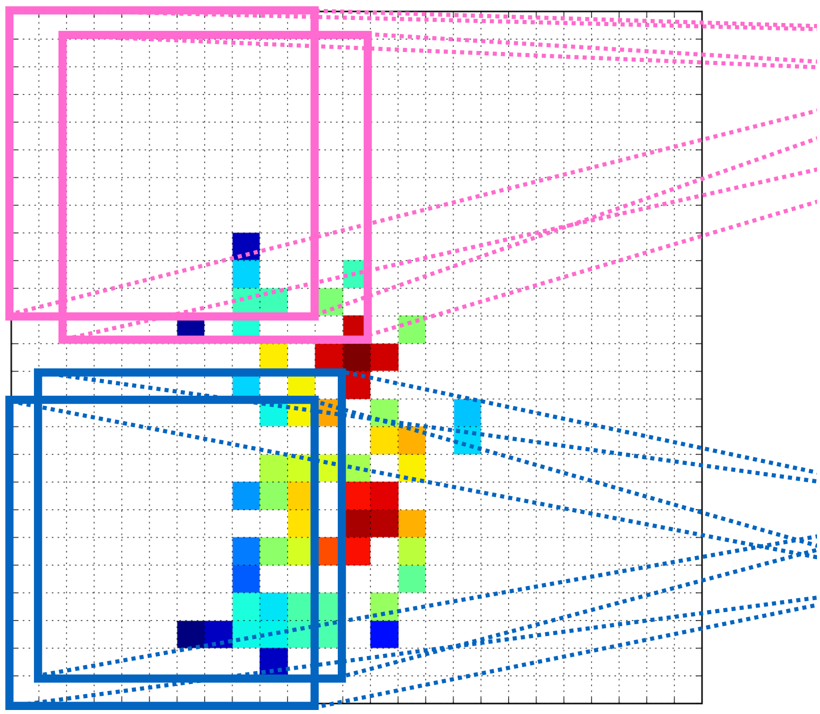
Differentiable Simulations

Benjamin Nachman

Lawrence Berkeley National Laboratory

bpnachman.com
bpnachman@lbl.gov

@bpnachman bnachman



BERKELEY LAB

BIDS
BERKELEY INSTITUTE FOR DATA SCIENCE

ai4eic
Oct. 11, 2022

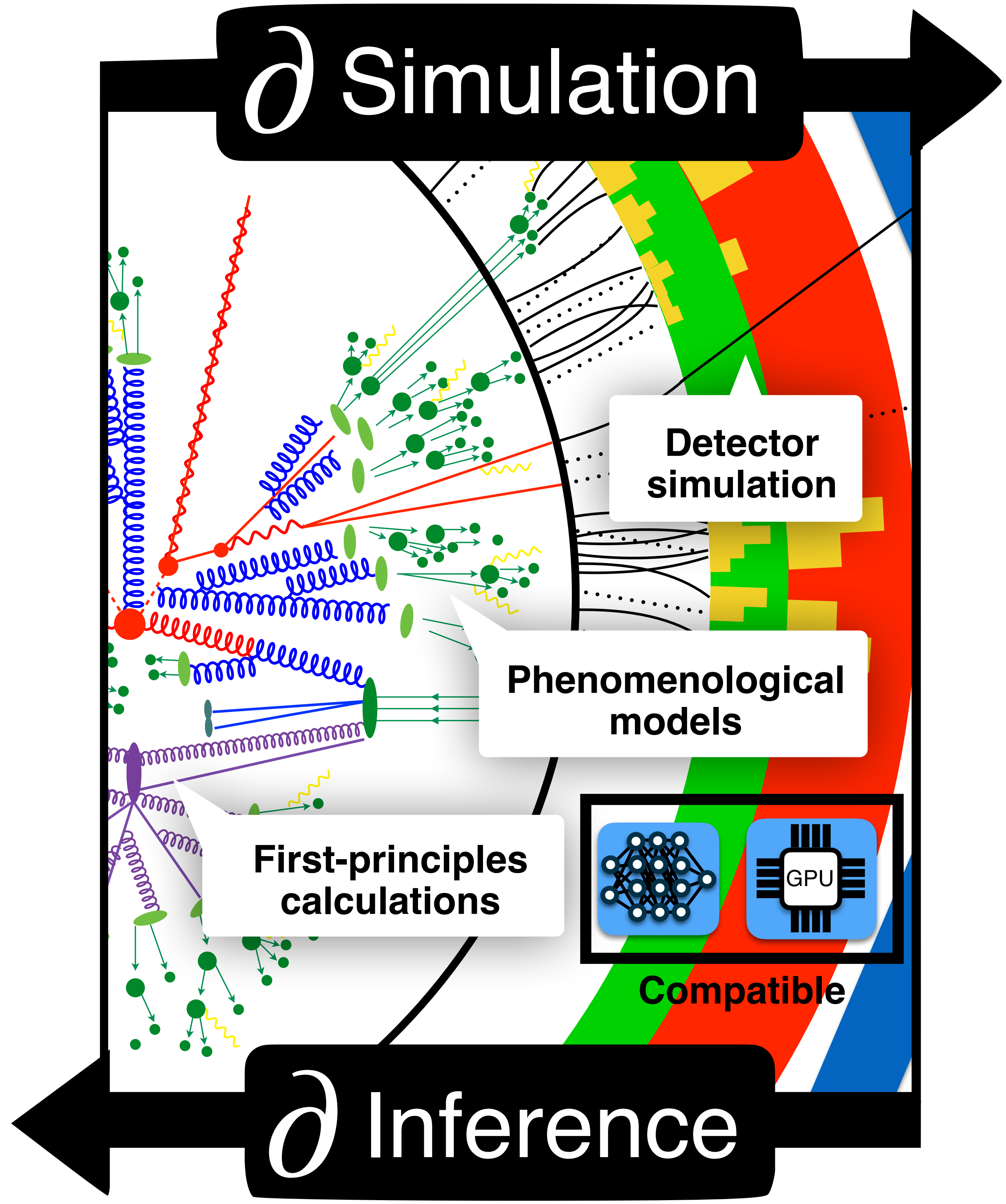
Differentiable Simulation

What is a differentiable simulation?

A differentiable parton shower

Other differentiable simulations

Outlook/ conclusions



Differentiable Simulation

$$X \sim \mathcal{N}(\mu, \sigma)$$

Differentiable Simulation

$$X \sim \mathcal{N}(\mu, \sigma)$$

$$\downarrow$$

```
x = np.random.normal(mu,sigma)
```

Differentiable Simulation

$$X \sim \mathcal{N}(\mu, \sigma)$$

↓

```
x = np.random.normal(mu,sigma)
```

↓

Removed randomness from simulator

```
Z = np.random.uniform(0,1)
x = sigma*Phiinv(z)+mu
```

(**`Phiinv`** = inverse Gaussian CDF)

Differentiable Simulation

$$X \sim \mathcal{N}(\mu, \sigma)$$

$\downarrow$

```
x = np.random.normal(mu,sigma)
```

$\downarrow$

```
Z = np.random.uniform(0,1)
x = sigma*Phiinv(z)+mu
```

(**Phiinv** = inverse Gaussian CDF)

Removed randomness from simulator

Now, can compute $\partial/\partial\mu$ and $\partial/\partial\sigma$

Differentiable Simulation

$$X \sim \mathcal{N}(\mu, \sigma)$$

$\downarrow$

```
x = np.random.normal(mu,sigma)
```

$\downarrow$

```
Z = np.random.uniform(0,1)
x = sigma*Phiinv(z)+mu
```

(**Phiinv** = inverse Gaussian CDF)

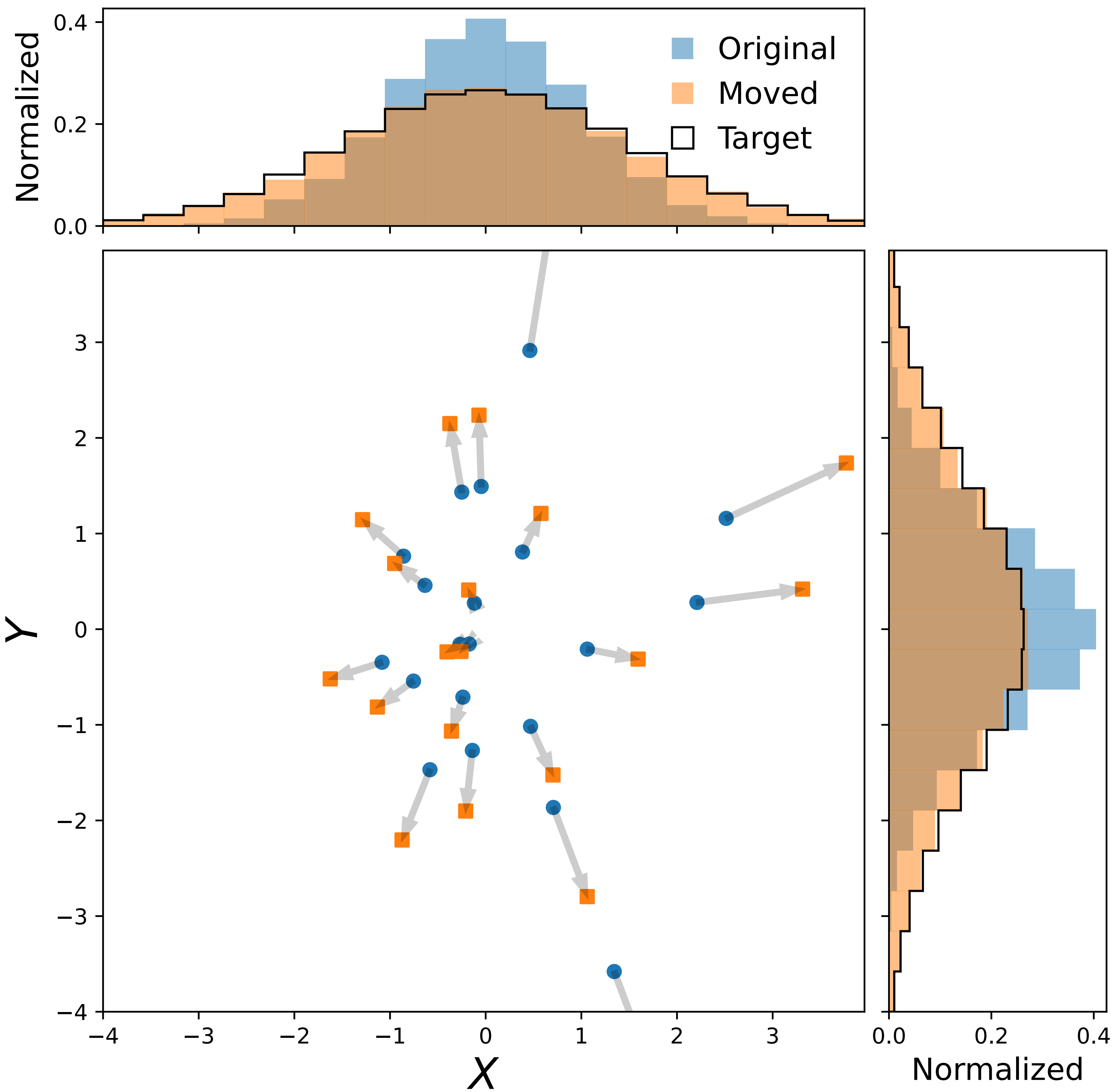
Removed randomness from simulator

Now, can compute $\partial/\partial\mu$ and $\partial/\partial\sigma$

We can then do:

$$\mathrm{sim}(\mu_0 + \epsilon) \approx \mathrm{sim}(\mu_0) + \frac{\partial \mathrm{sim}}{\partial \mu}\epsilon$$

Differentiable Simulation



$$X \sim \mathcal{N}(\mu, \sigma)$$

$\downarrow$

```
x = np.random.normal(mu,sigma)
```

$\downarrow$

```
z = np.random.uniform(0,1)
x = sigma*Phiinv(z)+mu
```

(**Phiinv** = inverse Gaussian CDF)

Now, can compute $\partial/\partial\mu$ and $\partial/\partial\sigma$

We can then do:

$$\text{sim}(\mu_0 + \epsilon) \approx \text{sim}(\mu_0) + \frac{\partial \text{sim}}{\partial \mu}\epsilon$$

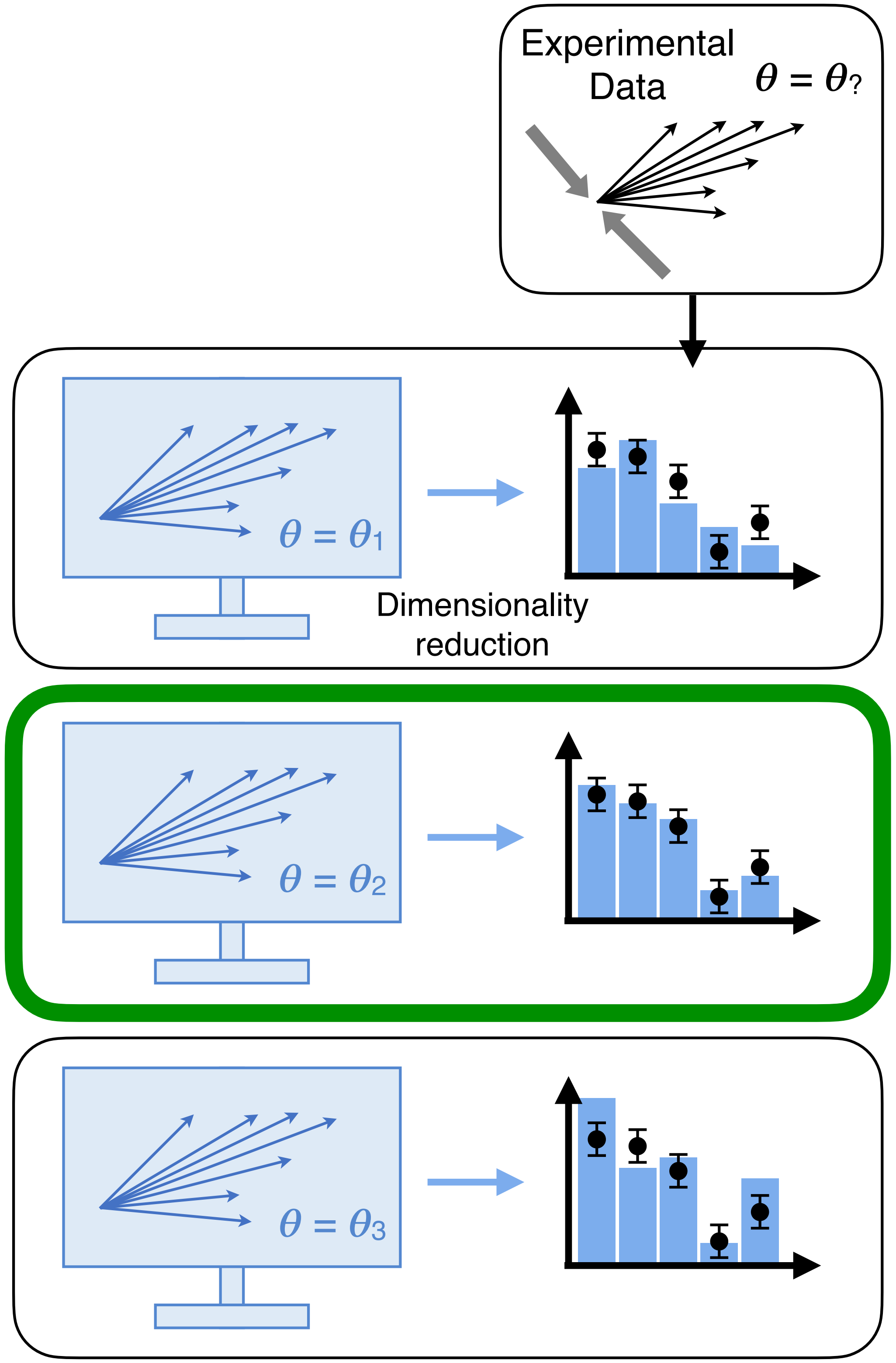
2208.02274

Why event moving?

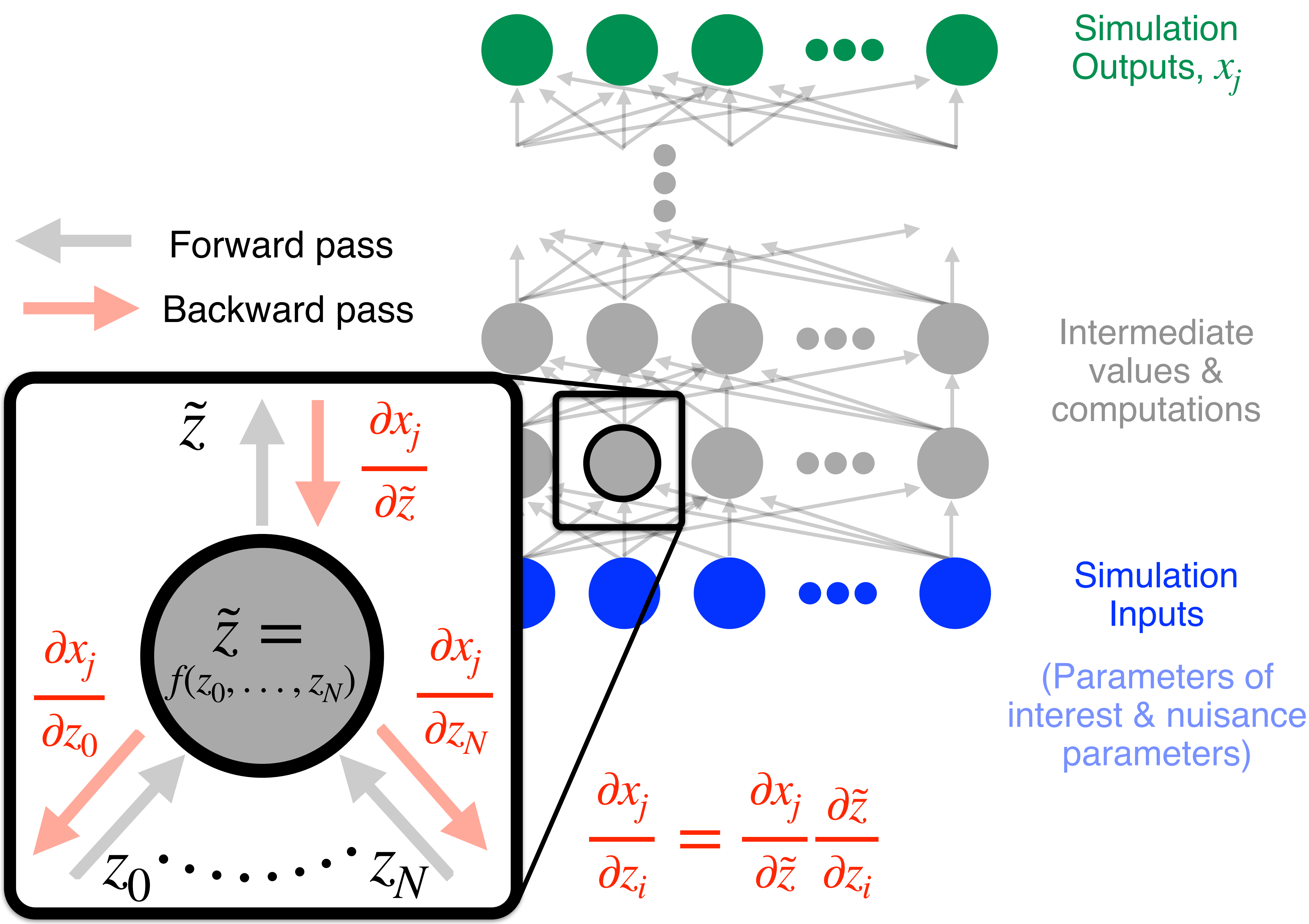
Often, we generate many simulations with different parameters (templates) and fit them to data.

We also often have to use histograms in order to interpolate.

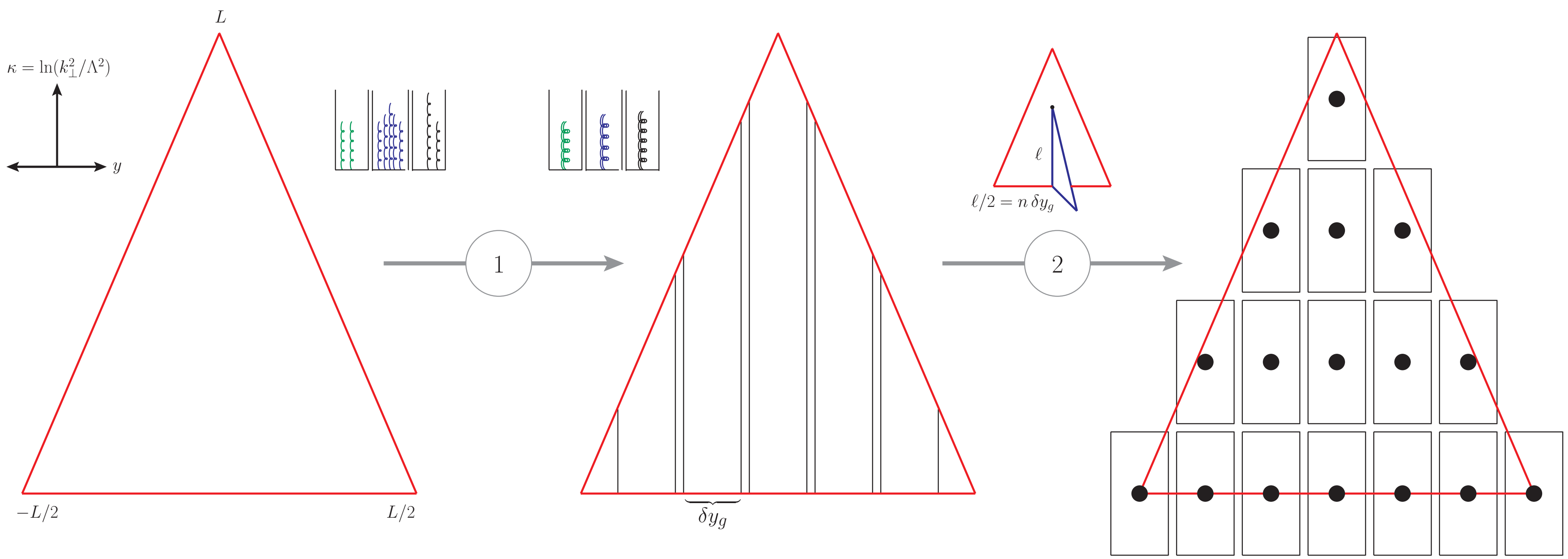
With event moving, we can interpolate in many dimensions and eliminate MC stat. uncertainties!

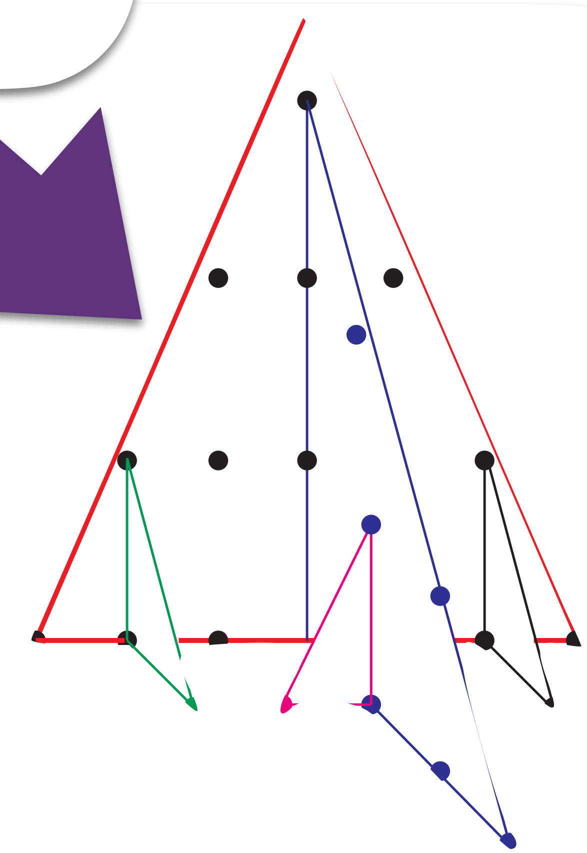


A brief word on Autodiff



Towards a differential parton shower

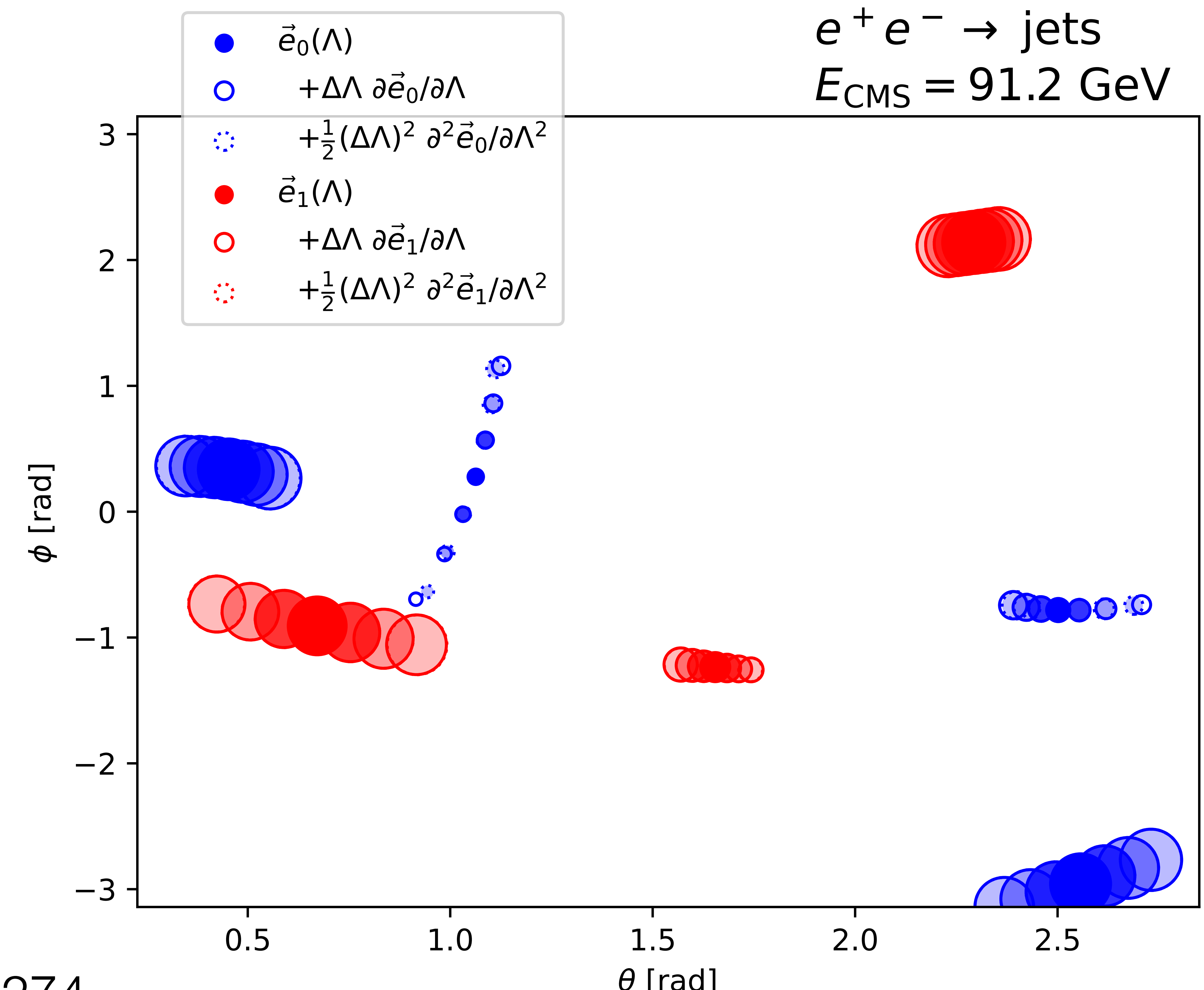




Full parton shower is a bit tricky since variable (unbounded) number of random numbers. Let's start with "Discrete QCD" where the number is fixed.

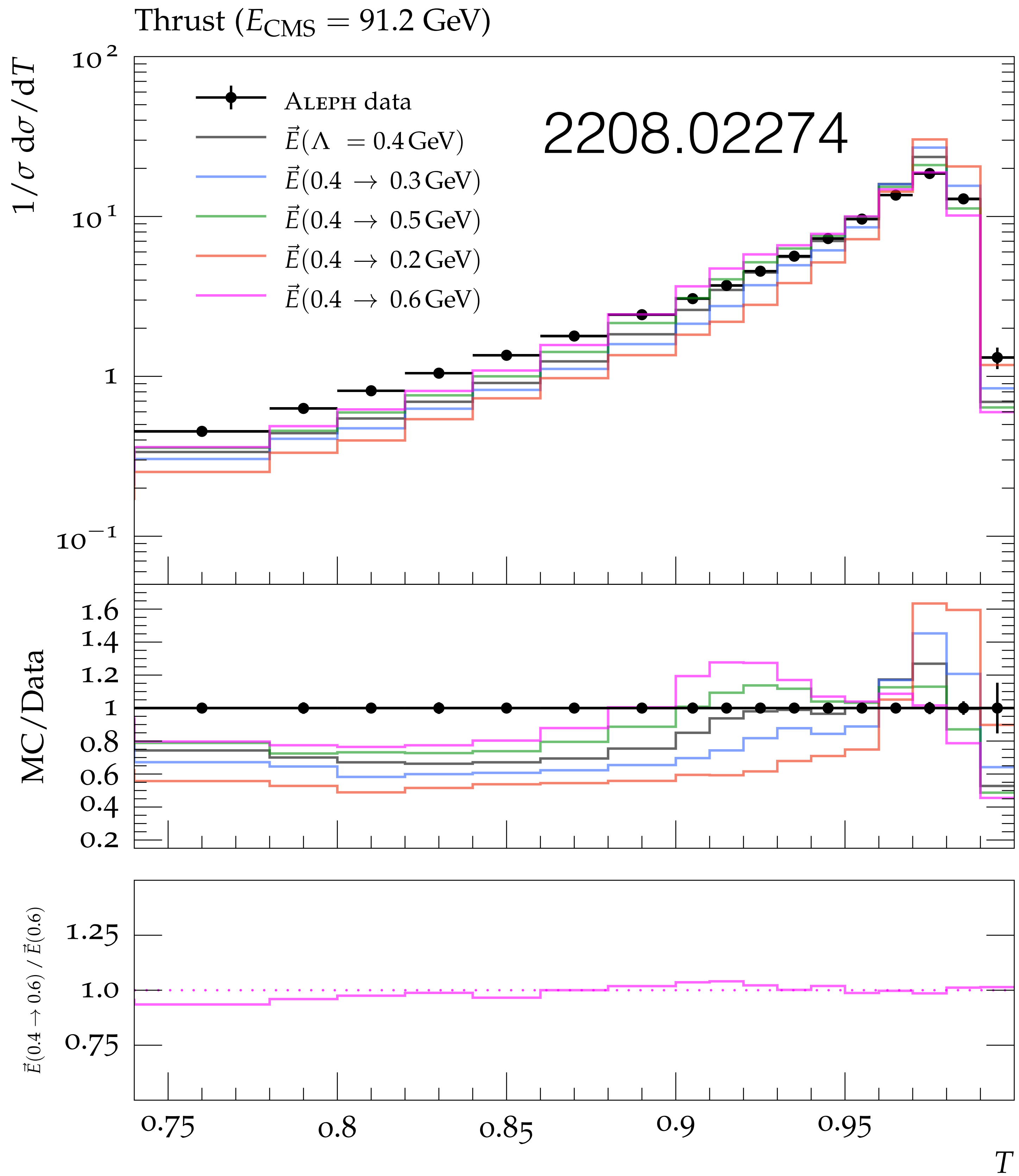
Images from 2207.10694; algorithm from Nuclear Physics B 463 (1996) 217

Towards a differential parton shower



2208.02274

Towards a differential parton shower

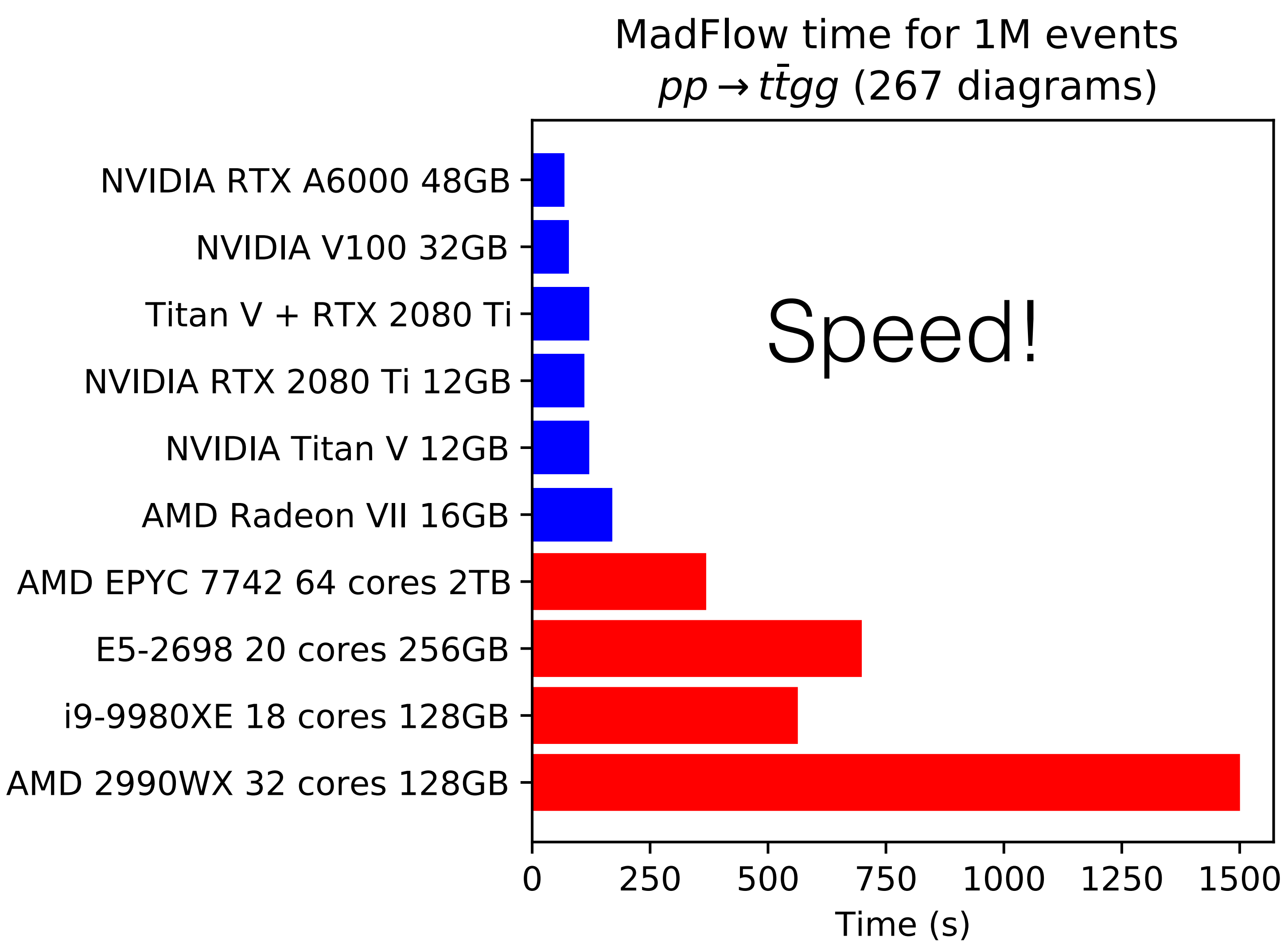


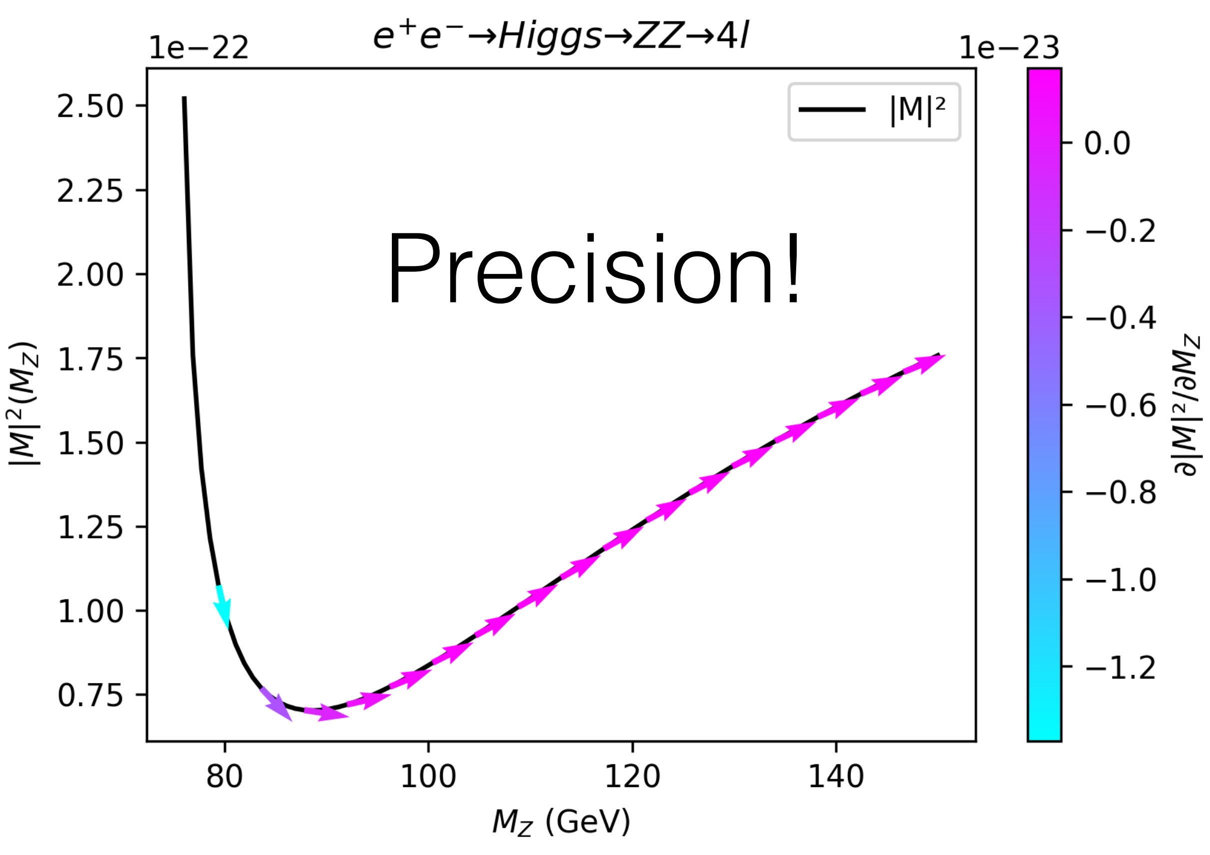
As a first test, we show how this can be used to extract the strong coupling constant.

All of these samples have the same random numbers!

Other Examples

(no surrogates - see my talk from yesterday)



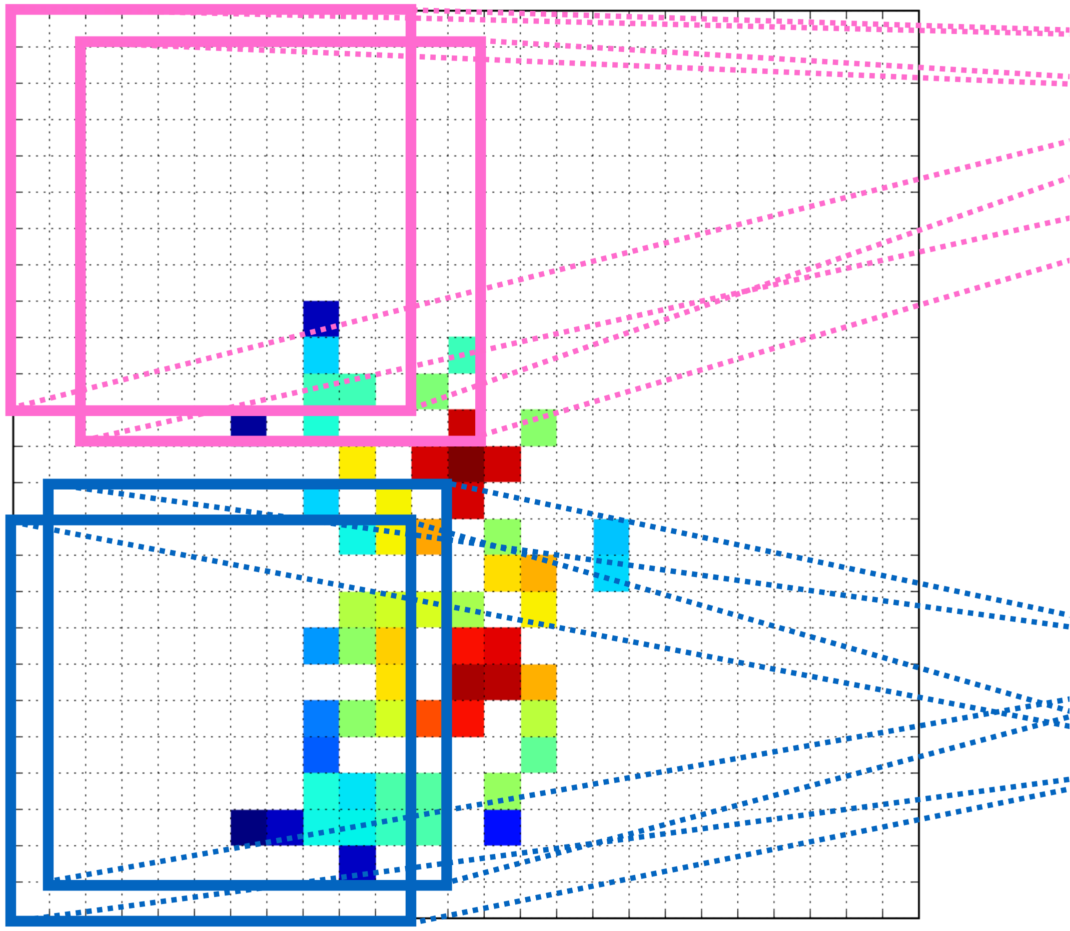


MadFlow 2106.10279, MadJax 2203.00057

Conclusions

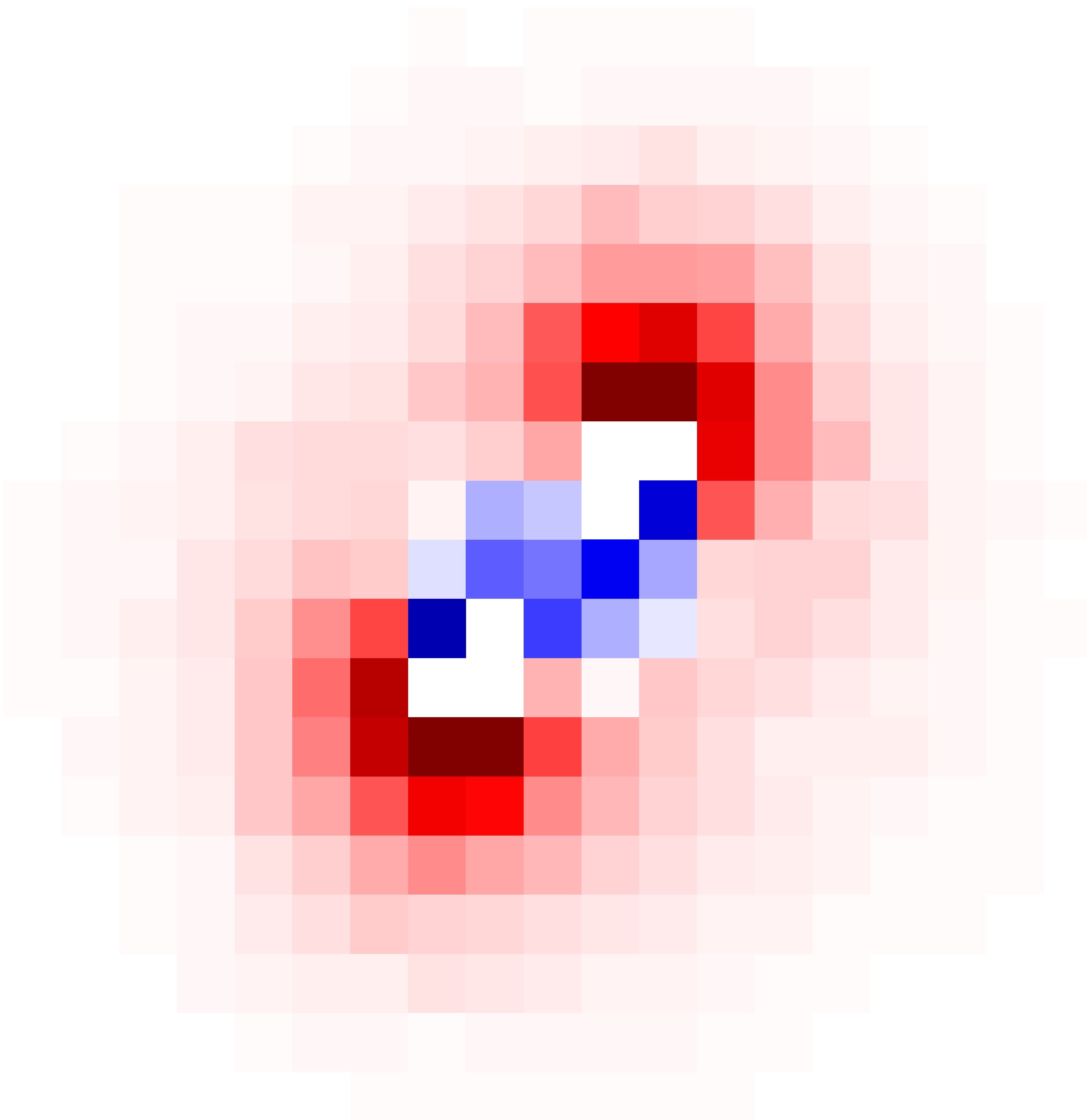
Thinking of our simulations as differentiable is a new and powerful paradigm.

We can do optimal inference and run our codes on accelerators "for free".



How far can we push this? Particle-level simulators, detector-level simulators? Analysis code? [see LR]

"inference-aware learning"



Fin.