Micro from macro

(and applications in event generators)

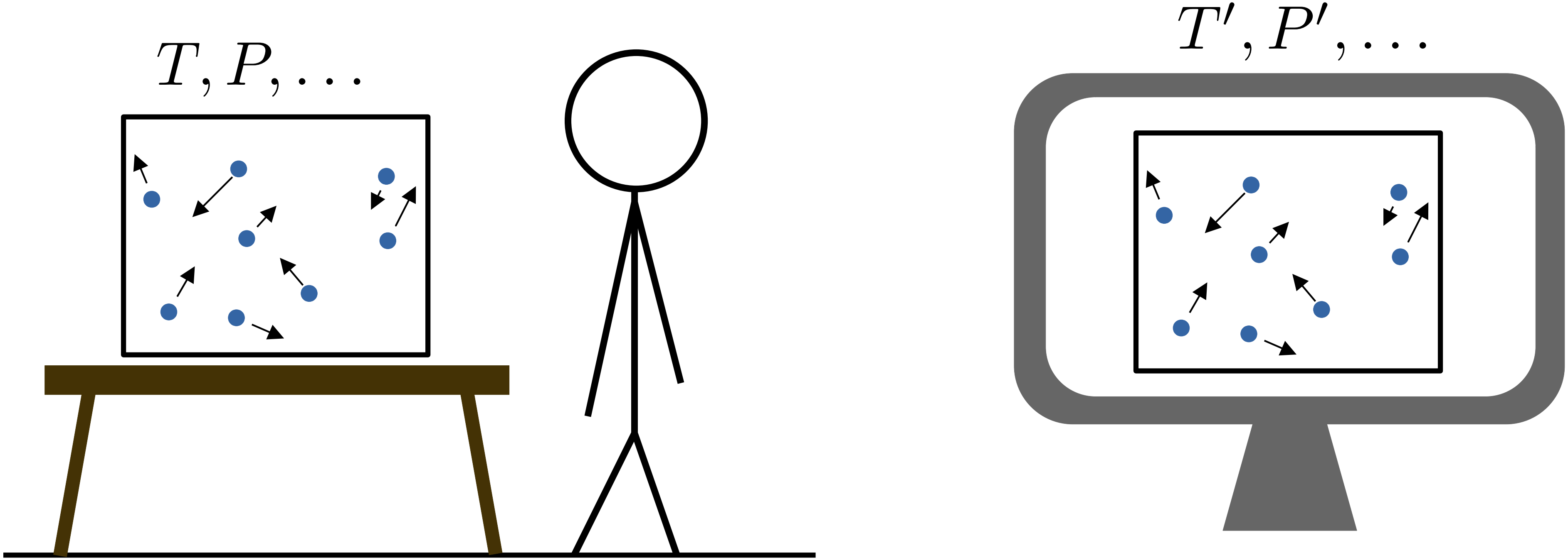
ATLAS Lunch

Tony Menzo

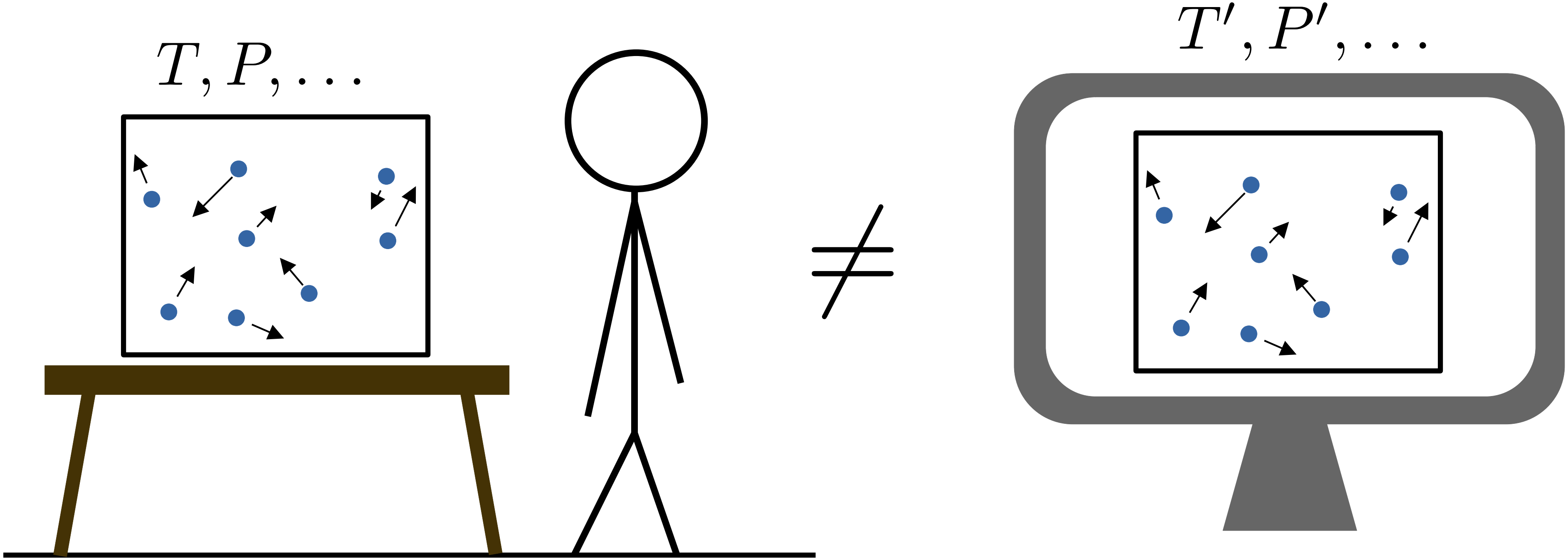
PhD candidate, University of Cincinnati

MLHAD

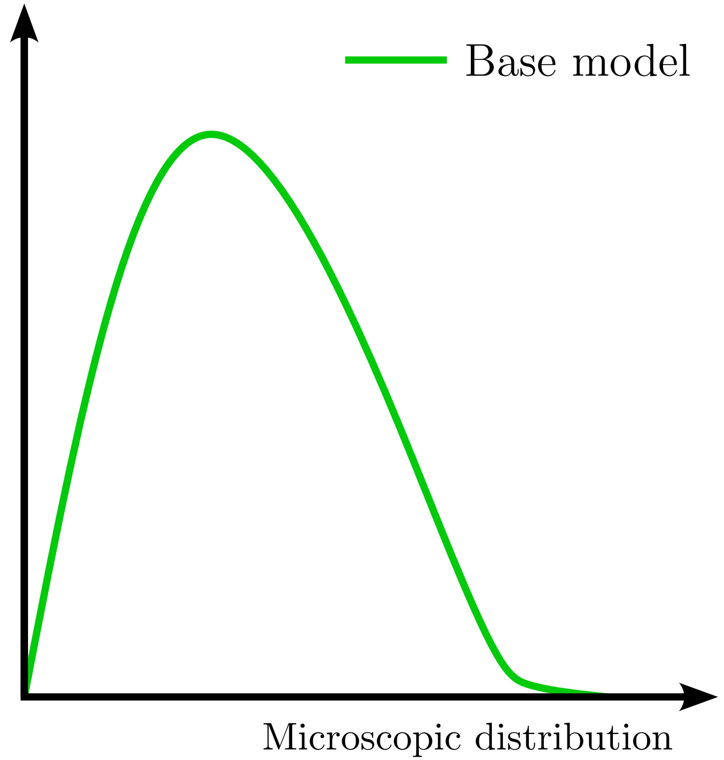
The problem.



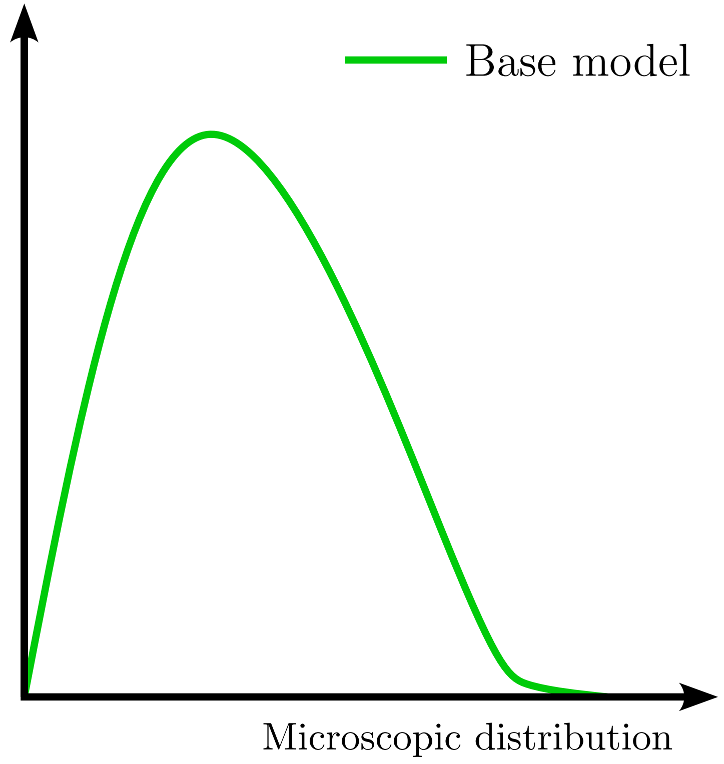
The problem.

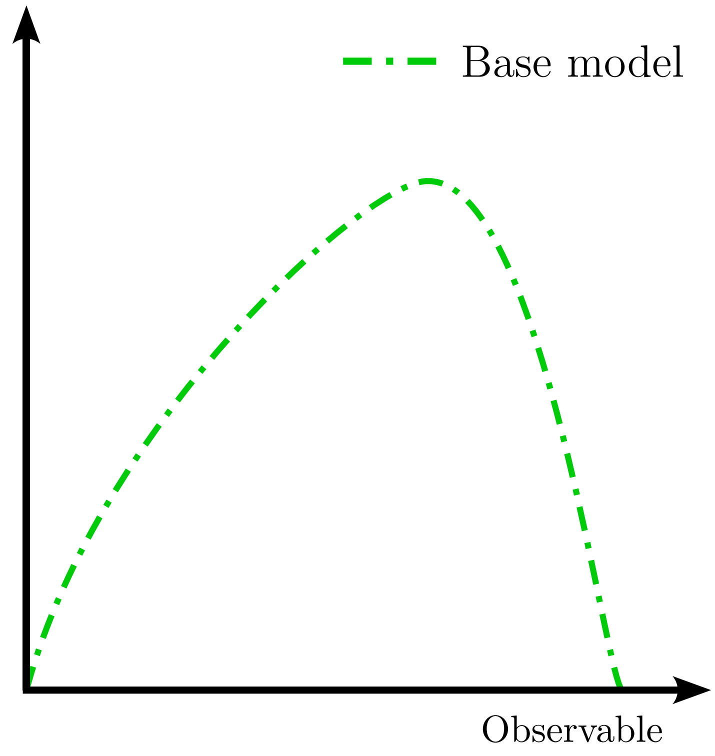


A potential solution

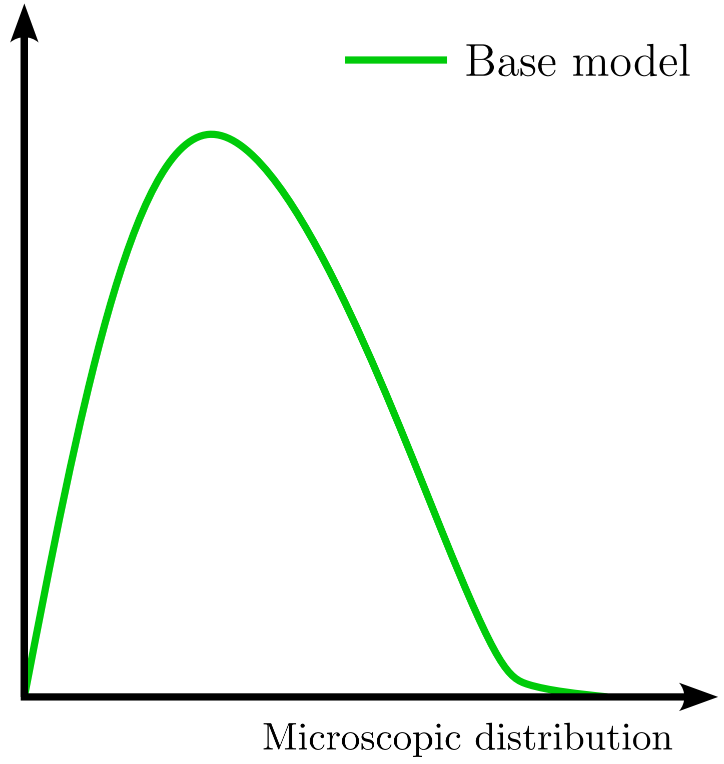


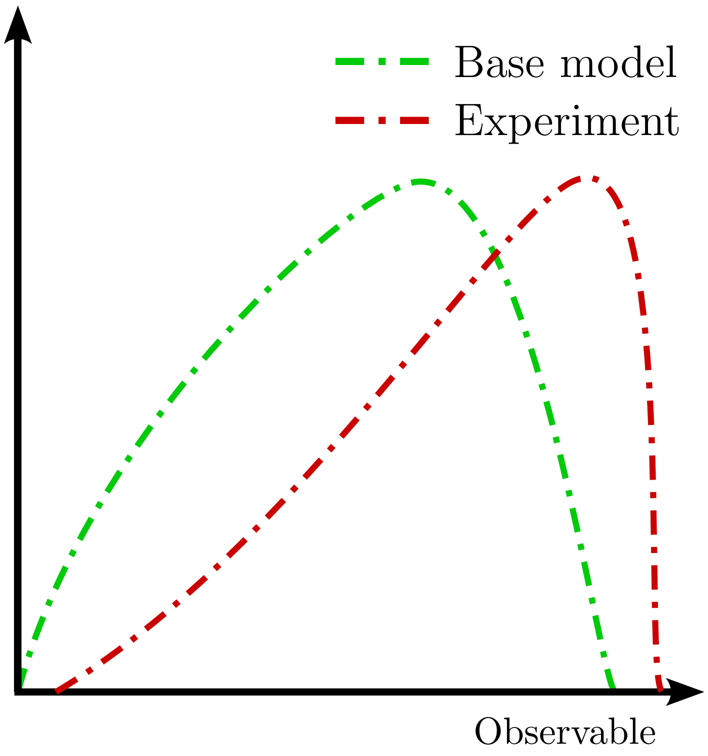
A potential solution



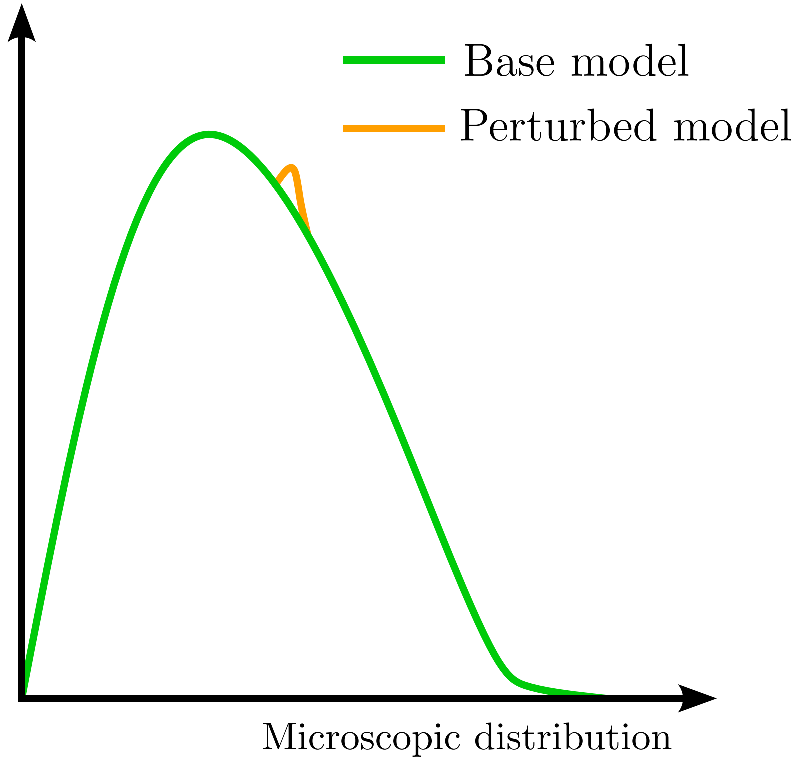


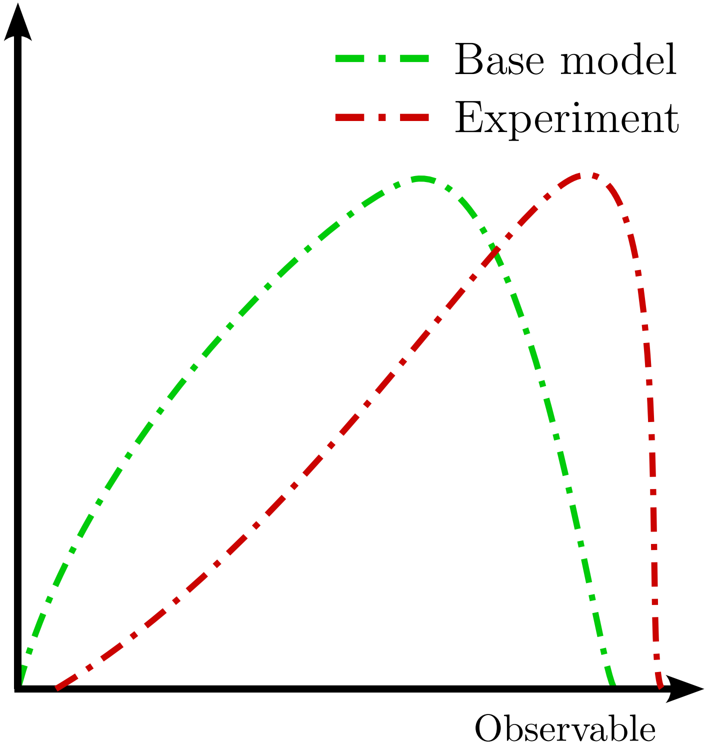
A potential solution



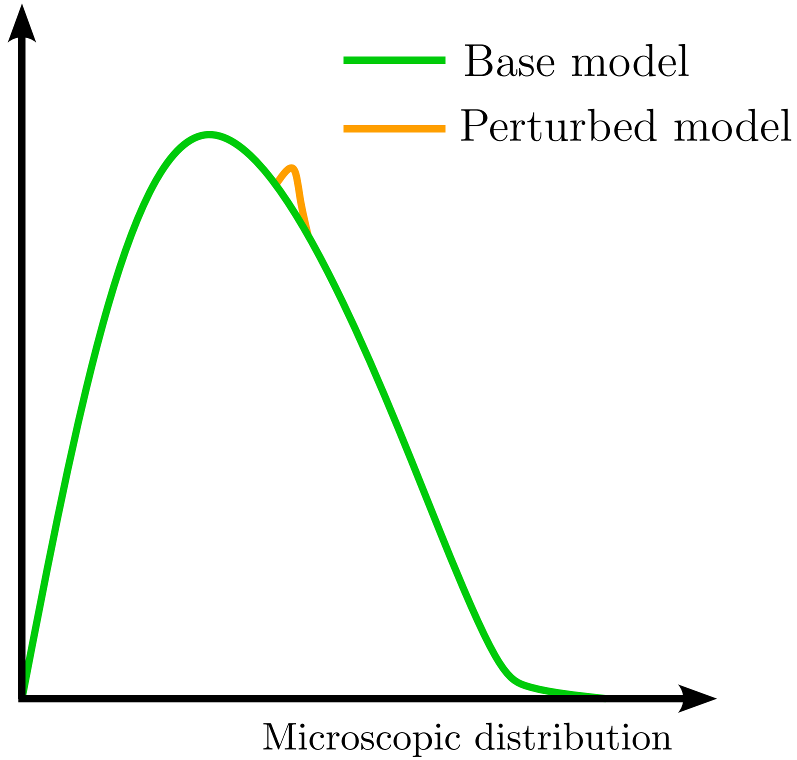


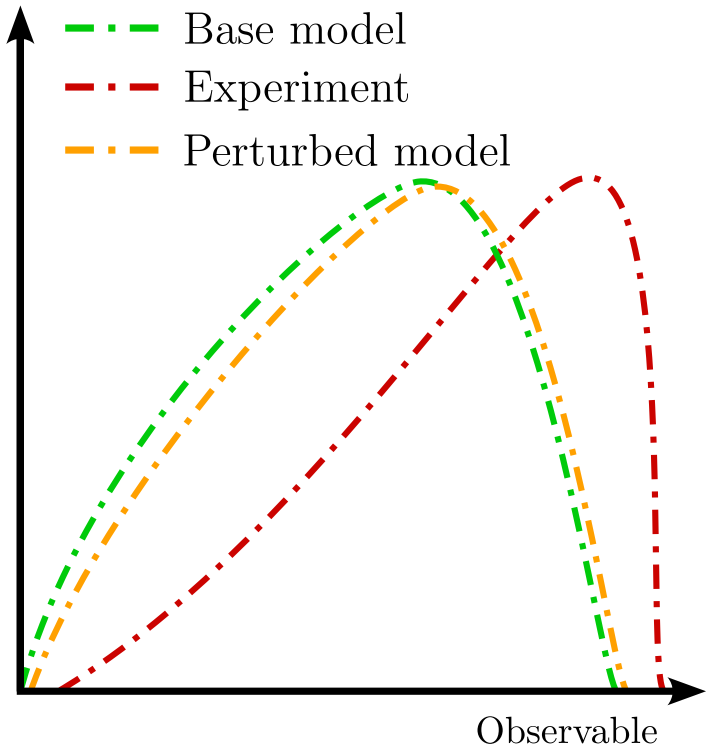
A potential solution



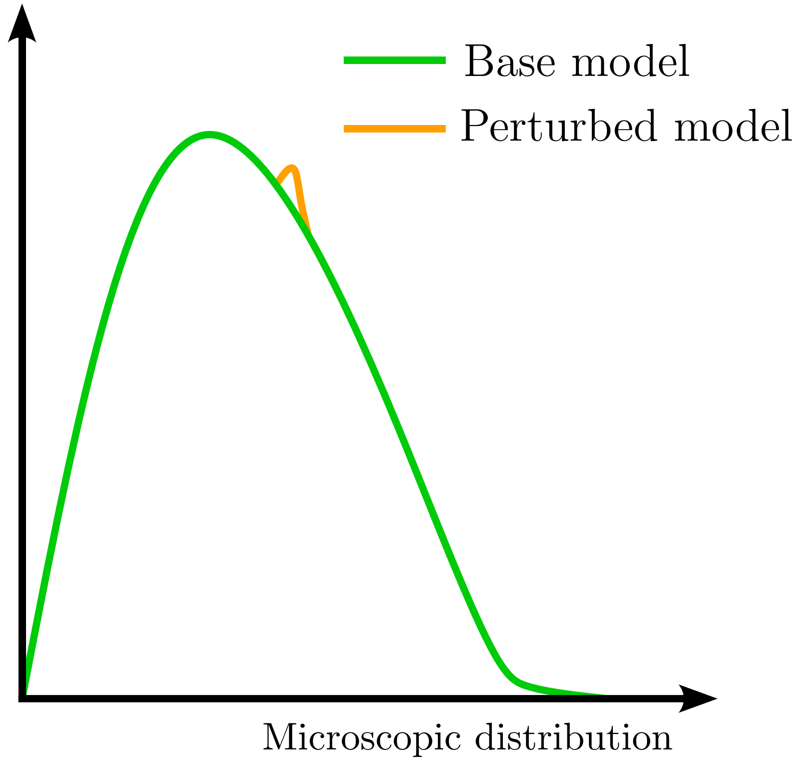


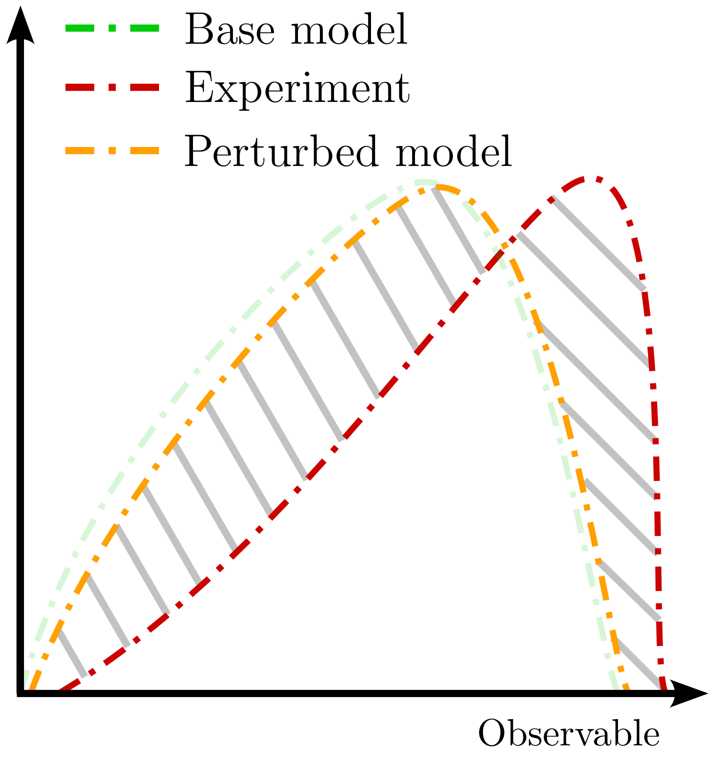
A potential solution



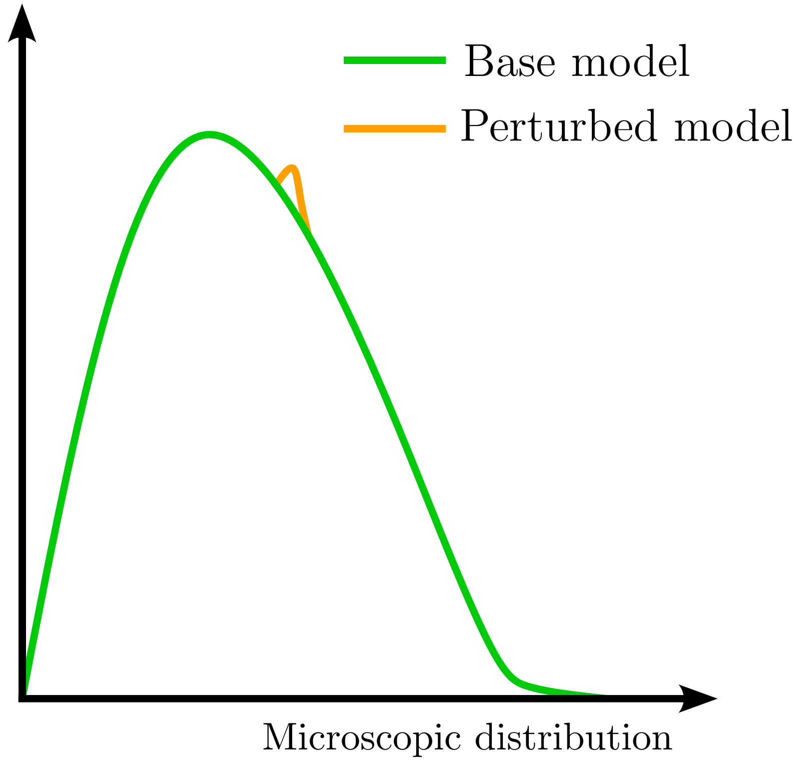


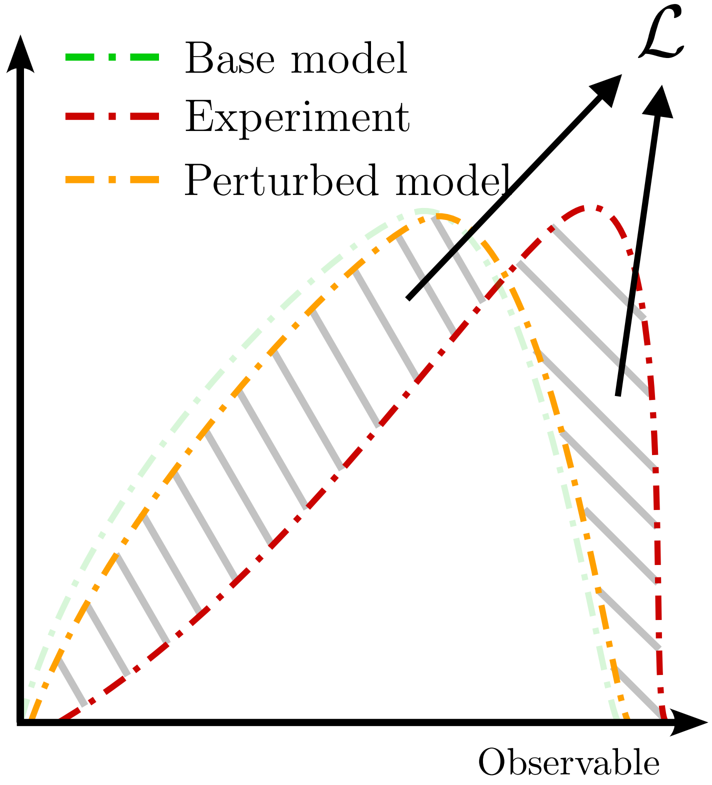
A potential solution



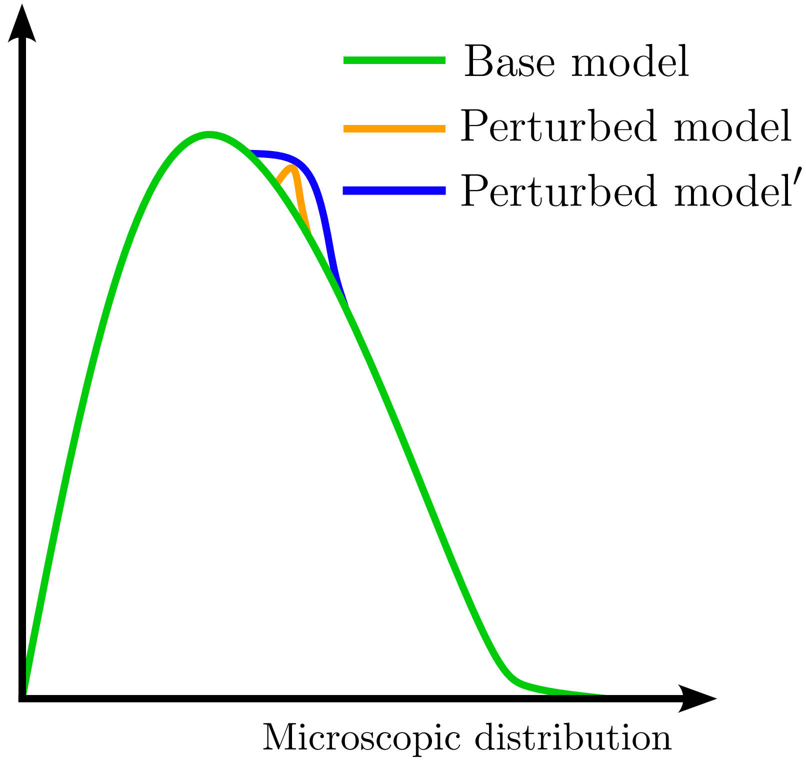


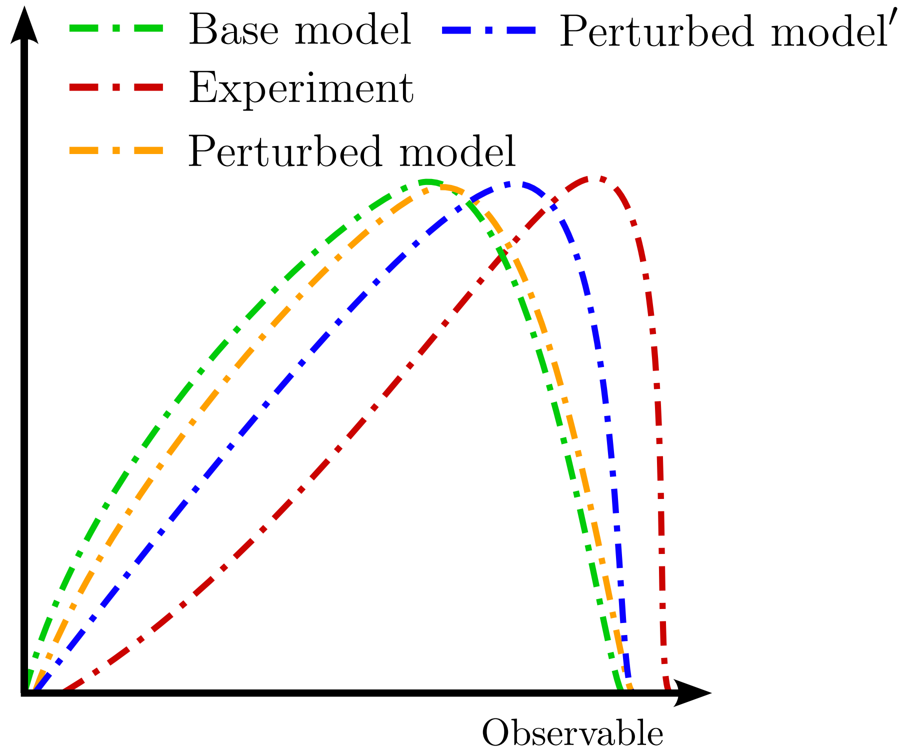
A potential solution



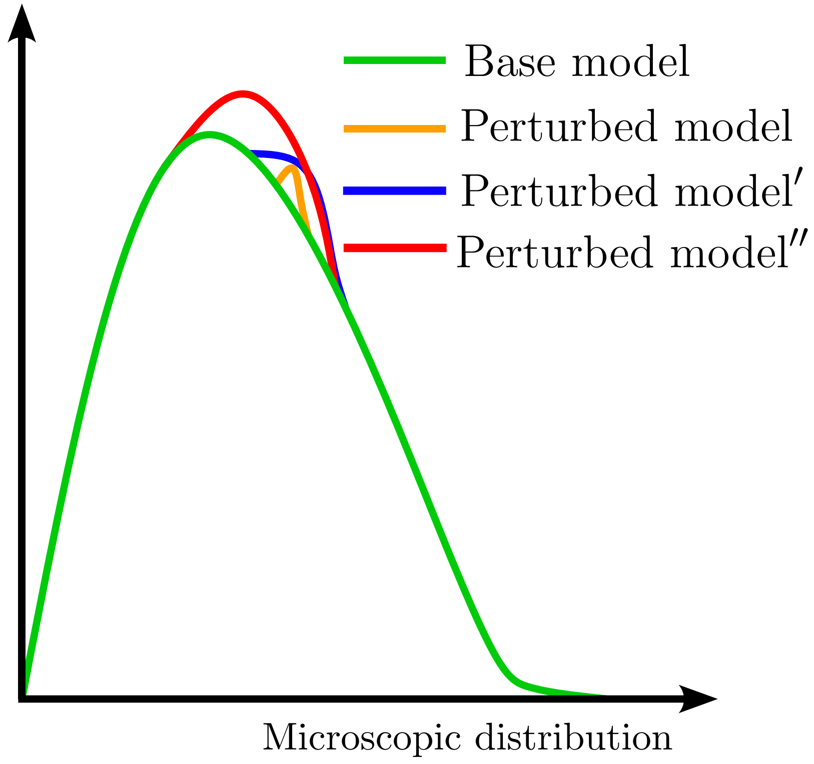


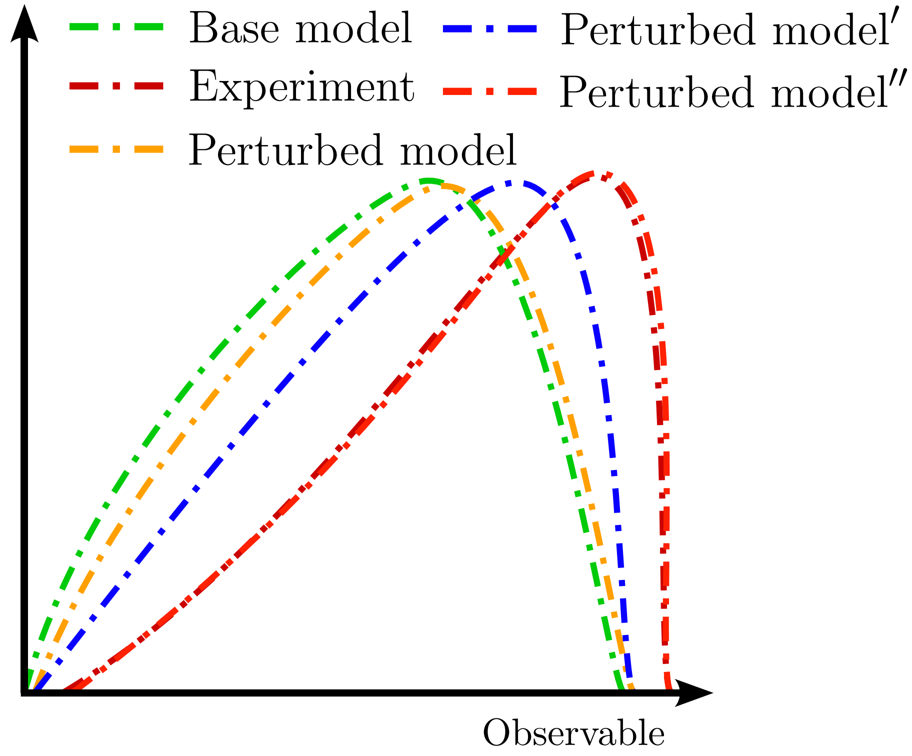
A potential solution





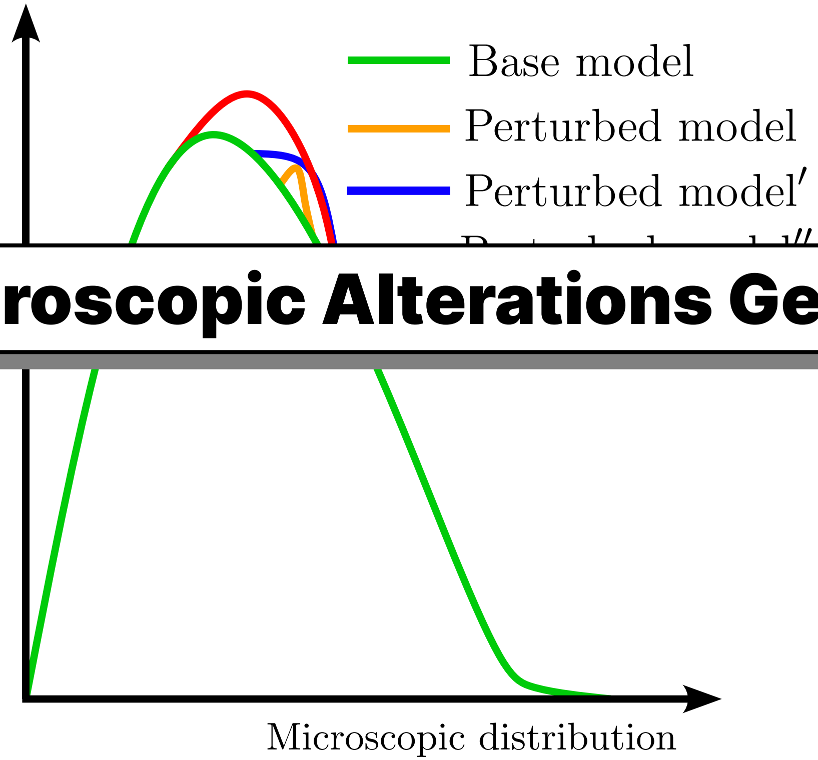
A potential solution

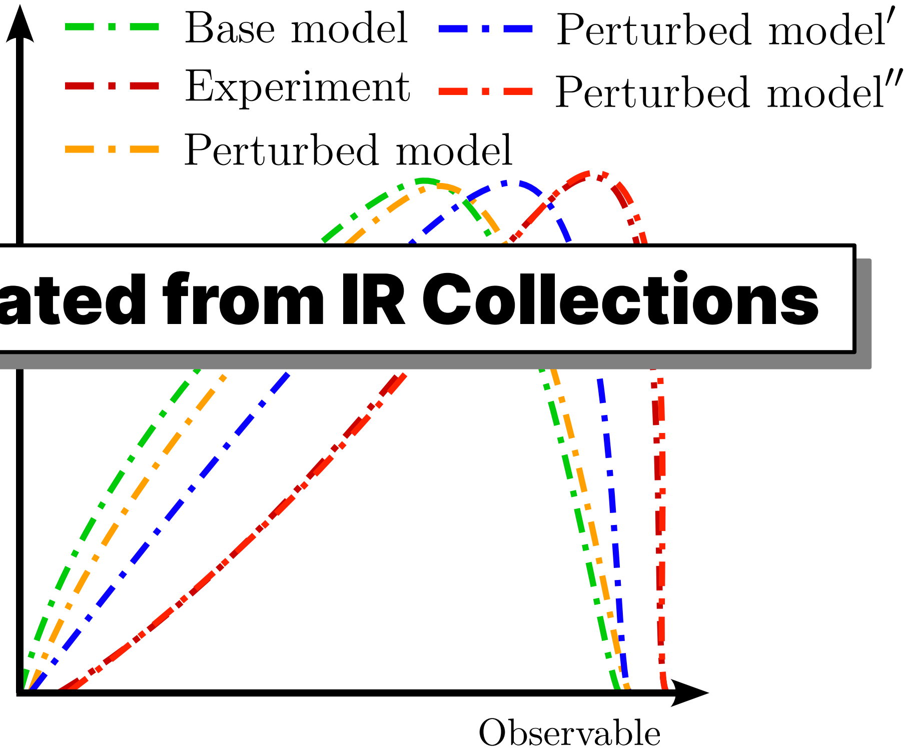




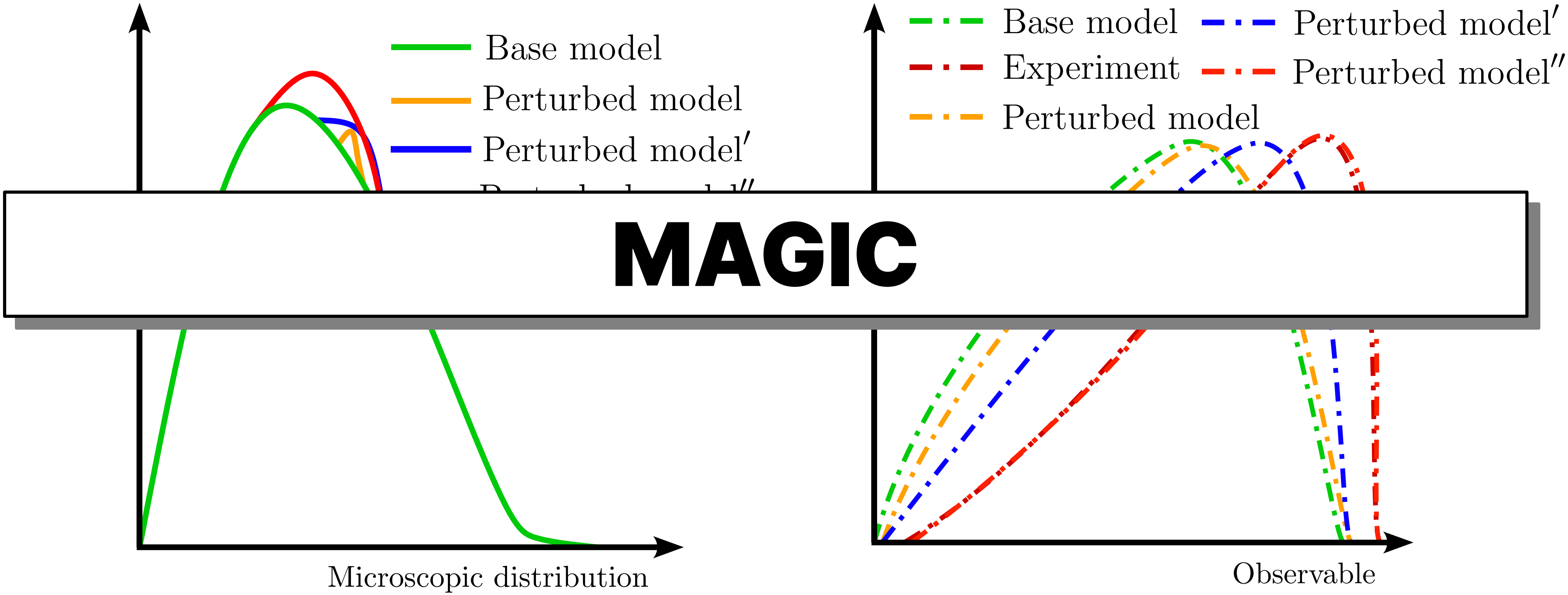
A potential solution

Microscopic Alterations Generated from IR Collections

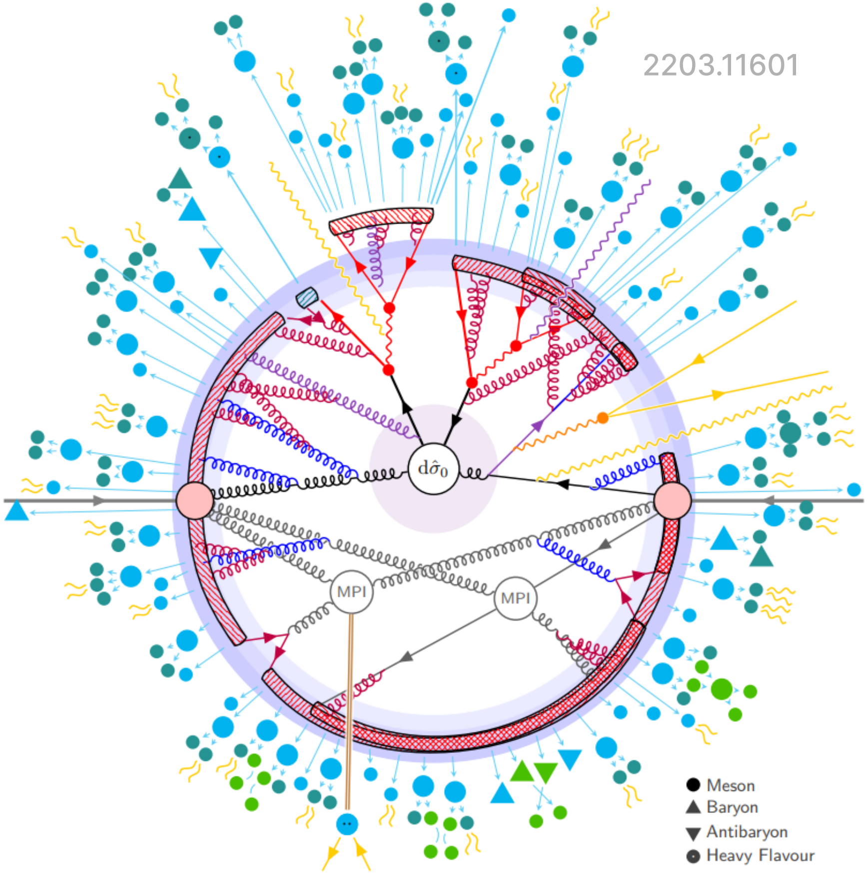




A potential solution



Hadronization



Considerations

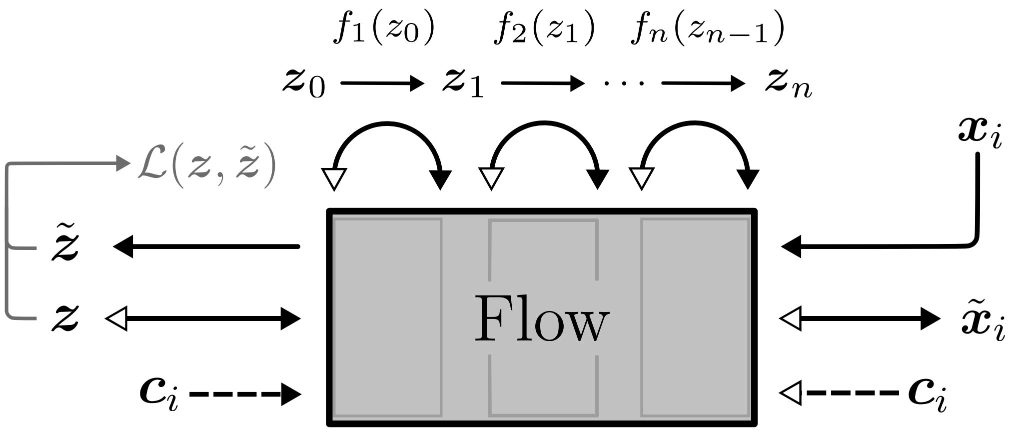
1. Error analysis
2. Incorporation into event generation
3. How to check that we're not masking other issues (e.g. parton shower effects)
4. How should the new solution be interpreted
5. Other applications

BACK-UP

Invertible nerual networks (INN)

a.k.a normalizing flow

Learn a composition of n independent bijective transformations that relate a probability distribution $p_Z(z)$ on latent space Z to the target distribution $p_X(x)$ on target space X.



The probability distribution for the random variable x = f(z) is given by

$$p_X^f(\boldsymbol{x}) = p_Z(\boldsymbol{z}) |\det J_f(\boldsymbol{z})|^{-1}$$

$$J_f = \partial f / \partial x$$

For n iterative transformations:

$$p_X^F(\boldsymbol{x}) = p_Z(\boldsymbol{z}_0) \prod_{i=1}^{n} |\det J_{f_i}(\boldsymbol{z}_{i-1})|^{-1}$$

Invertible nerual networks (INN)

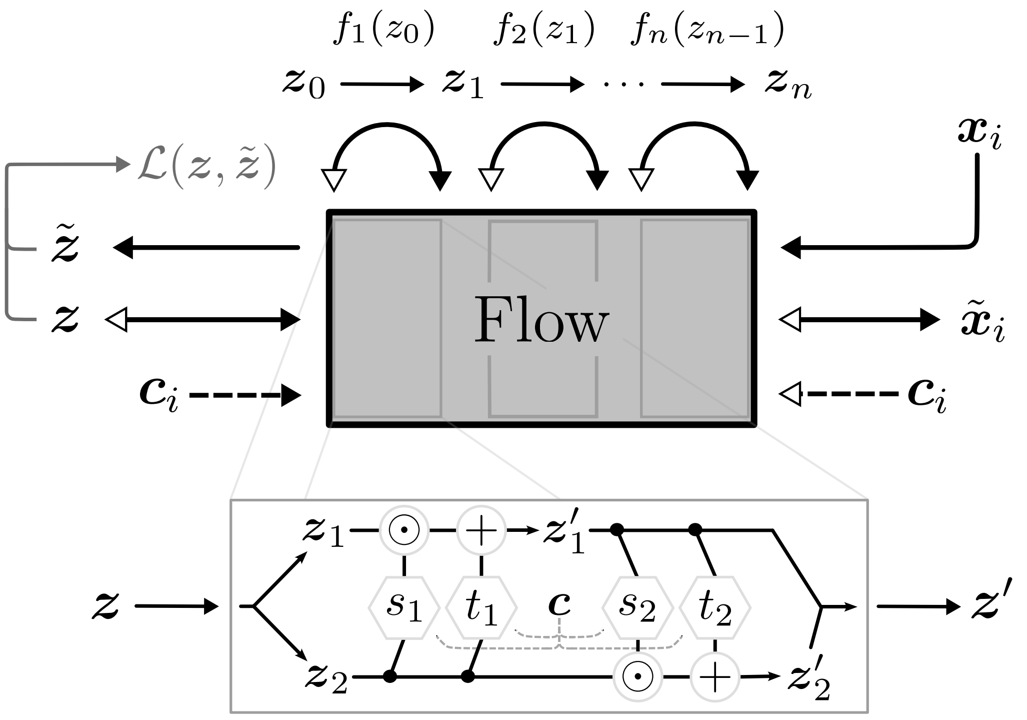
a.k.a normalizing flow

Invertible Real NVP transformations:

$$z_1' = z_1 \odot \exp\left(s_1\left(z_2\right)\right) + t_1(z_2),$$
$$z_2' = z_2 \odot \exp\left(s_2\left(z_1'\right)\right) + t_2(z_1'),$$

Inverse:

$$z_2 = \left(z_2' - t\left(z_1'\right)\right) \odot \exp\left(-s\left(z_1'\right)\right),$$
$$z_1 = \left(z_1' - t\left(z_2\right)\right) \odot \exp\left(-s\left(z_2\right)\right),$$



Invertible nerual networks (INN)

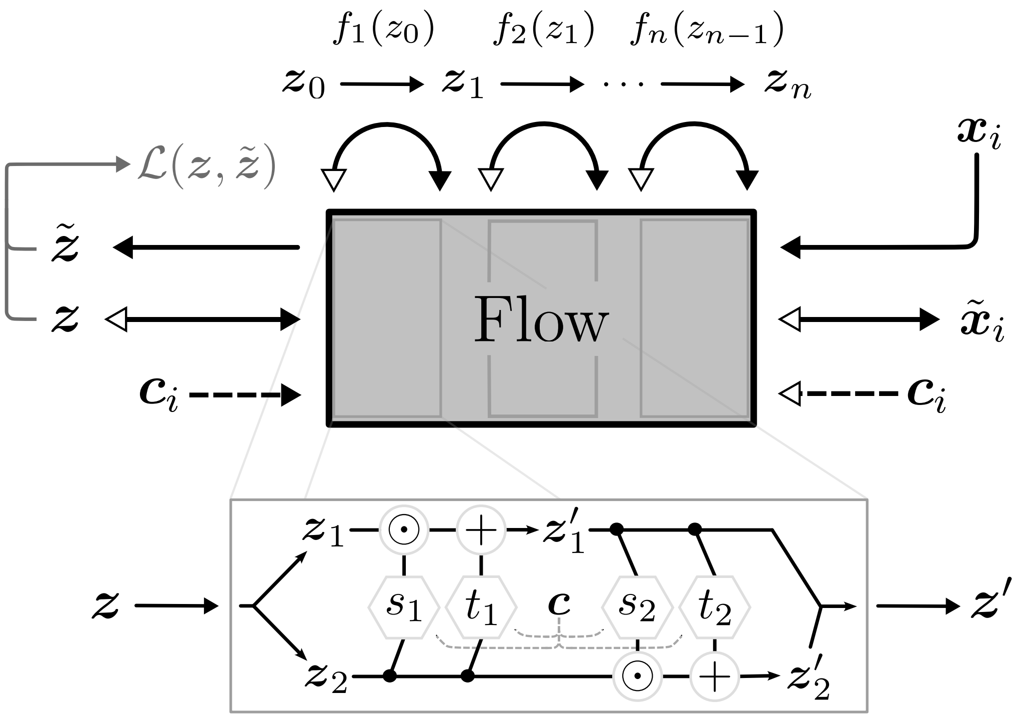
a.k.a normalizing flow

Invertible Real NVP transformations:

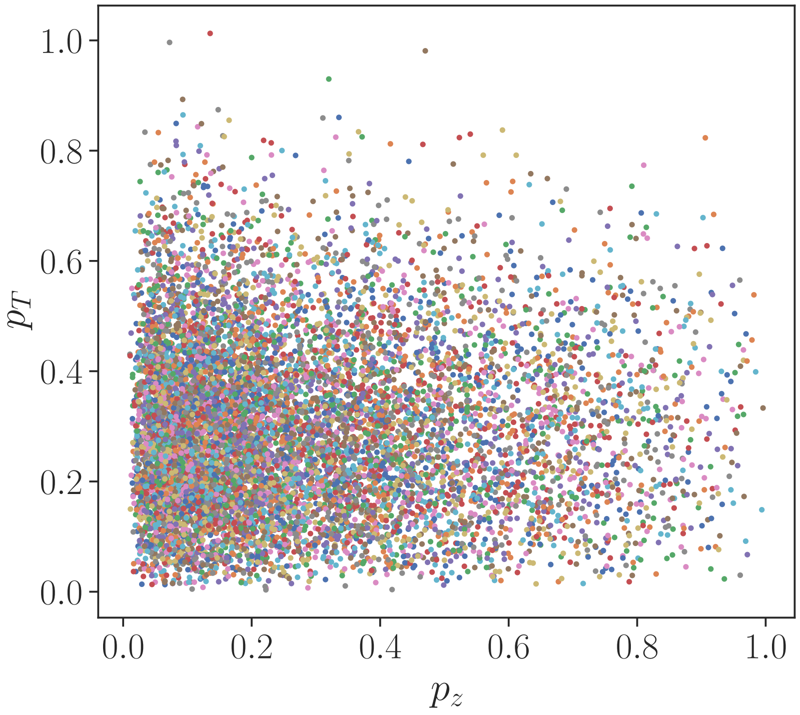
$$z'_1 = z_1 \odot \exp\left(s_1\left(z_2\right)\right) + t_1(z_2),$$
$$z'_2 = z_2 \odot \exp\left(s_2\left(z'_1\right)\right) + t_2(z'_1),$$

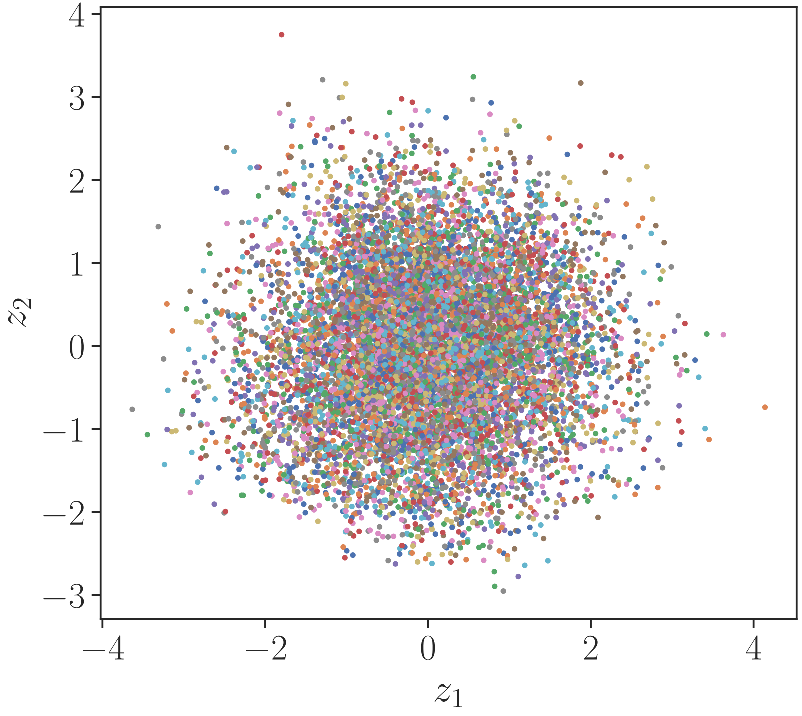
Scale transform

Translation transform

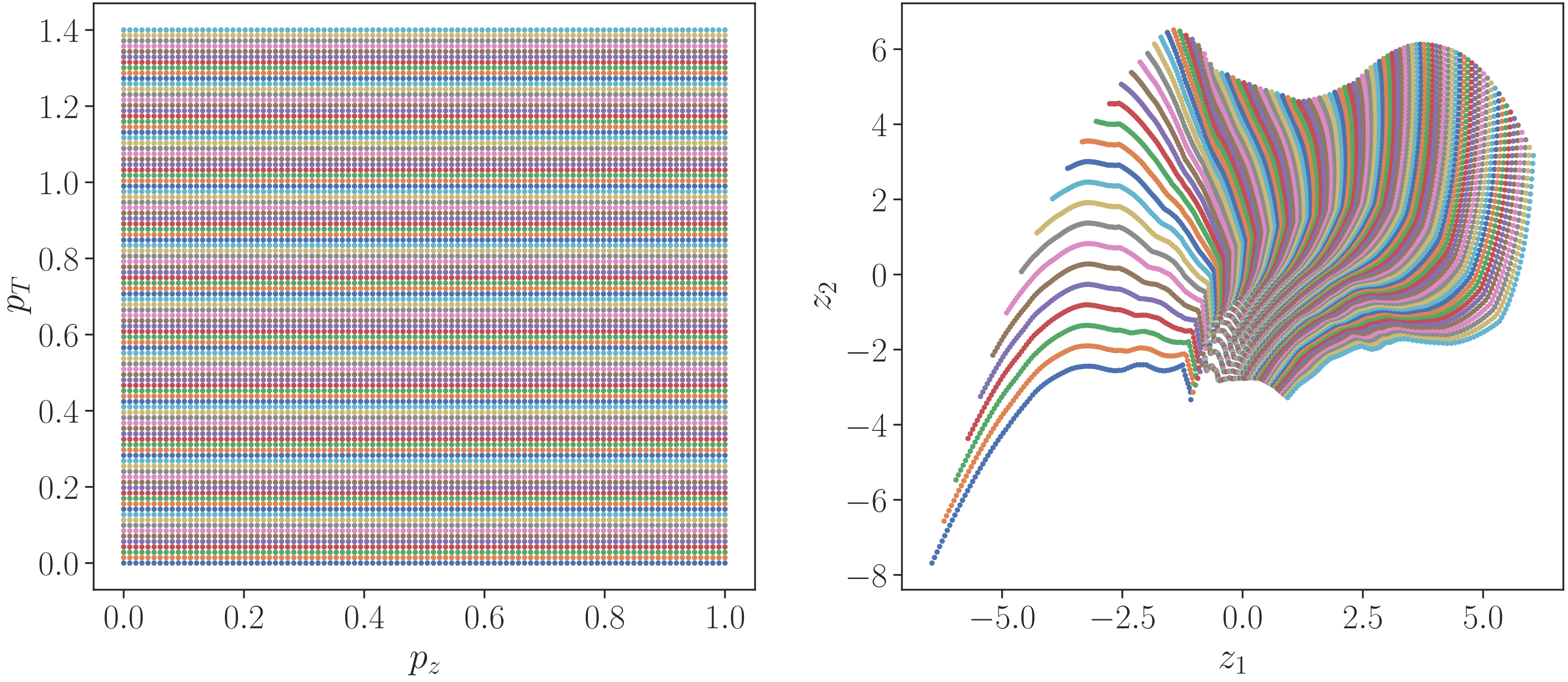


INN learned mapping

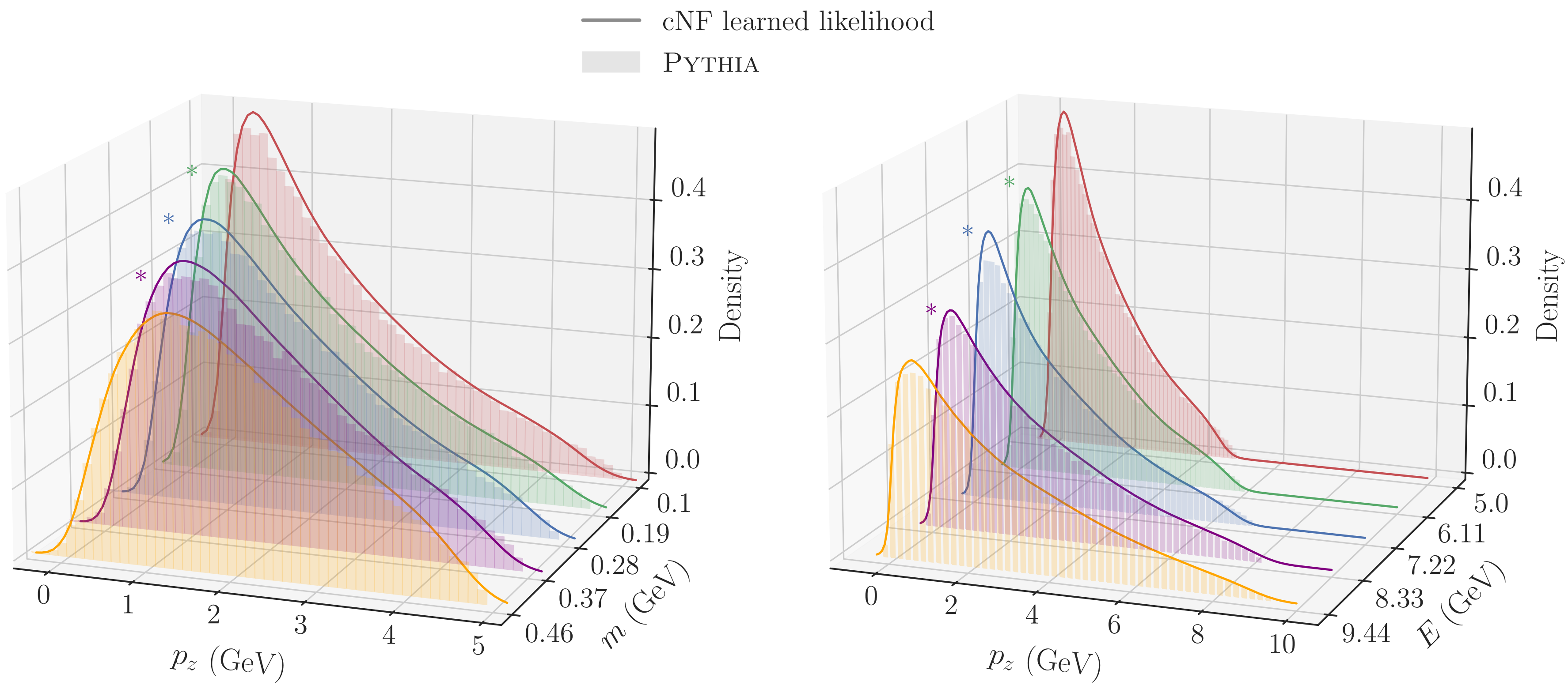




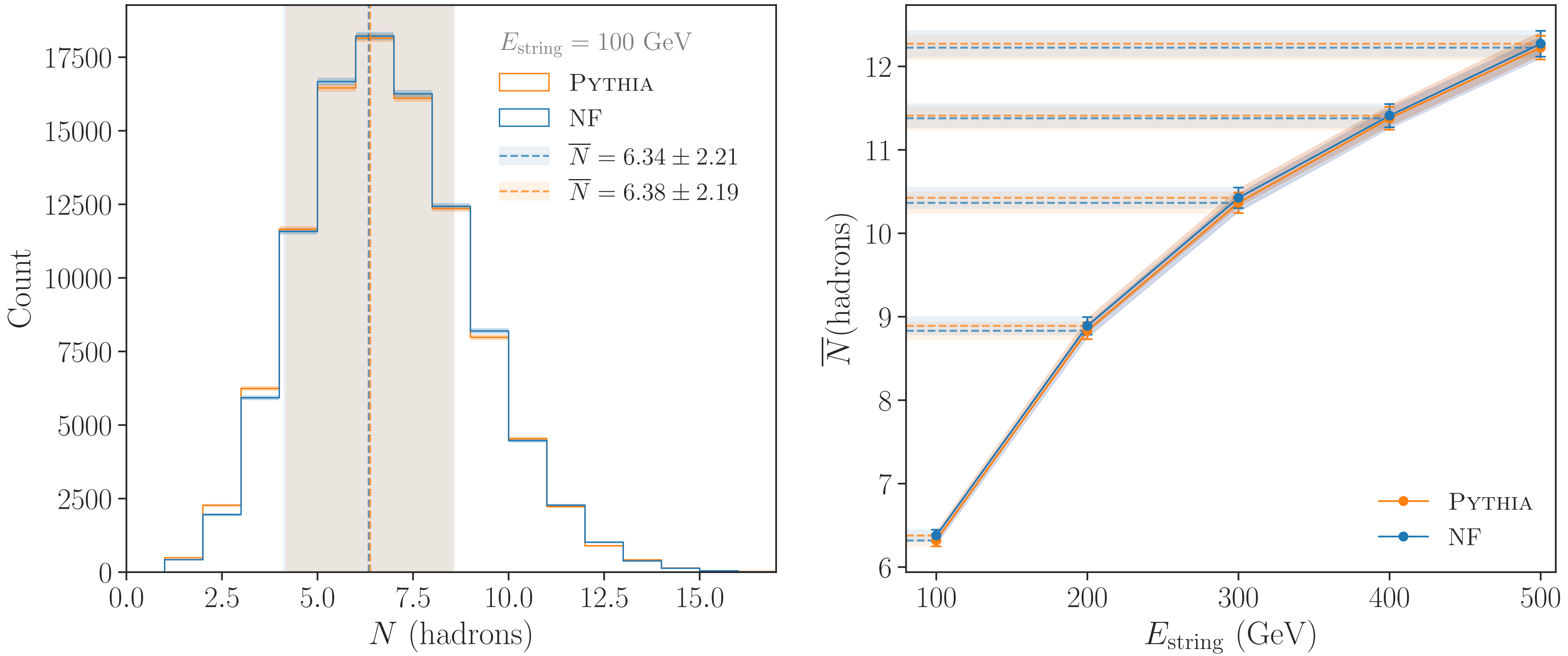
INN learned mapping



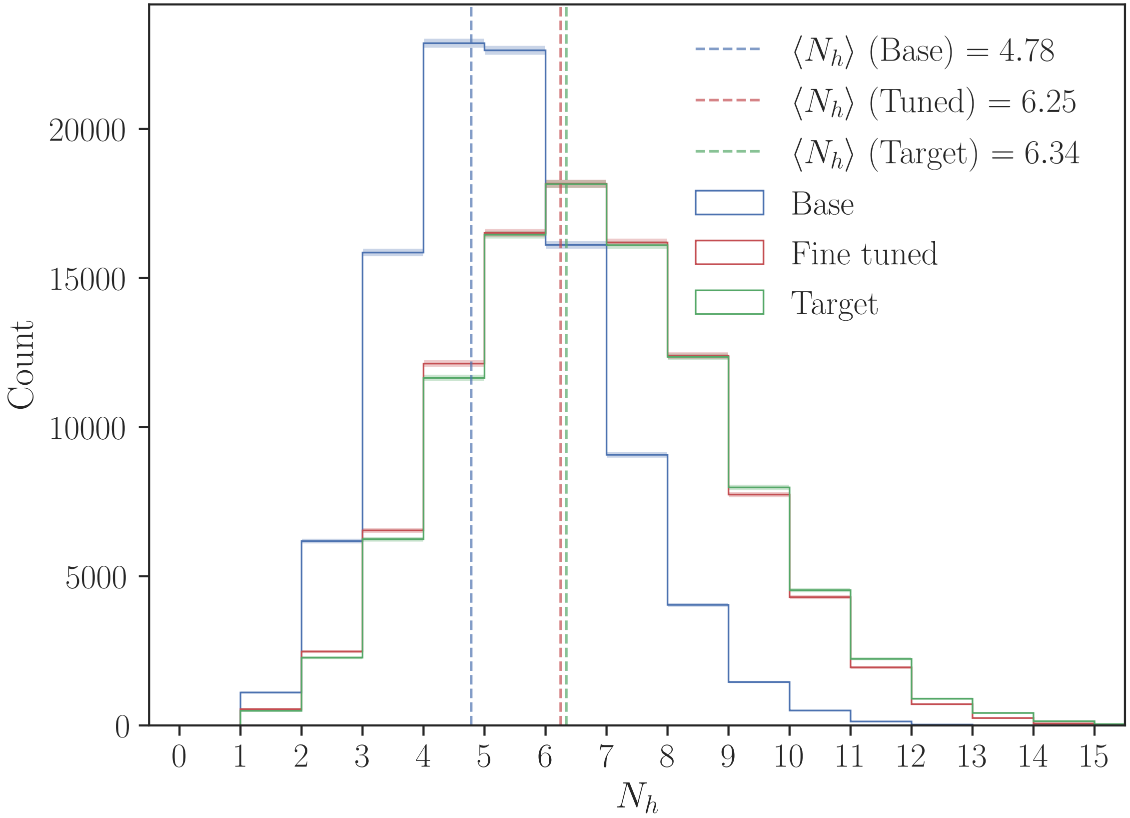
Validation: single emission kinematics



Validation: global observable (hadron multiplicity N)

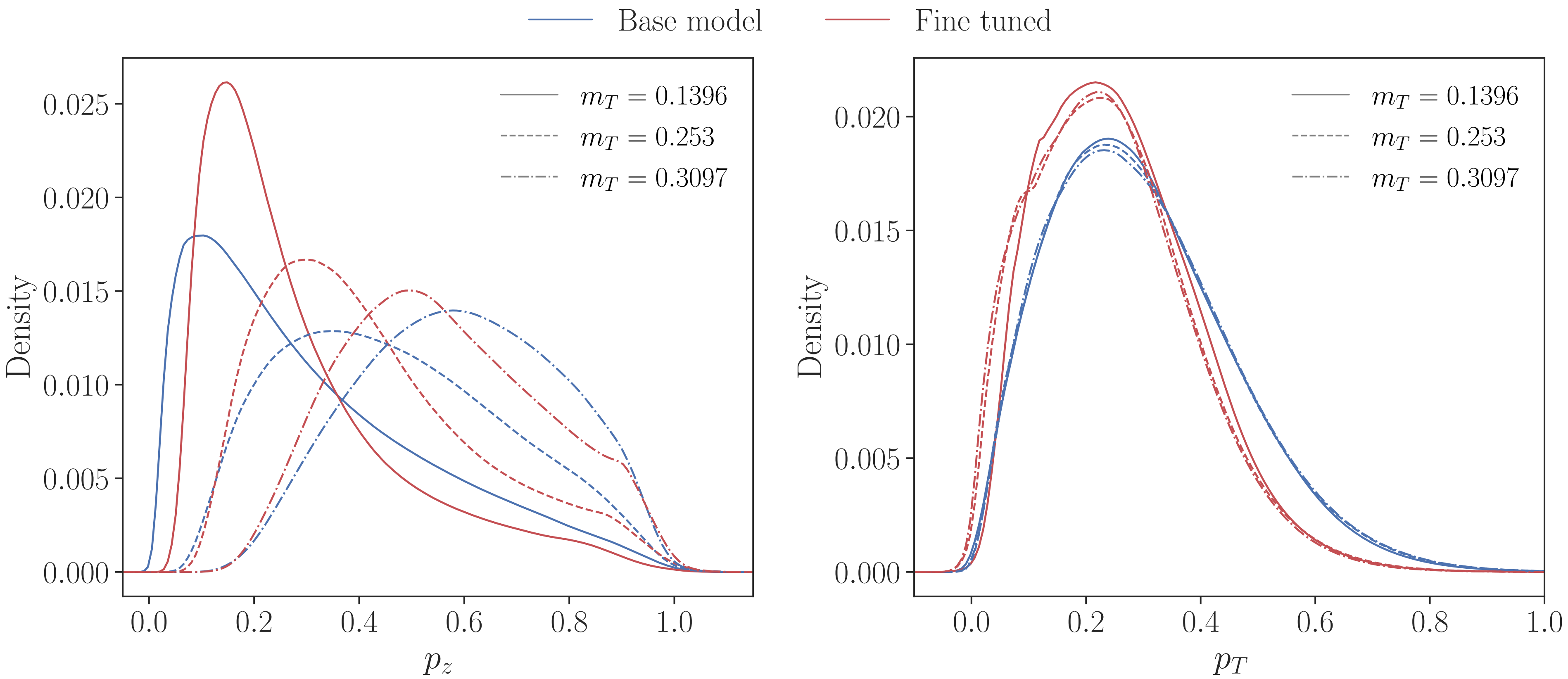


Toy example



$\langle N_h \rangle$ (Base) = 4.78
$\langle N_h \rangle$ (Tuned) = 6.25
$\langle N_h \rangle$ (Target) = 6.34
Base
Fine tuned
Target
Count
20000
15000
10000
5000
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
N_h

Toy example



Base model
Fine tuned
$m_T = 0.1396$
$m_T = 0.253$
$m_T = 0.3097$
Density
0.000
0.005
0.010
0.015
0.020
0.025
0.0
0.2
0.4
0.6
0.8
1.0
p_z
$m_T = 0.1396$
$m_T = 0.253$
$m_T = 0.3097$
Density
0.000
0.005
0.010
0.015
0.020
0.0
0.2
0.4
0.6
0.8
1.0
p_T