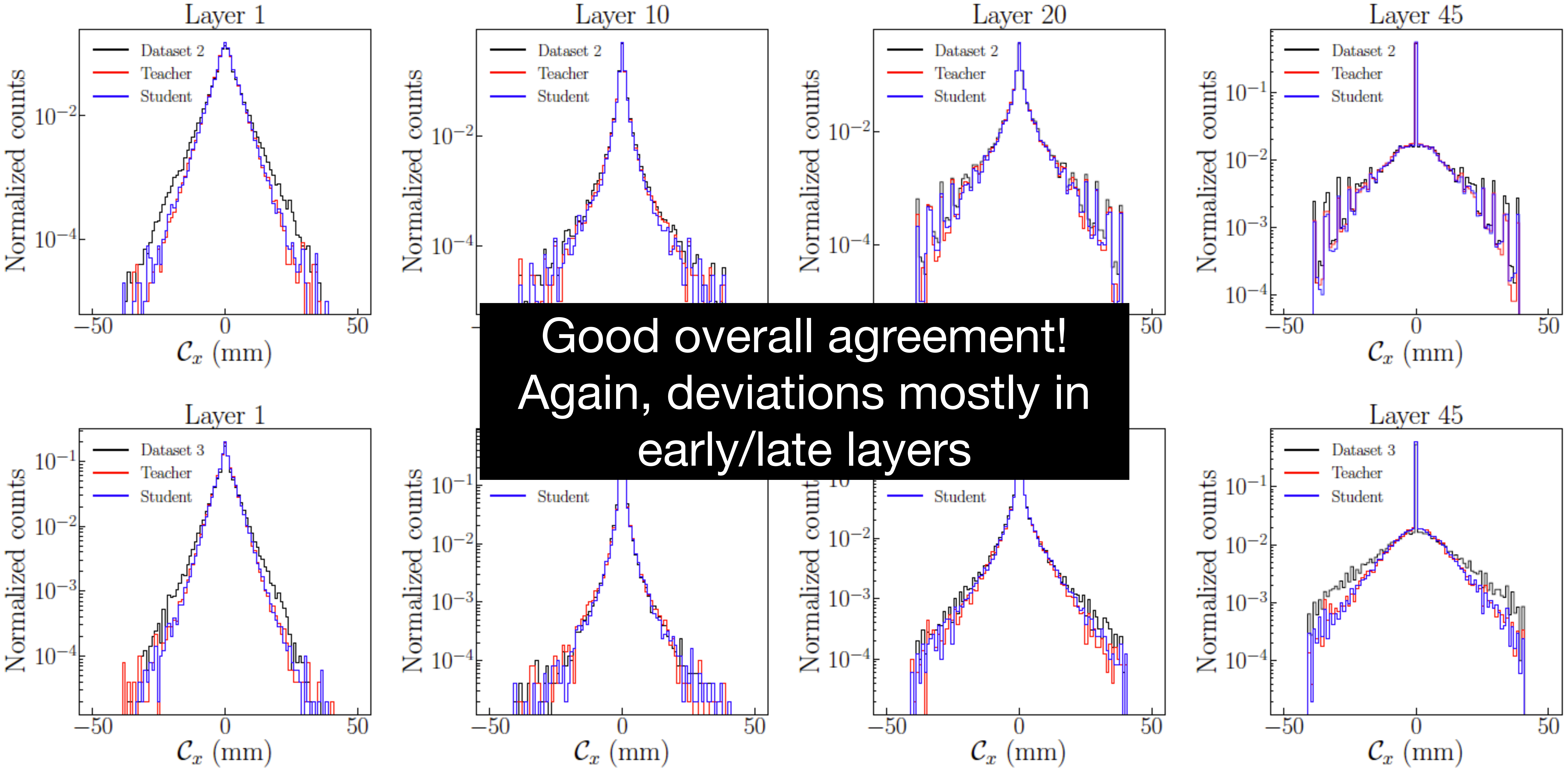
Voxel energies



Shower shape



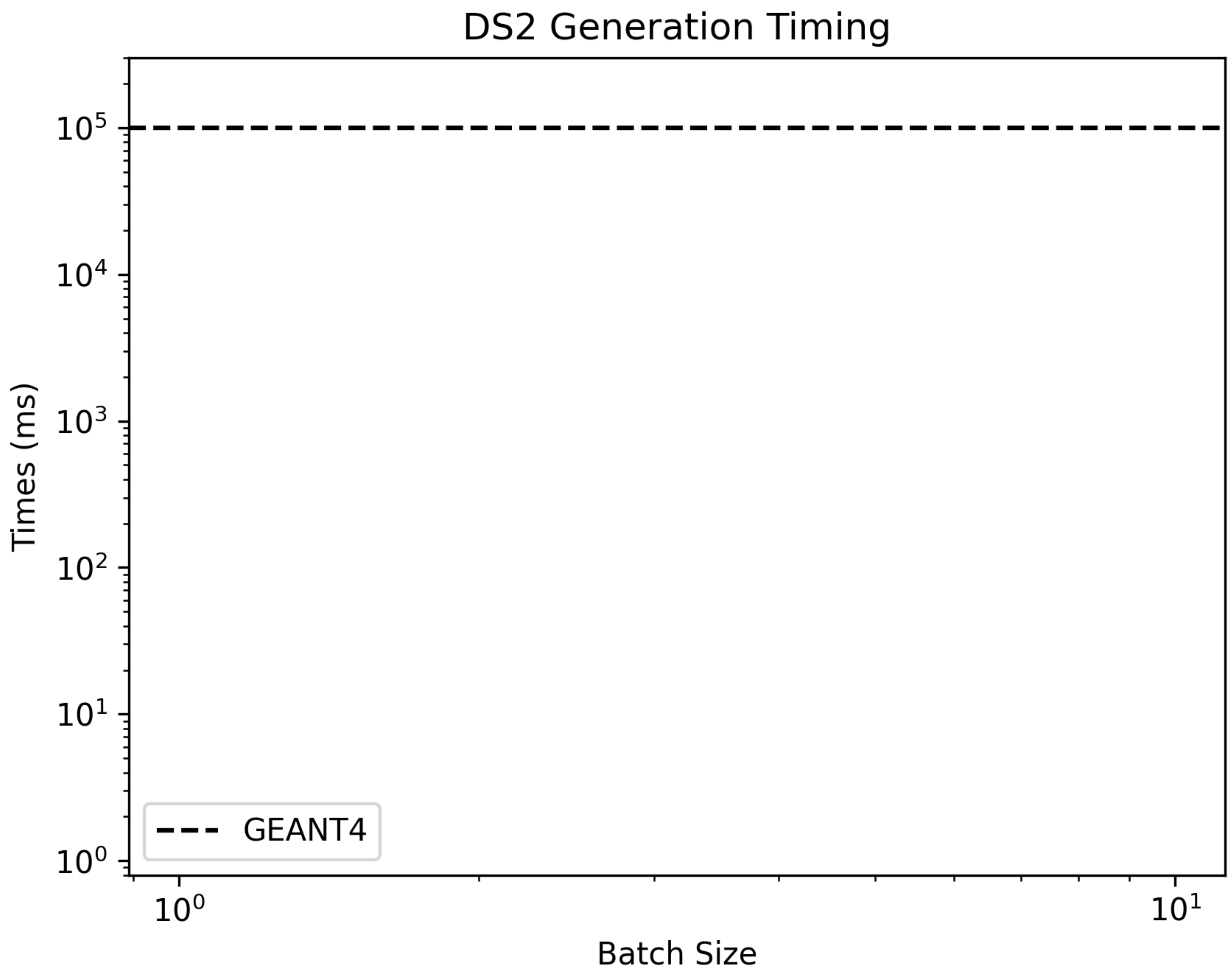
Shower shape



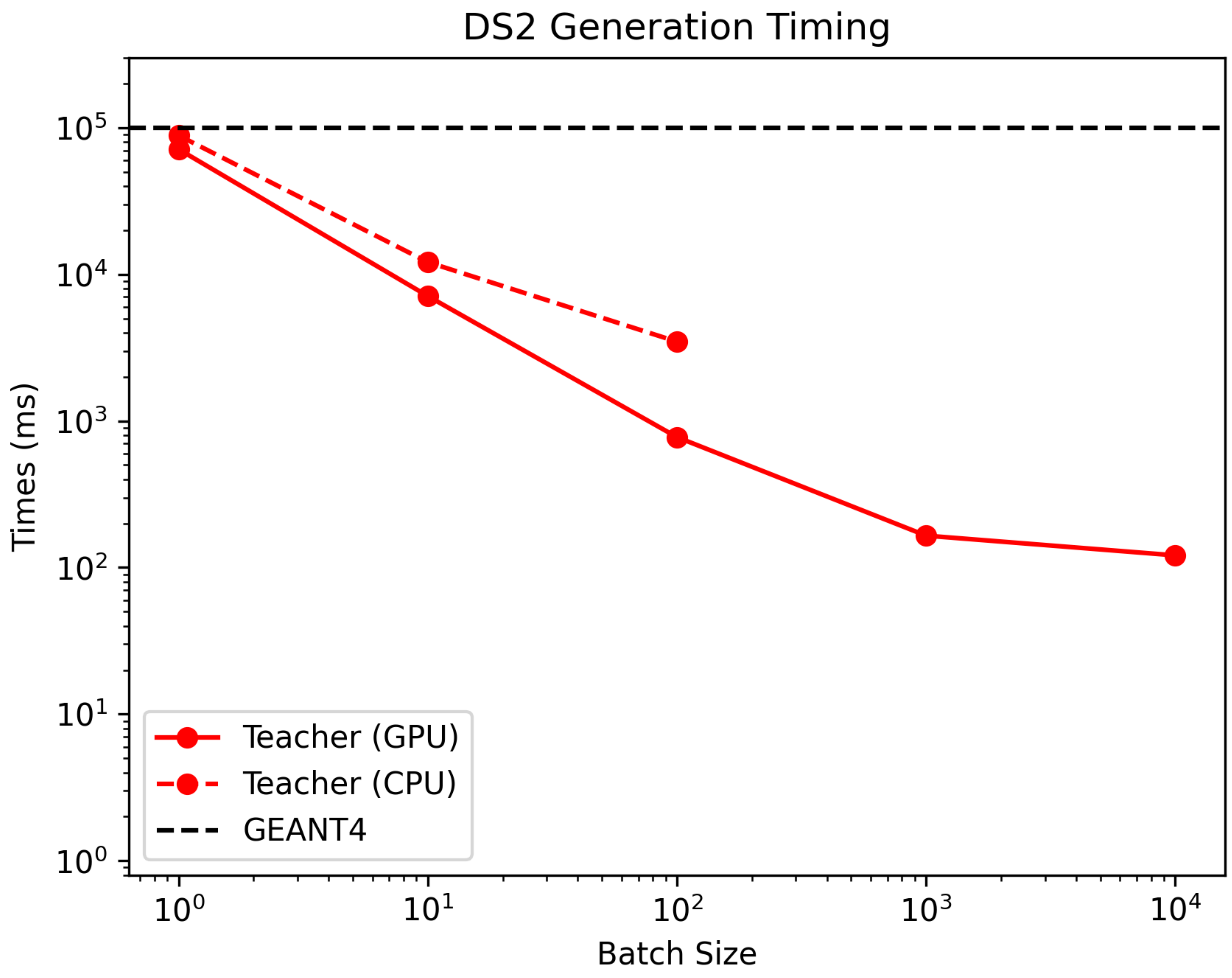
Good overall agreement!
Again, deviations mostly in early/late layers

Generation timing (DS2)

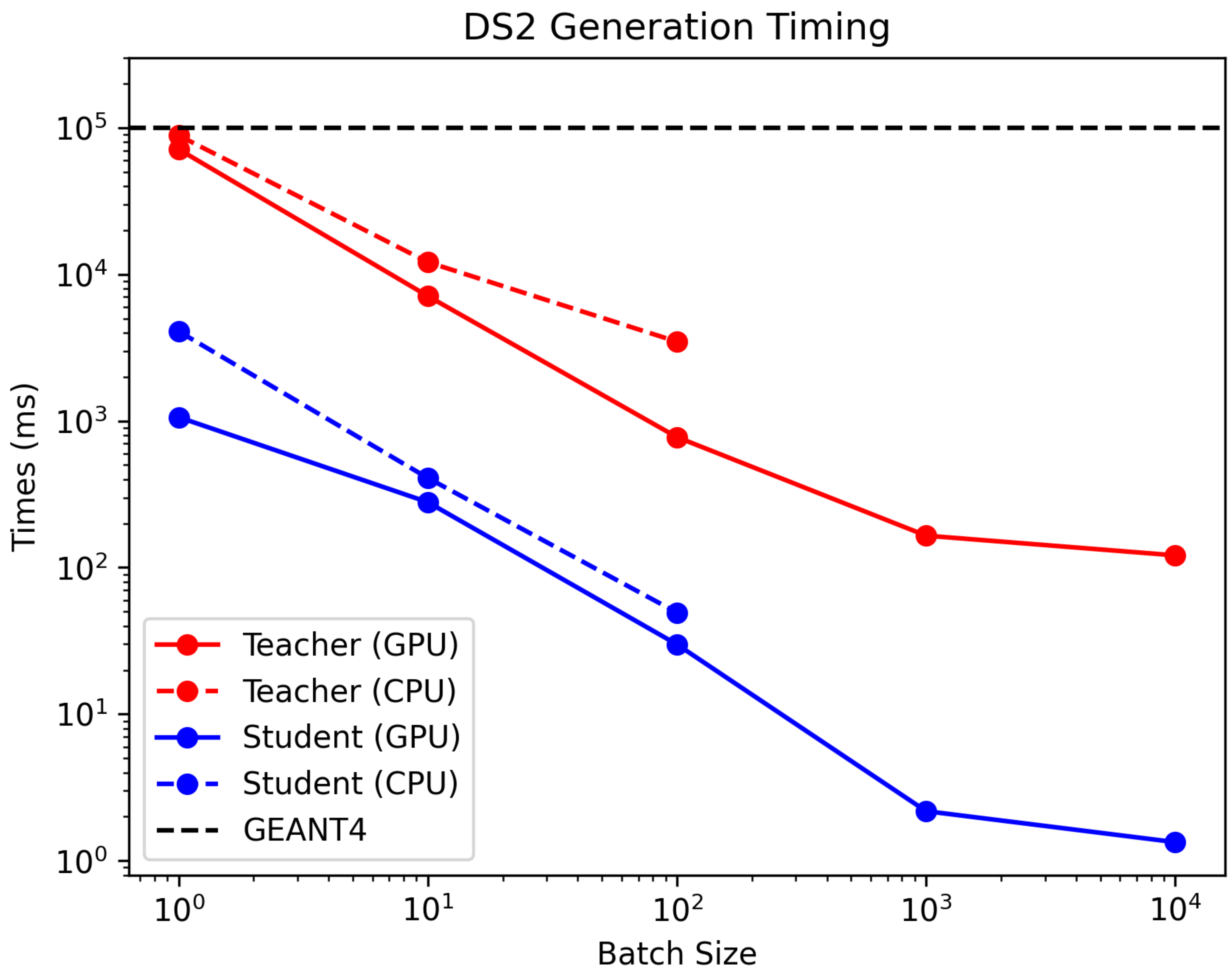
Generation timing (DS2)



Generation timing (DS2)

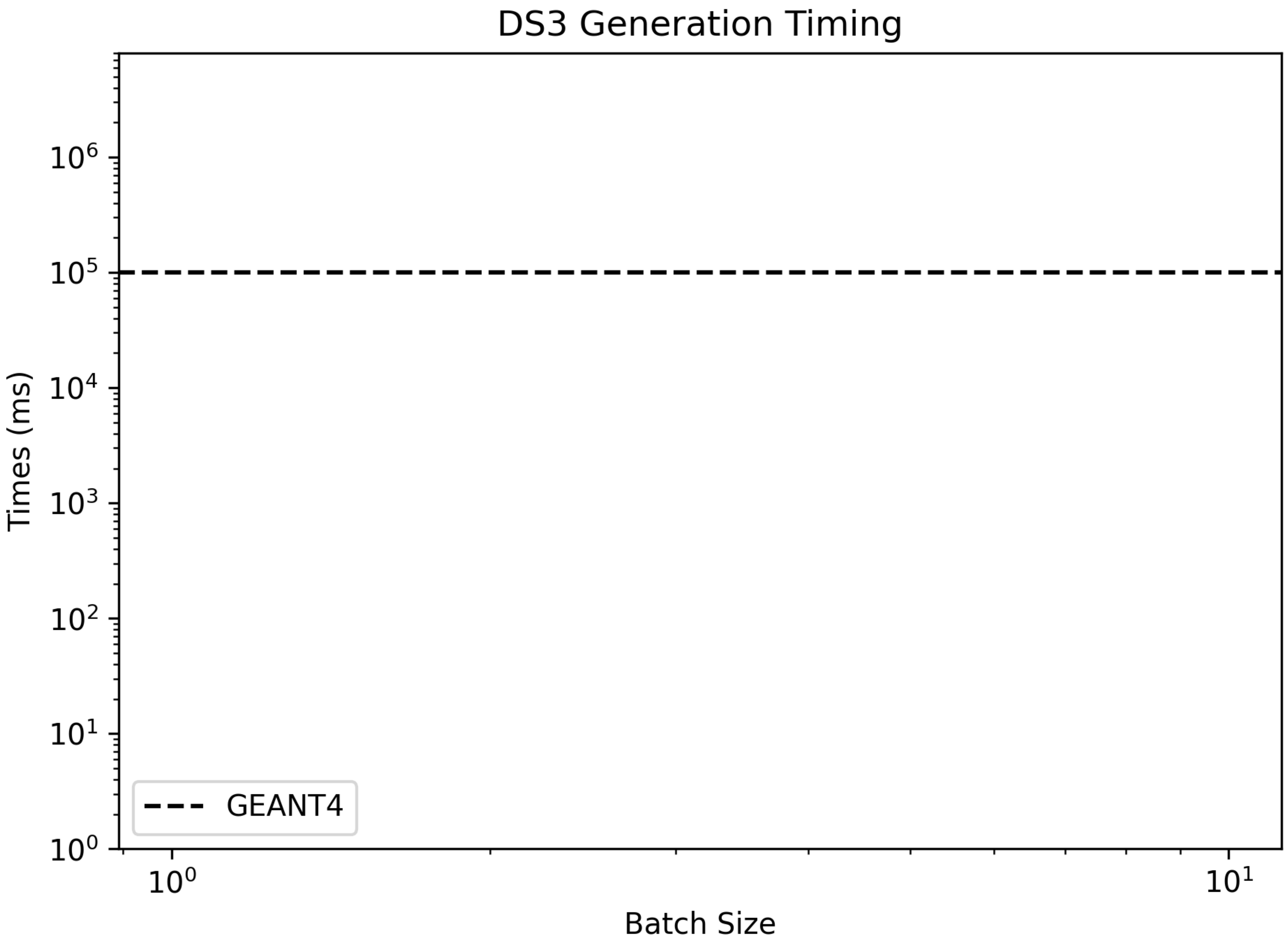


Generation timing (DS2)

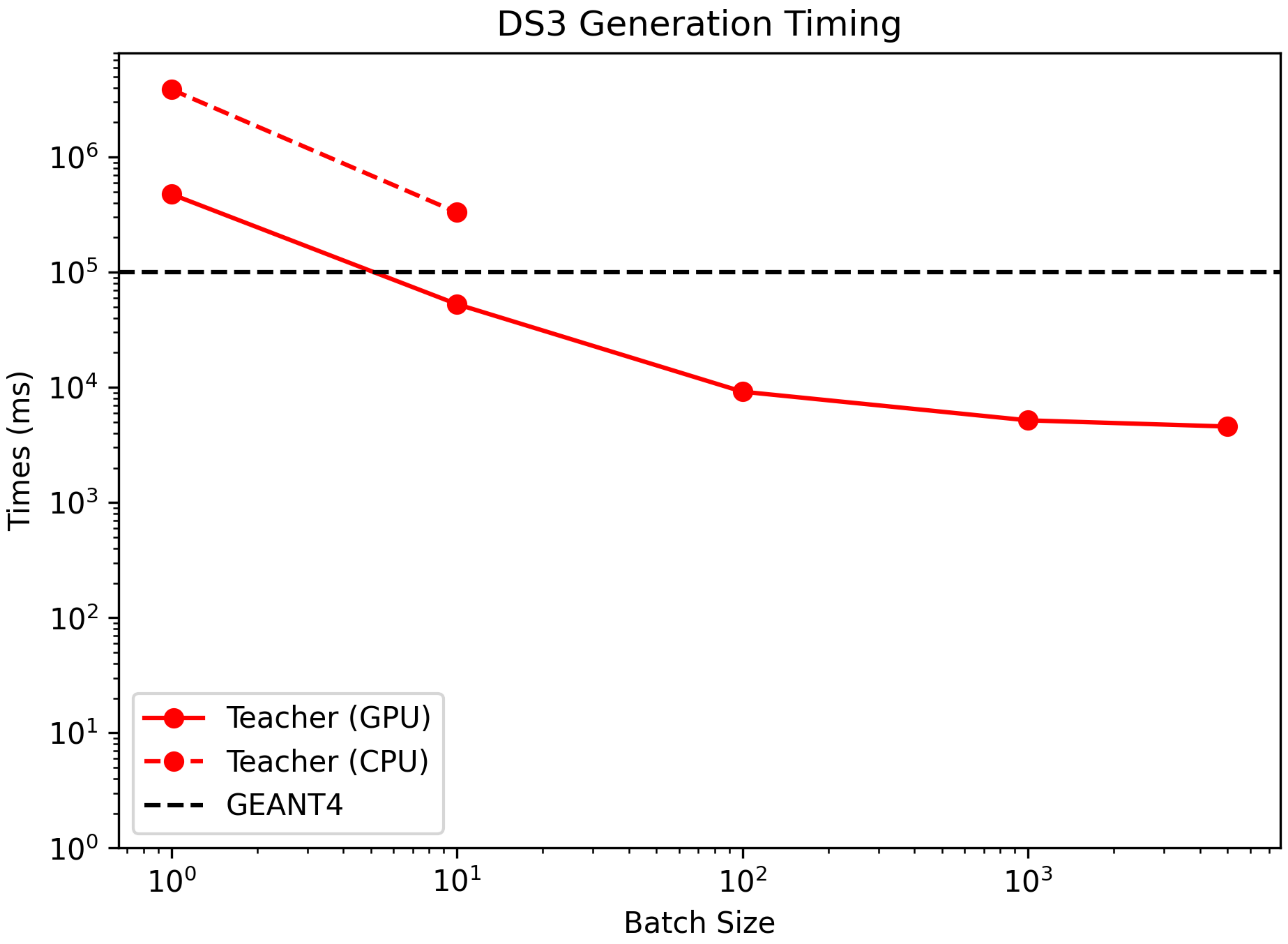


Generation timing (DS3)

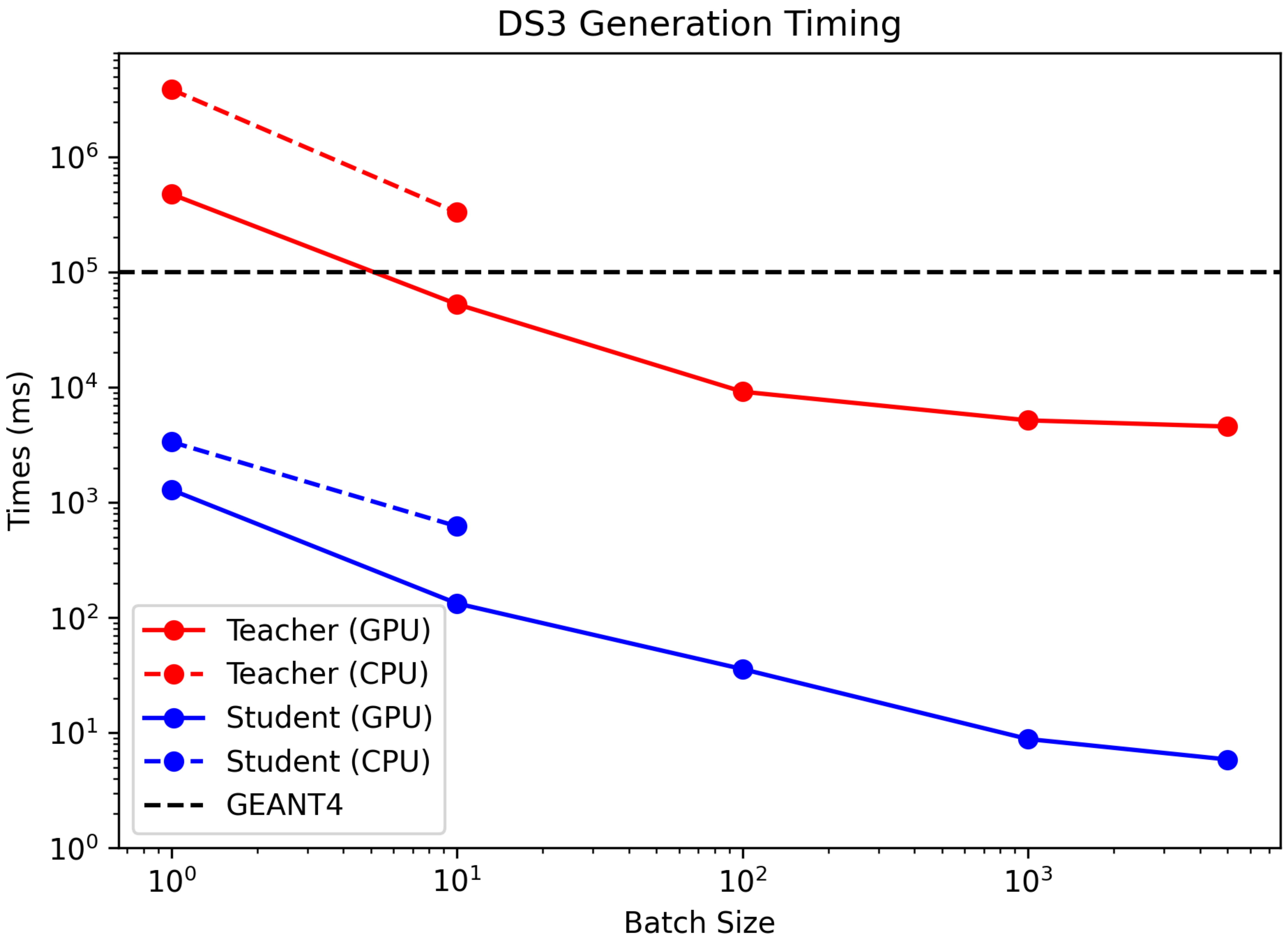
Generation timing (DS3)



Generation timing (DS3)



Generation timing (DS3)



Summary

- CaloFlow for Dataset 1 [arXiv: 2210.14245]
    - Good performance on DS 1 (histograms, χ^2/NDF, classifier scores)
    - Up to $\mathcal{O}(10^5)$ speed up compared to GEANT4

- iCaloFlow for Dataset 2 & 3 [arXiv: 2305.11934]
    - Inductive approach works quite well!
    - Memory efficient (only 3 flows)
    - Up to $\mathcal{O}(10^4)$-$\mathcal{O}(10^5)$ speed up compared to GEANT4
    - Future work: Coupling-layer flows realization of inductive setup

Thank you!

Backup

Pre-processing for CaloFlow

- Flow-1
  - **Logit and Unit**-space transformed E_i :

    $$u_1 = \frac{\sum_{i=1}^{N} E_i}{\beta E_{\text{inc}}},\ u_i = \frac{E_i}{\sum_{j=i}^{N} E_j} \text{ where } \beta \text{ is constant}$$

    $$\tilde{u}_i = \alpha + (1 - 2\alpha) u_i \quad \text{and} \quad \alpha = 10^{-6}.$$

    $$u_{\text{logit},i} = \log \frac{\tilde{u}_i}{1 - \tilde{u}_i},$$

  - **Log**-transformed E_{inc} (E_{inc} normalized by constant)

    $$\log_{10}\left(E_{\text{inc}}/33.3 \text{ GeV}\right) \in [-2.5, 2.5]$$

- Flow-2
  - **Log**-transformed E_{inc} and E_i (E_{inc} and E_i normalized by constant)

    $$\log_{10}\left(\left(E_i + 1 \text{ keV}\right)/100 \text{ GeV}\right) - 1 \in [-2, 4]$$

  - **Logit**-transformed voxel energies (voxel energies normalized by E_i)

Architecture for CaloFlow

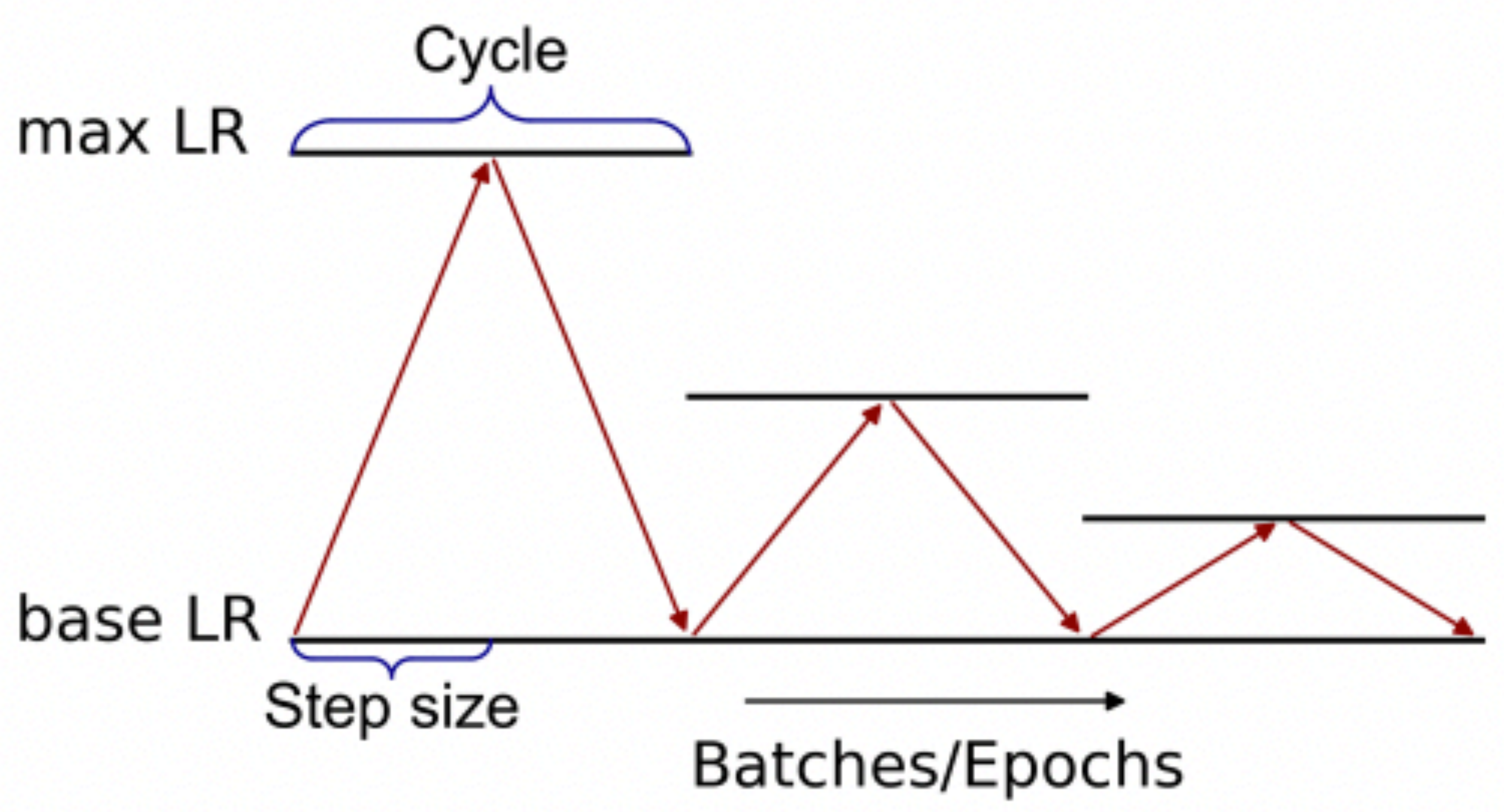
		input	hidden	output
γ	Teacher	378	1×378	8464
	Student	736	1×736	8464
π^+	Teacher	533	1×533	12259
	Student	500	1×500	12259

Training details of CaloFlow

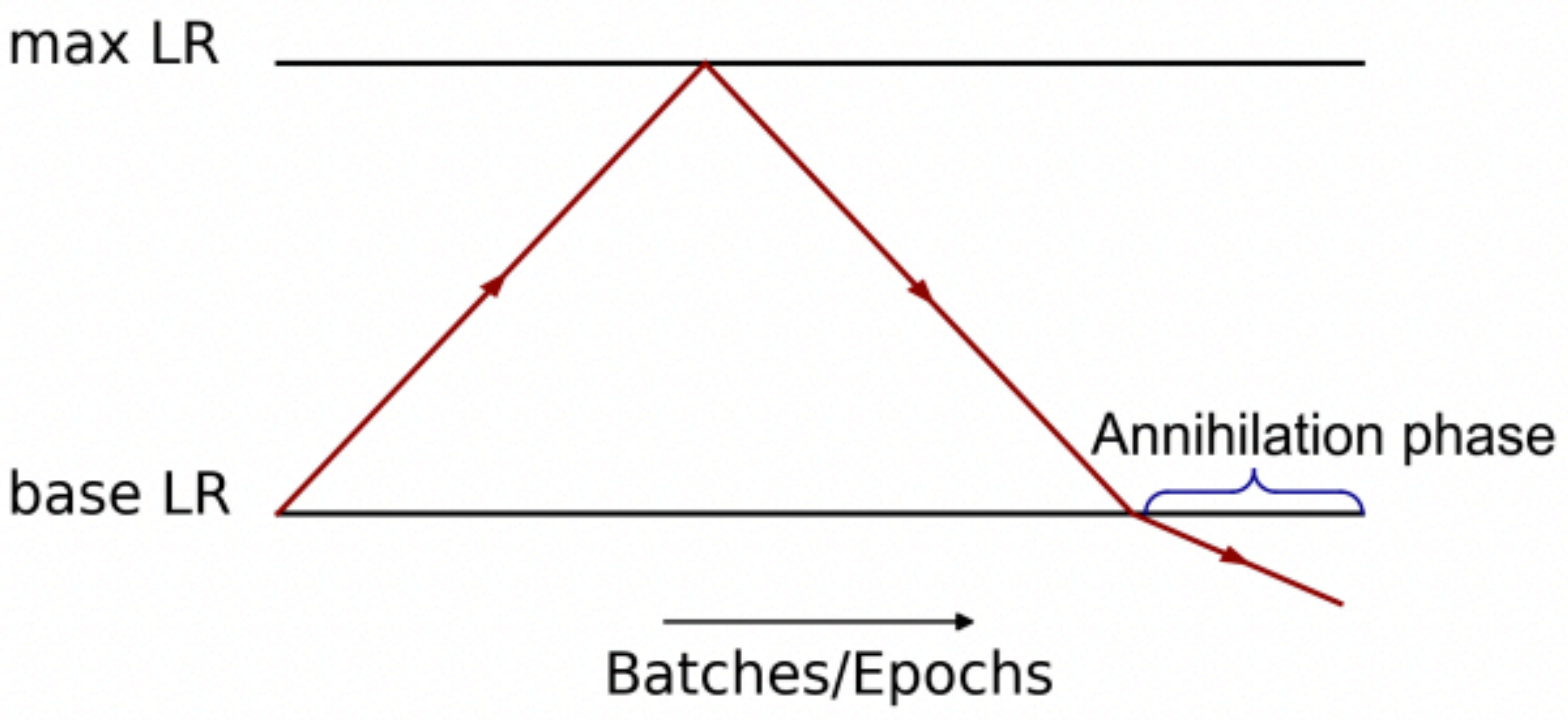
- Noise added to voxels during training
  - Uniform random of noise **[0, 0.1] keV** for π^+ flow-1
  - Uniform random of noise **[0, 1] keV** for other models

- LR schedule
  - **Regular Cyclic LR** for γ teacher
  - **OneCycle LR** for other models

Cyclic LR



(a) Regular cyclic LR



(b) OneCycle LR

Pre-processing for iCaloFlow

- Flow-1
  - **Log**-transformed E_{inc} (E_{inc} normalized by constant)
  - **Logit**-transformed E_i (E_i normalized by constant)

$$E_{\text{inc}} \to \log_{10} \frac{E_{\text{inc}}}{10^{4.5}\ \text{MeV}} \in [-1.5, 1.5].$$

$$E_i \to x_i \equiv (E_i + \text{rand}[0, 5\ \text{keV}])/65\ \text{GeV}.$$

$$y_i = \log \frac{u_i}{1-u_i}, \quad u_i \equiv \alpha + (1-2\alpha)x_i,$$

- Flow-2 & 3
  - **Log**-transformed E_{inc} (E_{inc} normalized by constant)
  - **Logit**-transformed E_i (E_i normalized by constant)
  - **Logit**-transformed voxel energies (voxel energies normalized by E_i)

$$\mathcal{I}_{ia} \to \mathcal{I}_{ia} + \text{rand}[0, 5\ \text{keV}]$$

$$\mathcal{I}_{ia} \to \mathcal{I}_{ia} / \sum_b \mathcal{I}_{ib} \equiv \hat{\mathcal{I}}_{ia}$$

$$u_{ia} = \alpha + (1-2\alpha)\hat{\mathcal{I}}_{ia}$$

$$y_{ia} = \log \frac{u_{ia}}{1-u_{ia}}.$$

Architecture for iCaloFlow

		dim of base distribution	number of MADE blocks	layer sizes input	hidden	output	number of RQS bins
DS2	Flow-1	45	8	256	2×256	1035	8
	Flow-2 teacher	144	8	256	2×256	3312	8
	Flow-2 student	144	8	256	2×256	3312	8
	Flow-3 teacher	144	8	256	2×256	3312	8
	Flow-3 student	144	8	348	2×384	3312	8
DS3	Flow-I	45	8	256	2×256	1035	8
	Flow-2 teacher	900	8	256	2×256	20700	8
	Flow-2 student	900	8	256	2×256	20700	8
	Flow-3 teacher	900	8	256	1×256	20700	8
	Flow-3 student	900	8	256	1×256	20700	8

Training details of iCaloFlow

- Noise added during training
  - Uniform random of noise **[0, 5] keV** added to E_i and voxel energies

- LR schedule
  - **Multistep LR** for DS3 teacher (including flow-1)
  - **OneCycle LR** for other models