Deep Learning for Collider Physics Simulation

Jesse Thaler

Deep Learning for Simulation, ICLR 2021 Workshop — May 7, 2021

The NSF AI Institute for Artificial Intelligence and Fundamental Interactions (IAIFI) *"eye-phi"*

Advance physics knowledge — from the smallest building blocks of nature to the largest structures in the universe — and galvanize AI research innovation

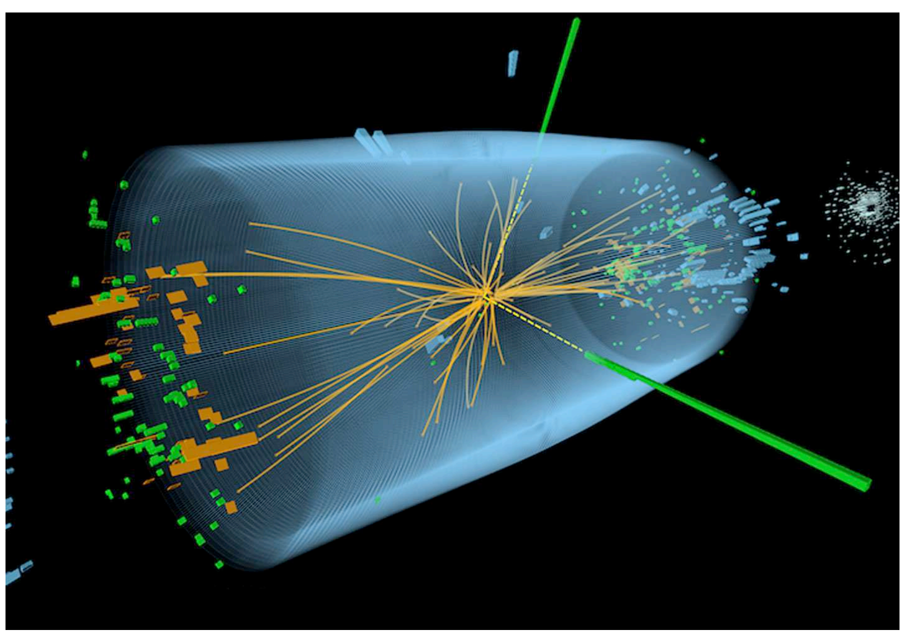
[http://iaifi.org, MIT News Announcement]

Simulation in Collider Physics?

SimDL: "We define simulation as the process of iteratively generating output of the next time step using the output of the previous time step as input starting from an initial condition."

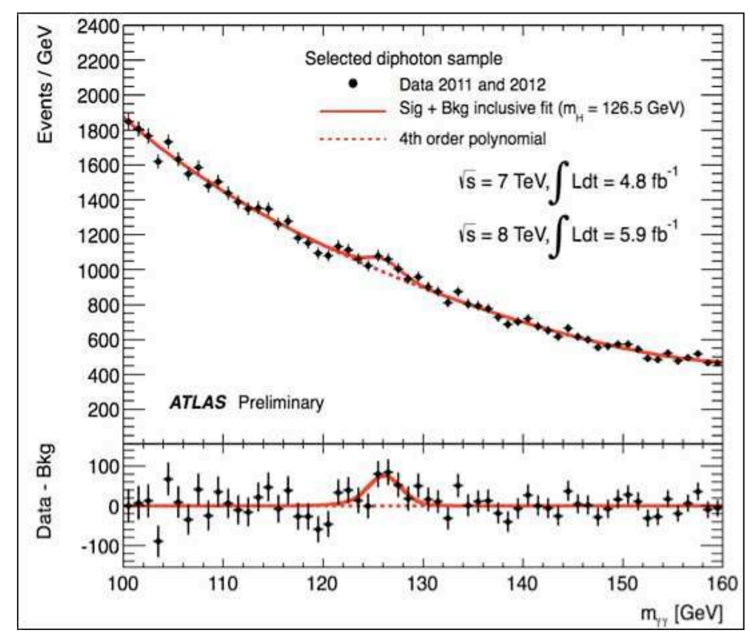
Does collider physics fit into this paradigm?

Yes, for simulating one collision event



(not my area of expertise)

No, when considering an ensemble of collisions

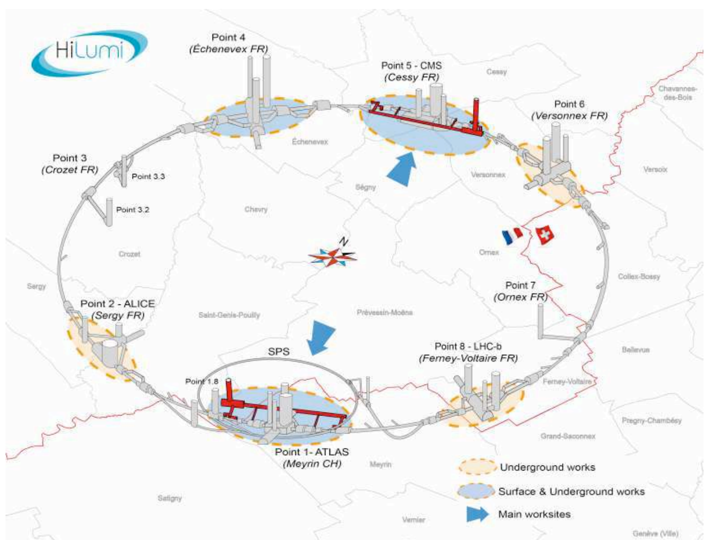


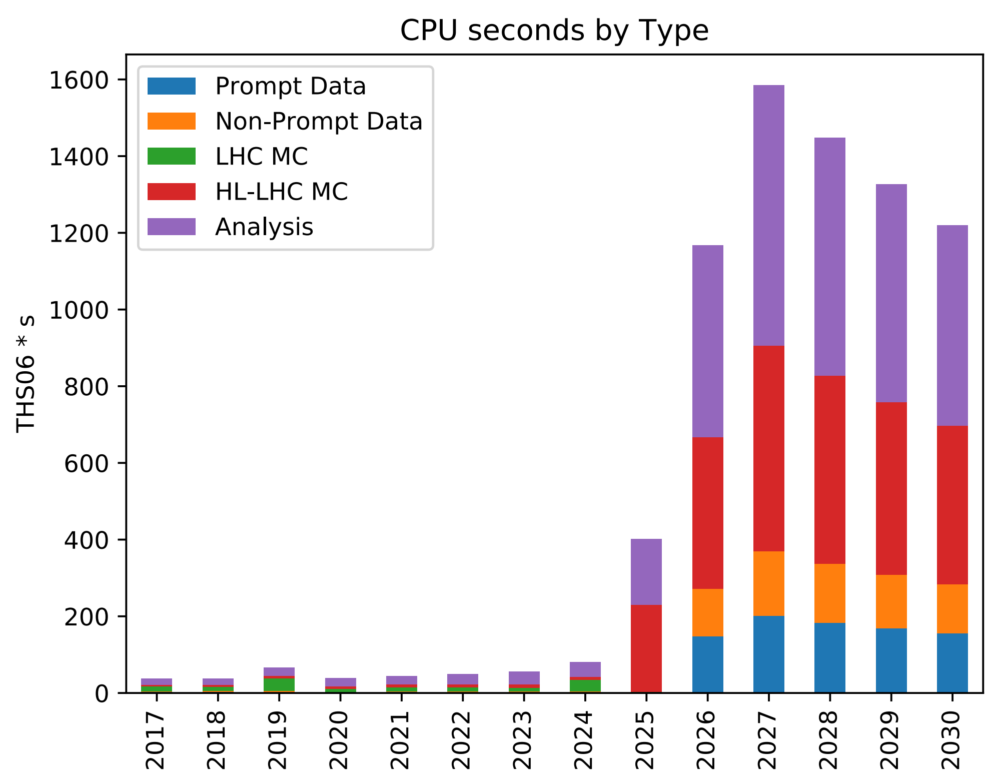
(key focus of my machine learning research)

Challenges for Simulating One Collision

(again & again & again & again & again & …)

Ballooning Computational Cost for the High Luminosity Large Hadron Collider





Analysis

Simulation

Reconstruction

Plus: Few intermediate checkpoints, heterogeneous software tools, inefficient computational workflows, supporting career development…

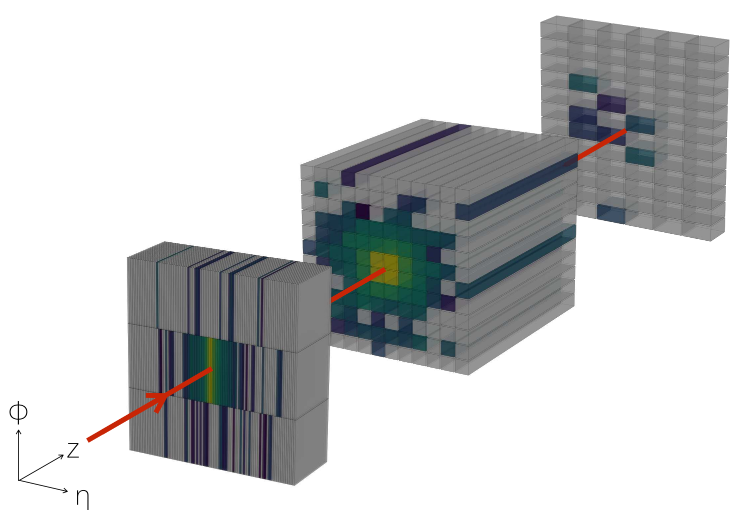
[HEP Software Foundation, CSBS 2019, HSF Physics Event Generator Working Group, arXiv 2020]

Challenges for Simulating One Collision

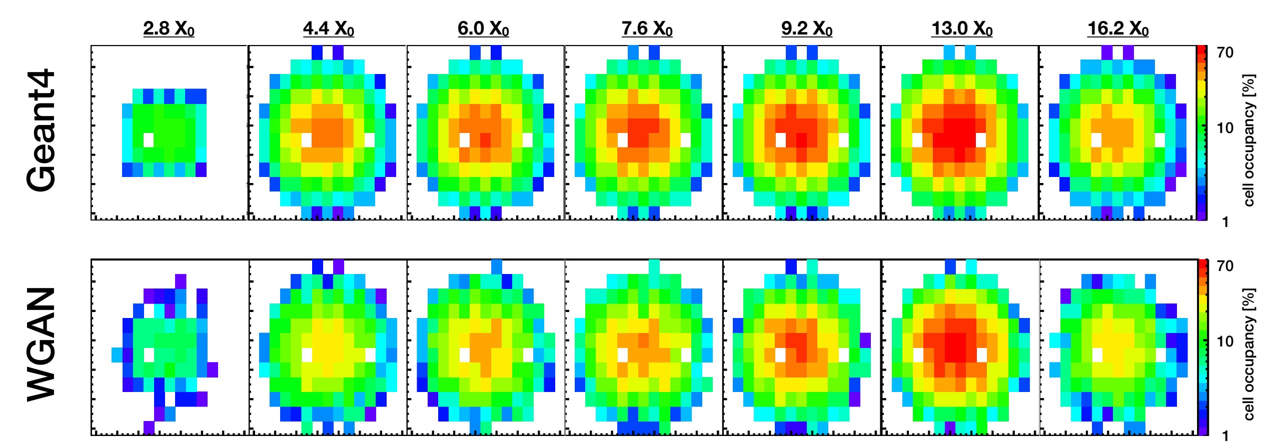
(again & again & again & again & again & …)

For Another Talk: Accelerating Simulation with Deep Learning

e.g. Electromagnetic Calorimetry with Adversarial Networks



[Paganini, de Oliveira, Nachman, PRD 2018]



[Erdmann, Glombitza, Quast, CSBS 2019]

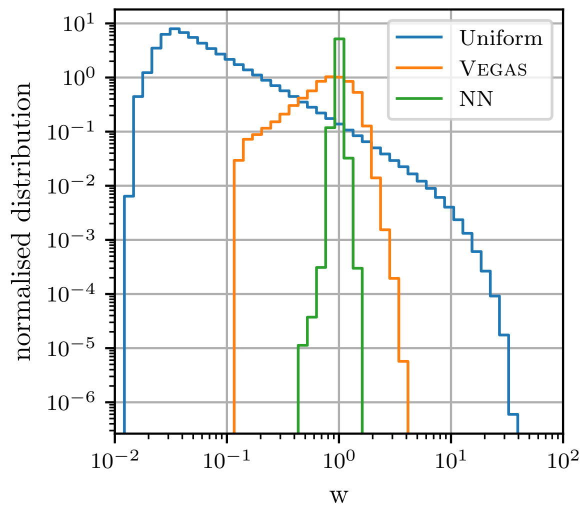
[for complete bibliography, see HEP-ML Living Review]

Plus: Few intermediate checkpoints, heterogeneous software tools, inefficient computational workflows, supporting career development…

[HEP Software Foundation, CSBS 2019, HSF Physics Event Generator Working Group, arXiv 2020]

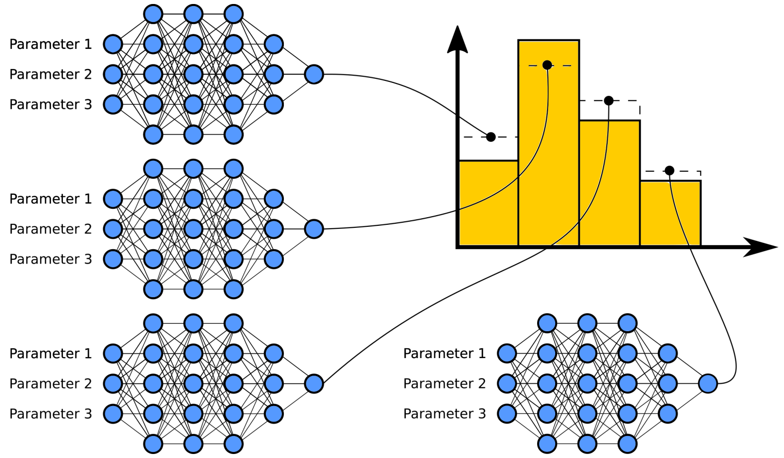
Challenges for Simulating Collision Ensembles

Phase Space Sampling



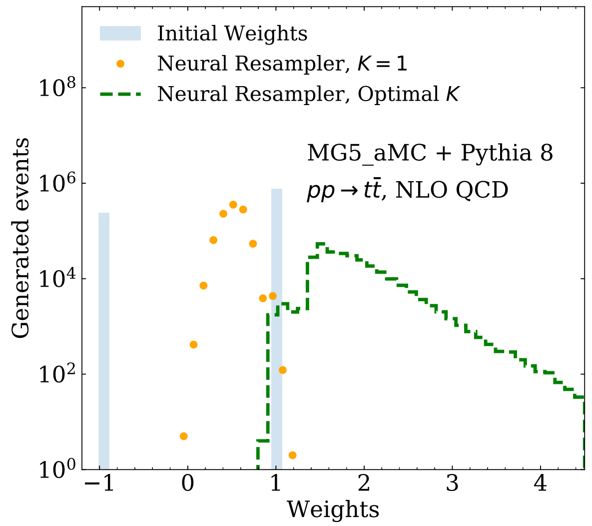
[Bothmann, Janßen, Knobbe, Schmale, Schumann, SciPost 2020]

Parameter Tuning



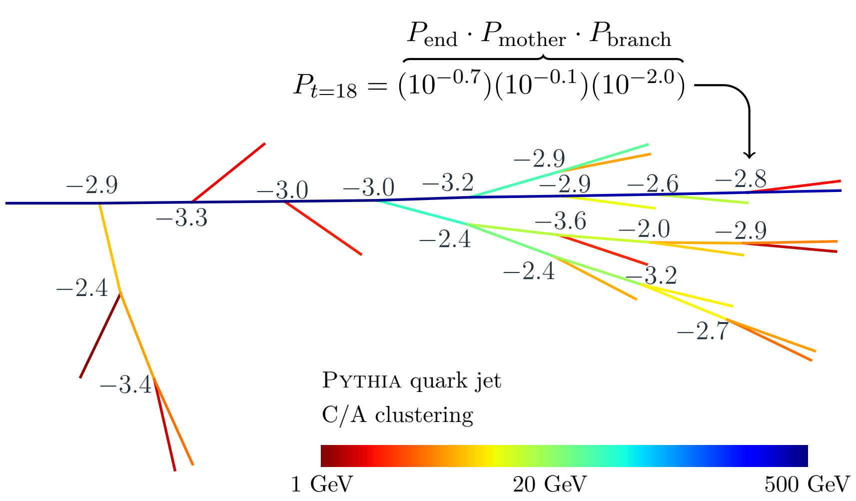
[Lazzarin, Alioli, Carrazza, CPC 2021]

Negative Weights



[Nachman, JDT, PRD 2020]

Interpretability



[Andreassen, Feige, Frye, Schwartz, EPJC 2019]

[for complete bibliography, see HEP-ML Living Review]

Challenges for Simulating Collision Ensembles

Key take away from this talk:

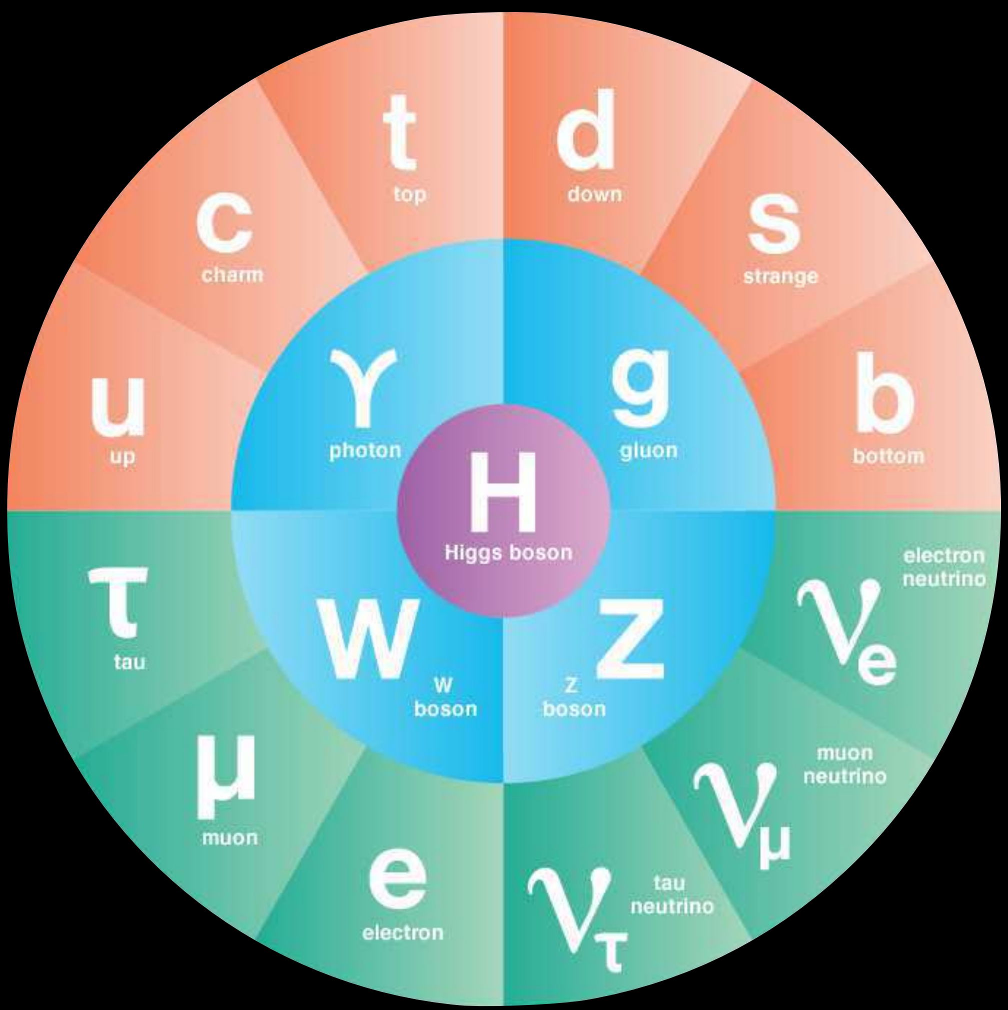
Collider physics deals with large sets of independent and identically distributed data

Here, simulation tasks (generative modeling) often map to classification tasks (inference)

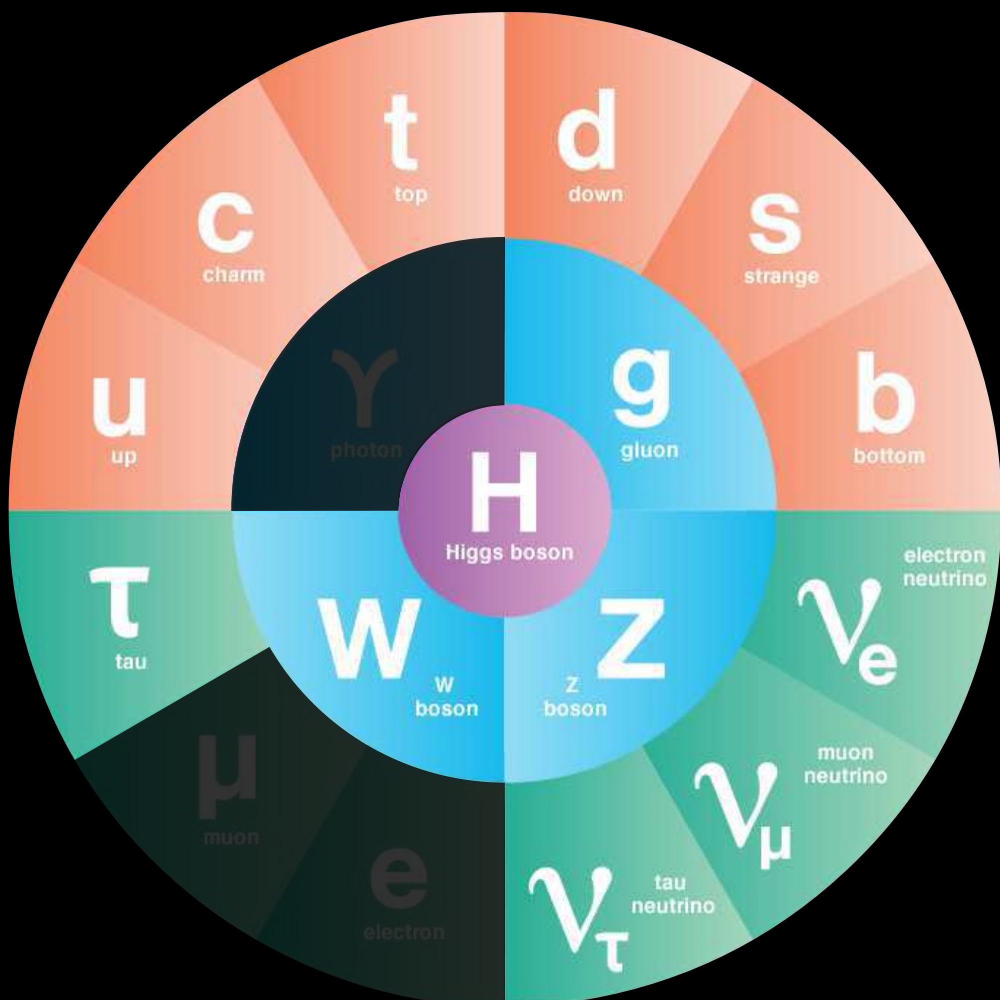
Deep learning excels at classification

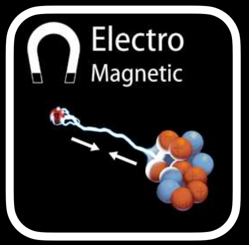
[for complete bibliography, see HEP-ML Living Review]

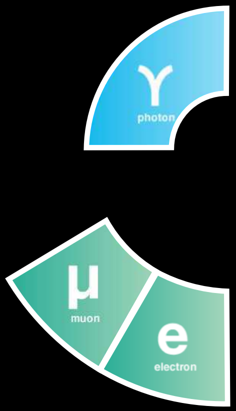
Particle Physics 101



Particle Physics 101







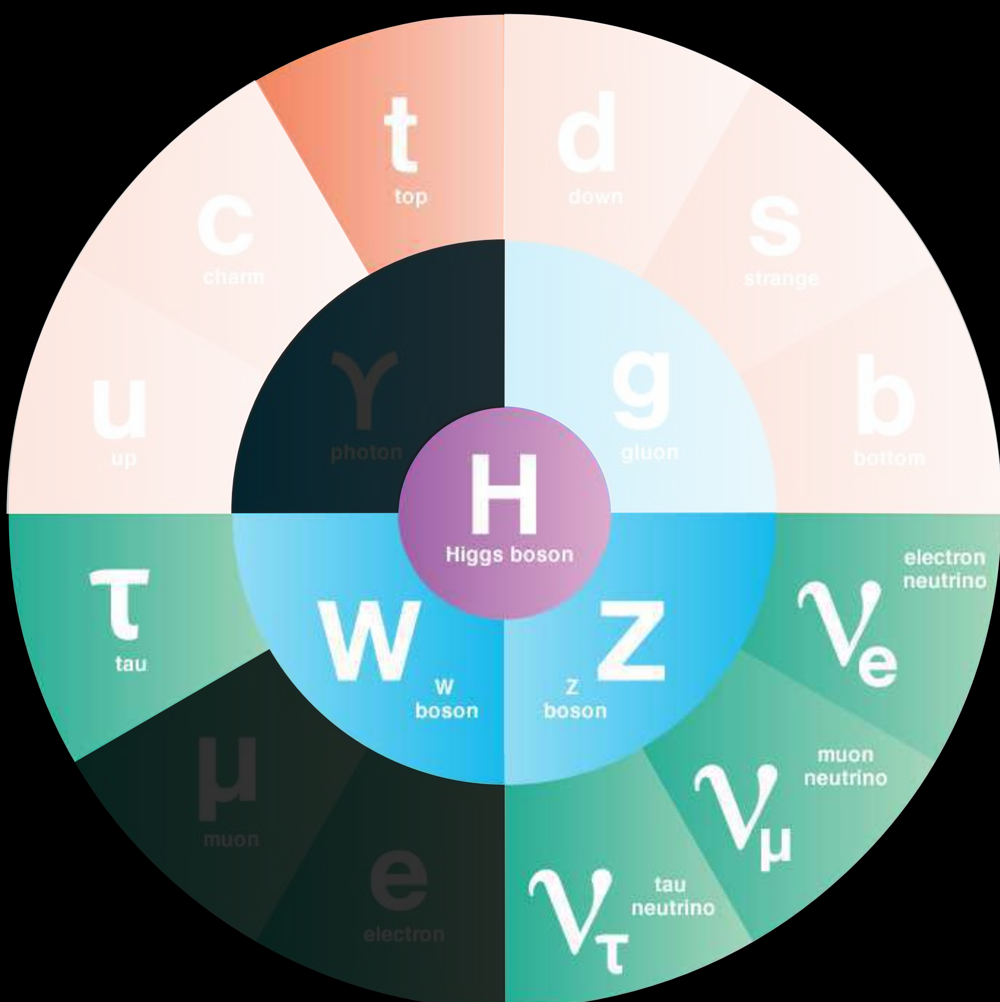
γ photon

$e^{\pm}$ electron

$\mu^{\pm}$ muon

elementary

Particle Physics 101



QCD Confinement

Quarks
&
Gluons

elementary

γ photon

$e^{\pm}$ electron

$\mu^{\pm}$ muon

composite

$\pi^{\pm}$ pion

$K^{\pm}$ kaon

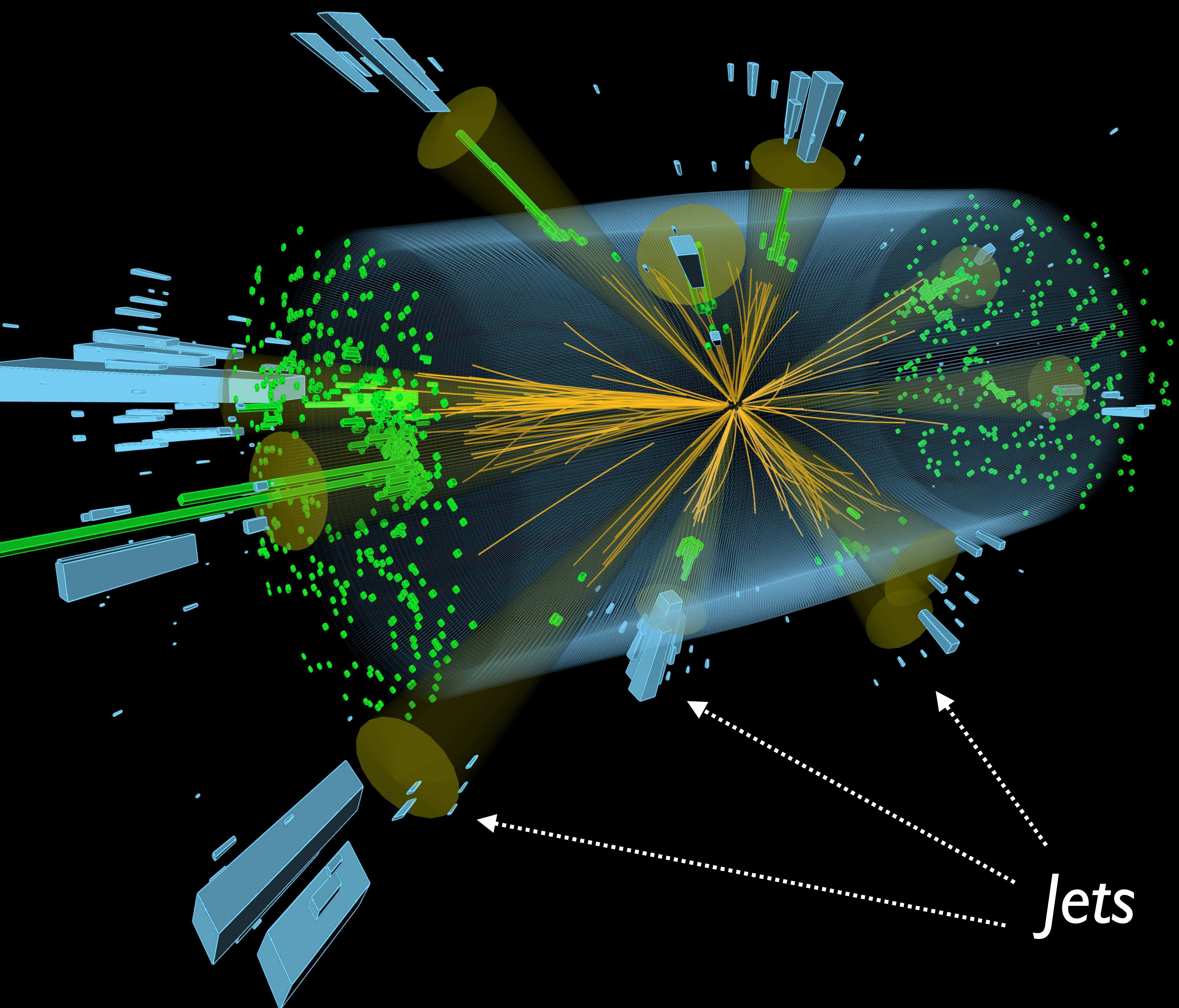
K_L^0 K-long

$p/\overline{p}$ proton

$n/\overline{n}$ neutron

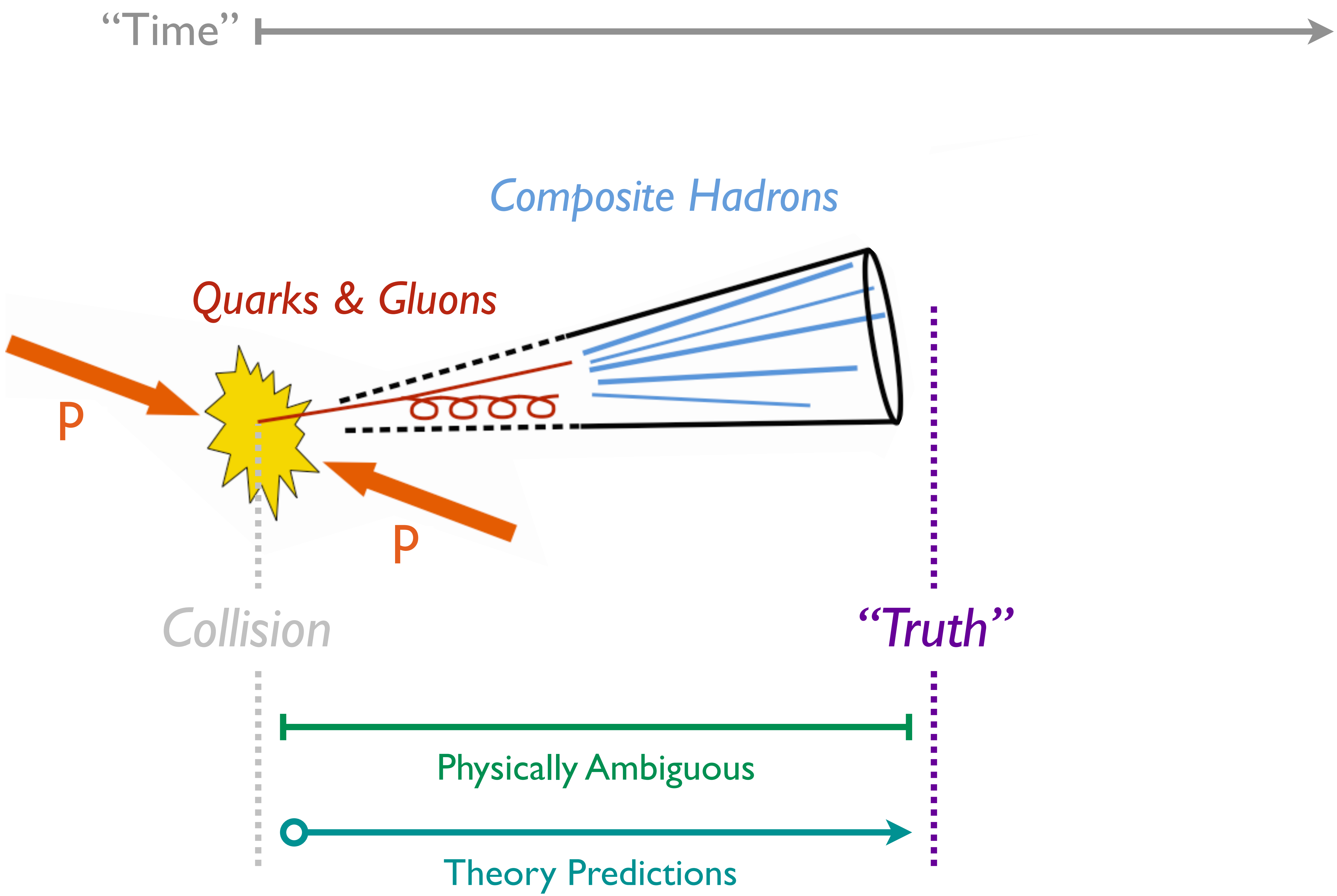
Collider Event

Every 25 nanoseconds at the LHC

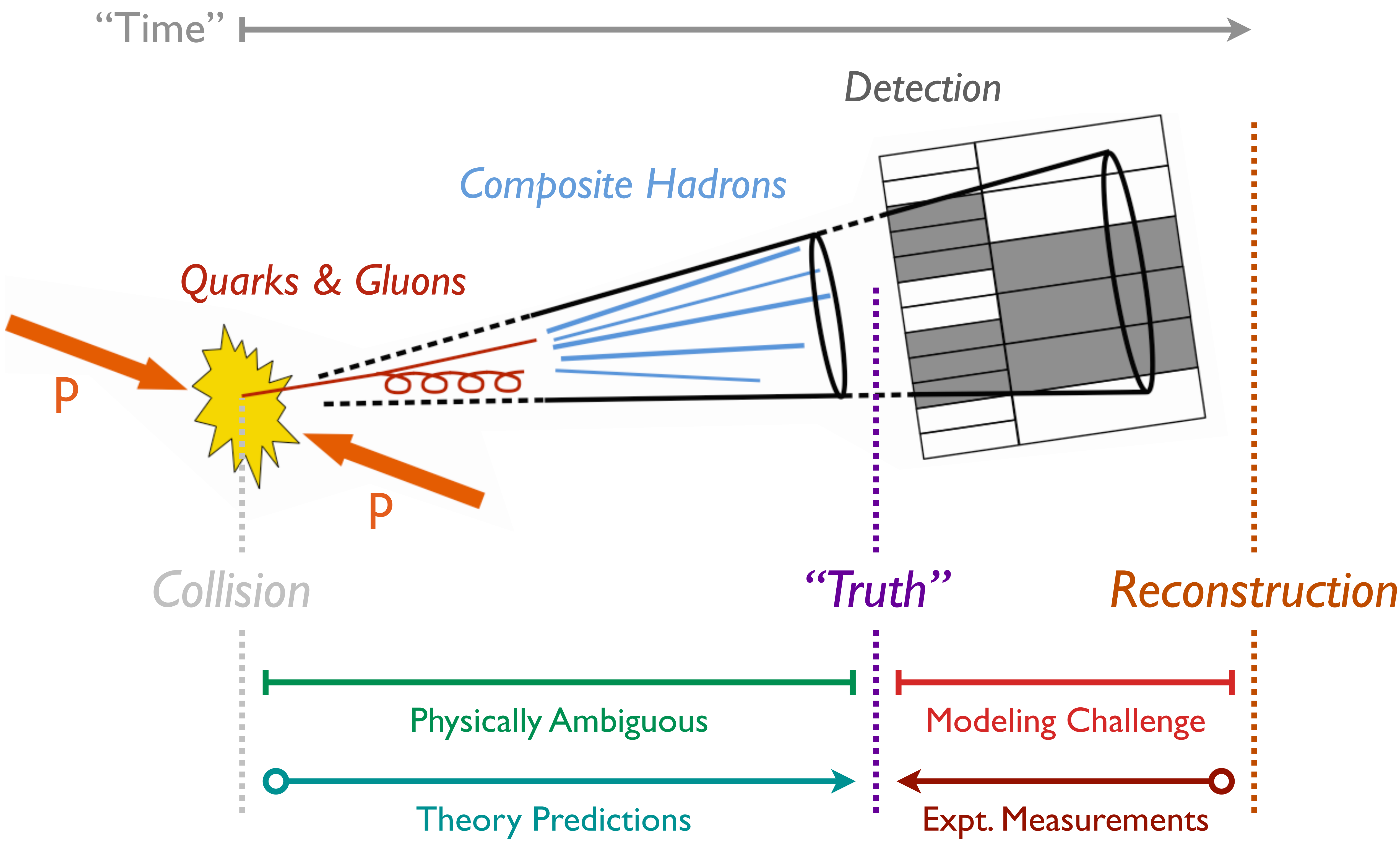


	T	E	H	M		
elementary		●			γ	photon
	●	●			$e^\pm$	electron
	●	●	●	●	$\mu^\pm$	muon
composite	●	●	●		$\pi^\pm$	pion
	●	●	●		$K^\pm$	kaon
		●	●		K_L^0	K-long
	●	●	●		$p/\overline{p}$	proton
		●	●		$n/\overline{n}$	neutron

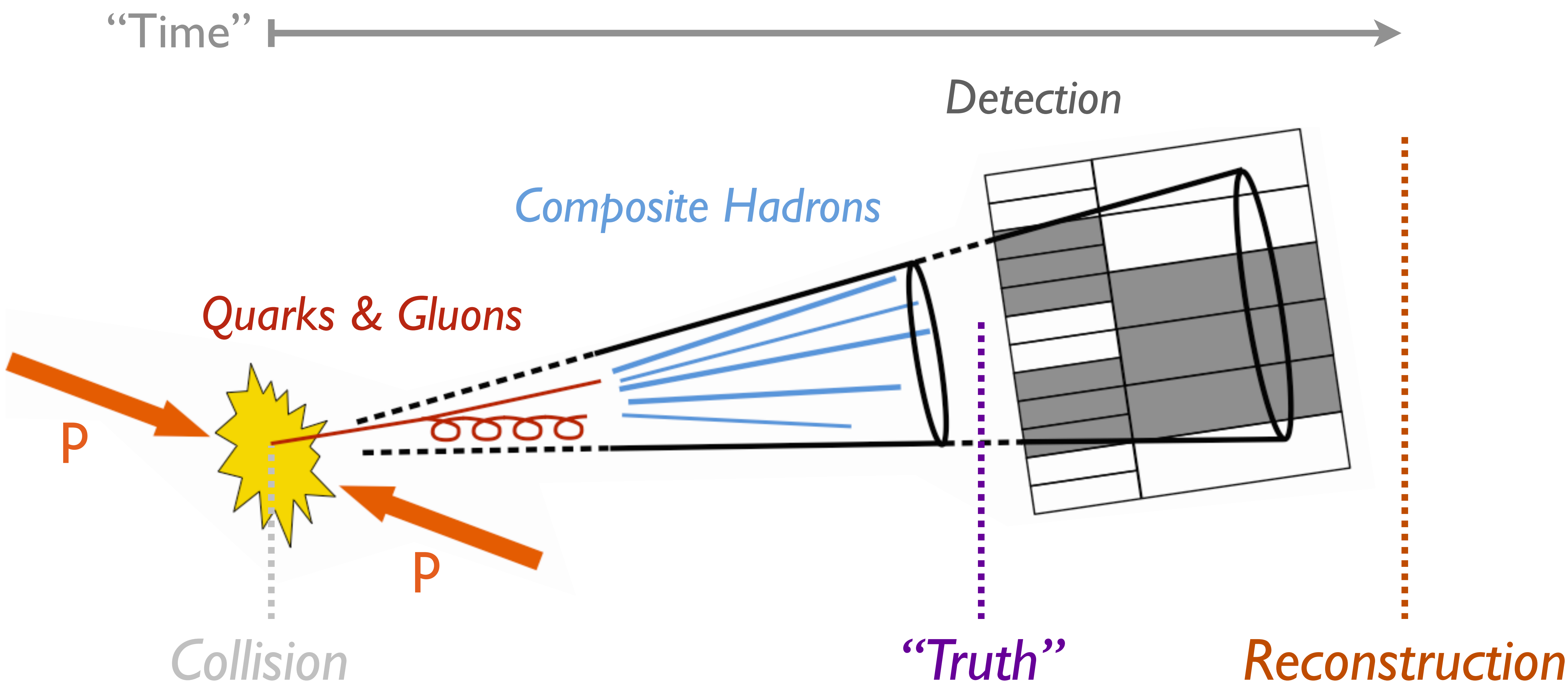
Simulating LHC Collisions



Simulating LHC Collisions



Simulating LHC Collisions



“Event Generation”

MadGraph5_aMC@NLO, POWHEG BOX, SHERPA, HERWIG, PYTHIA, …

“Detector Simulation”

GEANT4, FLUKA, …

Matrix Element | Parton Shower | Hadronization

Simulating LHC Collisions

Key: Collider data is i.i.d. to excellent approximation

Independent & Identical: $p(\{x_1, x_2, \ldots, x_N\}) = \underbrace{\prod_{i=1}^{N}} p(x_i)$

Simulation = "Generation" + "Simulation": $\int dt \, \underline{p_{\text{Gen}}(t)} \, \underline{p_{\text{Sim}}(r|t)} = p(r)$

Collision | *"Truth"* | *Reconstruction*

"Event Generation"

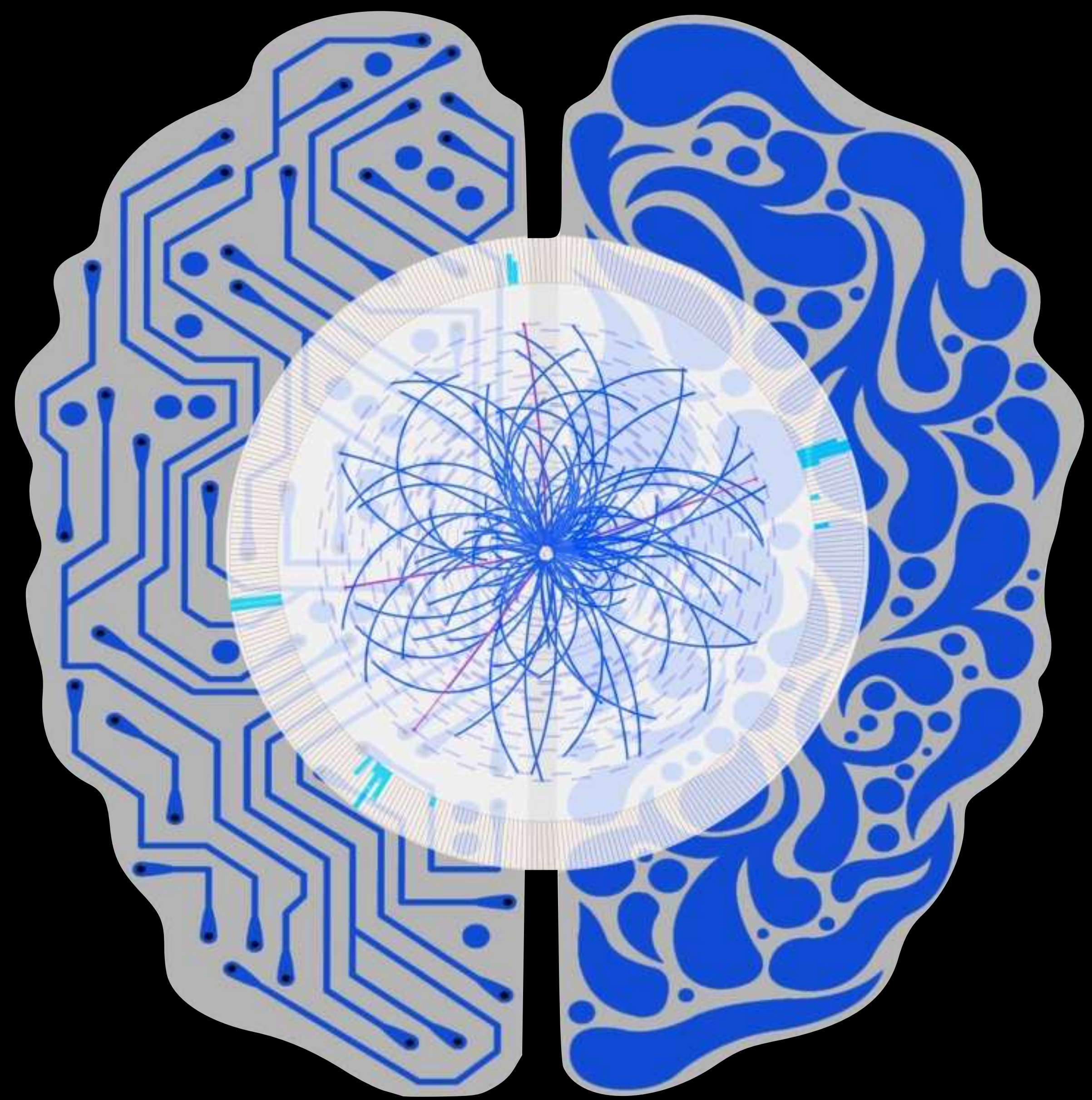
MadGraph5_aMC@NLO, POWHEG BOX, SHERPA, HERWIG, PYTHIA, …

"Detector Simulation"

GEANT4, FLUKA, …

Matrix Element | Parton Shower | Hadronization

Deep Learning for Inference



When can simulation be cast as classification?

Likelihood Ratio Trick

Many collider physics problems can be expressed in this form!

Key example of simulation-based inference

Goal: Estimate p(x) / q(x)

Training Data: Finite samples P and Q

Learnable Function: f(x) parametrized by, e.g., neural networks

Loss Function(al): $$L = -\langle \log f(x) \rangle_P + \langle f(x) - 1 \rangle_Q$$

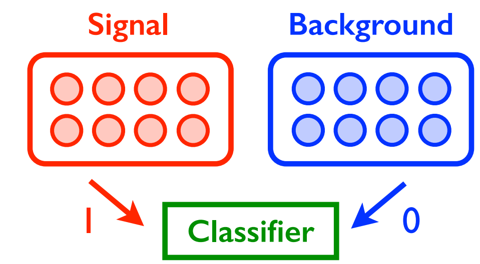
Asymptotically: $$\arg\min_{f(x)} L = \frac{p(x)}{q(x)}$$ *Likelihood ratio*

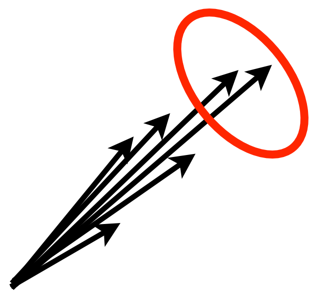
$$-\min_{f(x)} L = \int \mathrm{d}x \, p(x) \log \frac{p(x)}{q(x)}$$ *Kullback–Leibler divergence*

[see e.g. D'Agnolo, Wulzer, PRD 2019; simulation-based inference in Cranmer, Brehmer, Louppe, PNAS 2020; relation to f-divergences in Nguyen, Wainwright, Jordan, AoS 2009; Nachman, JDT, arXiv 2021]

Quark/Gluon Classification

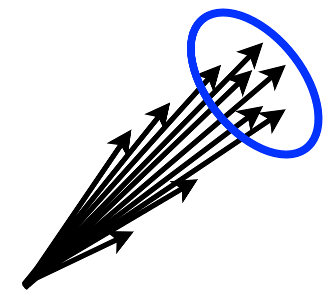
"Hello, World!" of Jet Physics



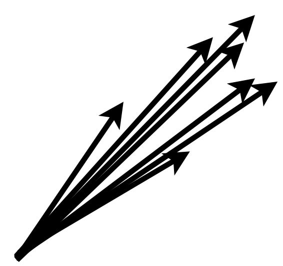


Quark
$C_q = 4/3$

vs.



Gluon
$C_g = 3 = 9/3$

Find h () such that

$$h(\textbf{Quark}) = 1$$

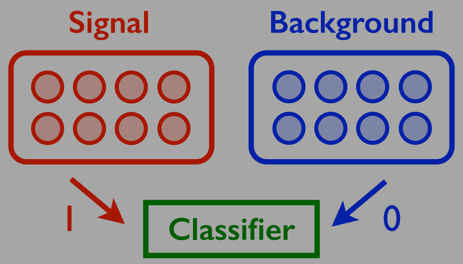
$$h(\textbf{Gluon}) = 0$$

Best you can do: $h(\mathcal{J}) = \left(1 + \dfrac{p(\mathcal{J}|\mathrm{G})}{p(\mathcal{J}|\mathrm{Q})}\right)^{-1}$

(Neyman-Pearson lemma)

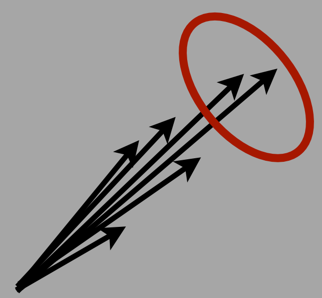
Likelihood ratio yields optimal binary classifier (and vice versa)

[see e.g. Gras, Höche, Kar, Larkoski, Lönnblad, Plätzer, Siódmok, Skands, Soyez, JDT, JHEP 2017;
Komiske, Metodiev, Schwartz, JHEP 2017; Komiske, Metodiev, JDT, JHEP 2018]



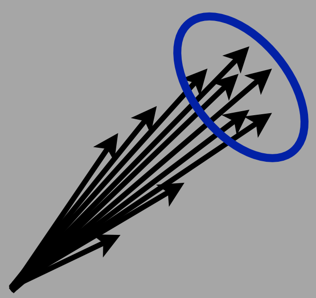
Quark/Gluon Classification

"Hello, World!" of Jet Physics



Quark
$C_q = 4/3$

vs.



Gluon
$C_g = 3 = 9/3$

This is all well and good,
but what does binary classification
have to do with simulation?

Likelihood ratio yields optimal binary classifier (and vice versa)

[see e.g. Gras, Höche, Kar, Larkoski, Lönnblad, Plätzer, Siódmok, Skands, Soyez, JDT, JHEP 2017;
Komiske, Metodiev, Schwartz, JHEP 2017; Komiske, Metodiev, JDT, JHEP 2018]

From Tautology to Essential Tool

Remember that collider data is i.i.d.

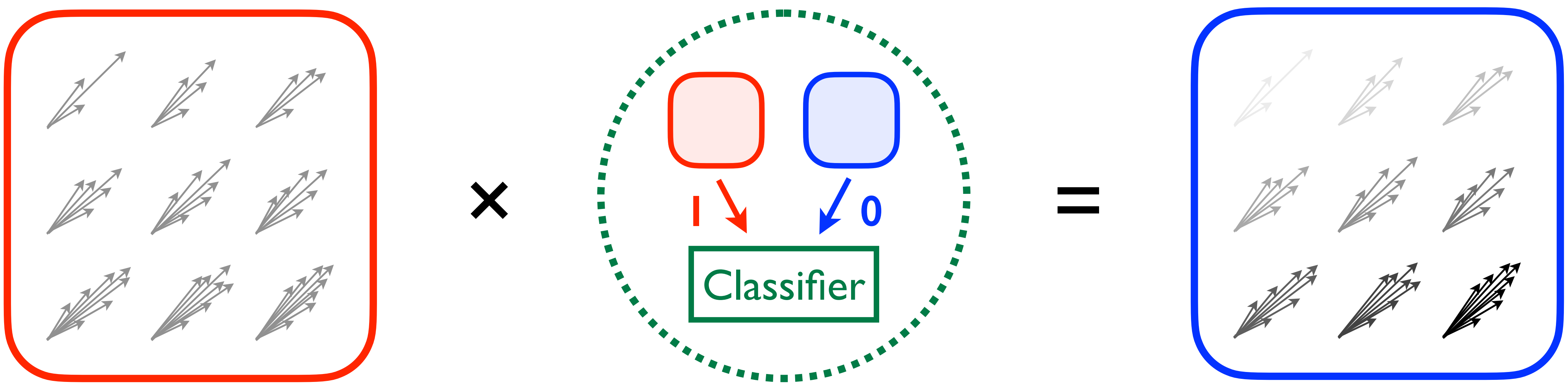
[see i.i.d. discussion in Nachman, JDT, arXiv 2021]

$$q(x) \times \frac{p(x)}{q(x)} = p(x)$$

Generate samples according to Q

Weight each sample by likelihood ratio

Obtain weighted samples distributed according to P

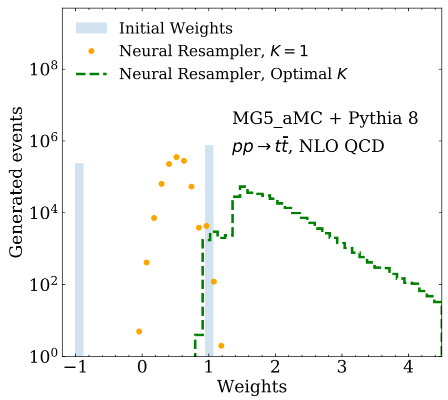


With large enough data samples,
binary classification yield weighted simulation tool

From Curmudgeon to Evangelist

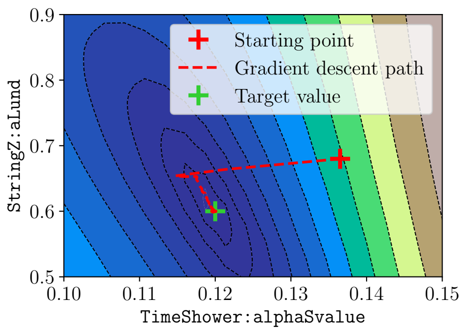
What simulation problems can be solved with deep learning?

Three Collider Simulation Examples



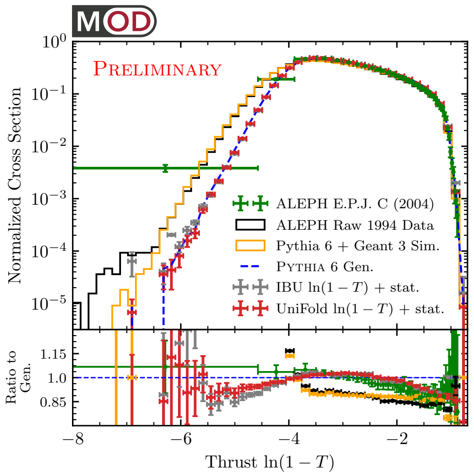
Removing Negative Weights

Leverages deep learning as function learning



Interpolating Simulation Parameters

Leverages robust generalizability



Deconvolving Detector Effects

Leverages ability to handle high-dimensional inputs

The Challenge of Negative Weights

“Event Generation”		
MadGraph5_aMC@NLO, POWHEG BOX, SHERPA, HERWIG, PYTHIA, …		
Matrix Element	Parton Shower	Hadronization

How do you interpolate between different calculational schemes?

$$p_{\text{improved}}(x) = p_{\text{baseline}}(x) + p_{\text{correction}}(x)$$

Positive (if physics is correct)

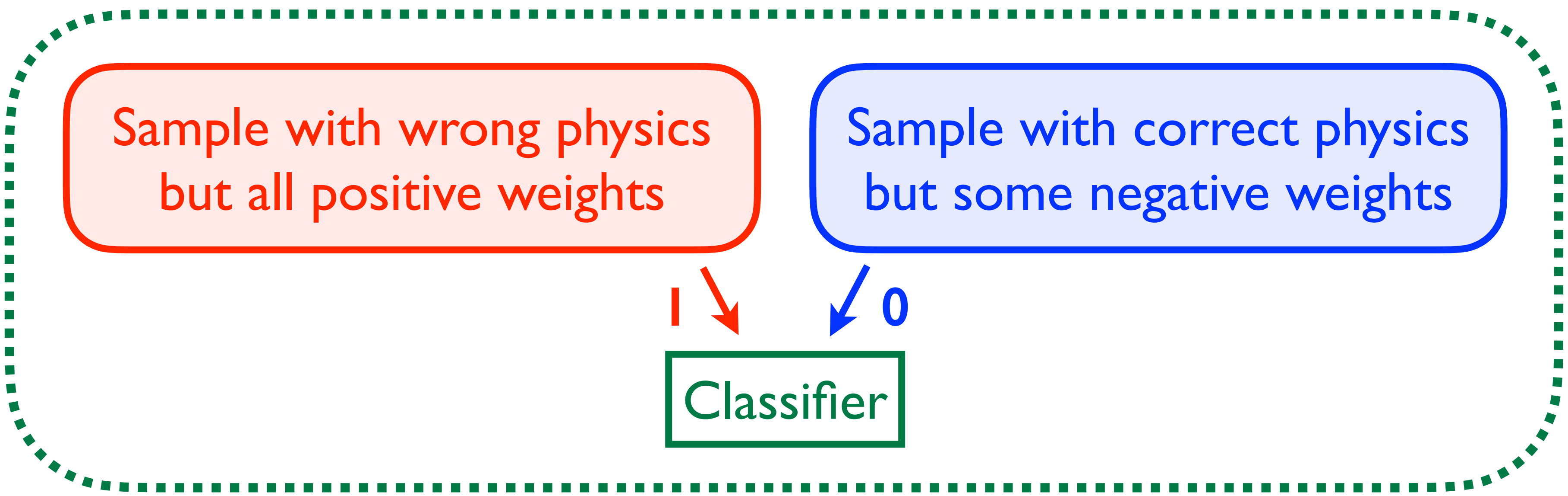
Positive by construction

Could be positive or negative!

Often need negative weight events to describe correct physics, leading to downstream computational inefficiencies

[see discussion in Andersen, Gütschow, Maier, Prestel, EPJC 2020]

Neural Resampling



$$\Rightarrow w(x)$$

Plain reweighting yields all positive weights with correct asymptotic probability density

$$p_{\text{wrong}}(x) \times w(x) = p_{\text{correct}}(x)$$

Improved resampling through auxiliary neural network yields correct statistical uncertainties

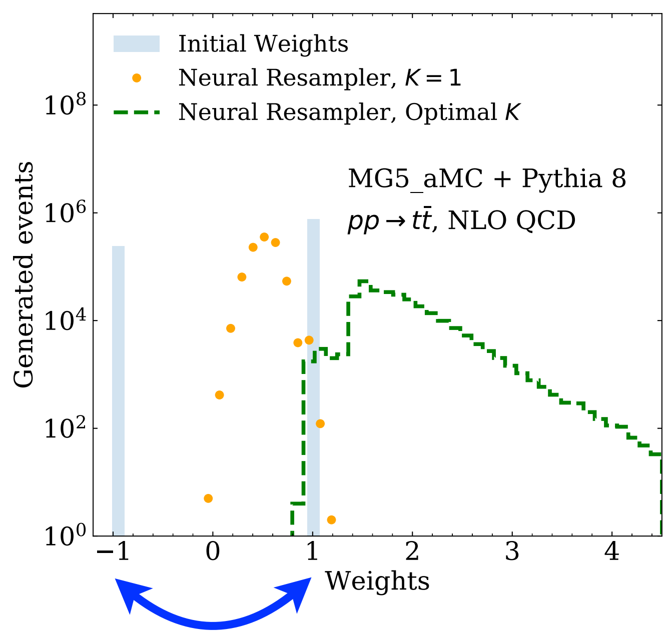
$$\left(\frac{\delta p}{p}\right)^2 = \frac{\langle w^2 \rangle}{\langle w \rangle^2}$$

[Nachman, JDT, PRD 2020]

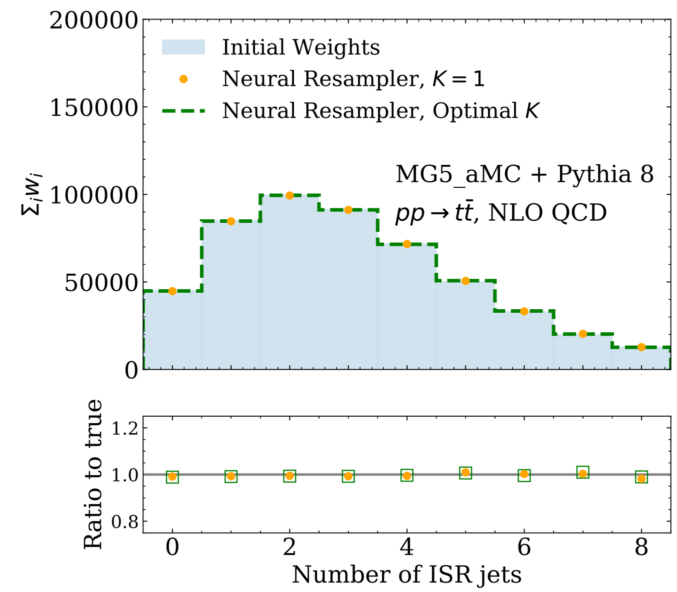
Neural Resampling

Case Study in Jet Physics at Large Hadron Collider

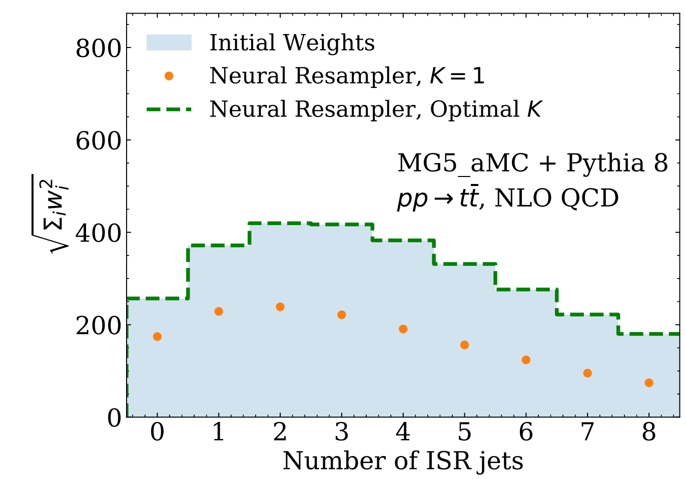
Original sample: large weight cancellations



Reweighting recovers desired distribution



Resampling recovers desired uncertainties



Improved resampling through auxiliary neural network yields correct statistical uncertainties

$$\left(\frac{\delta p}{p}\right)^2 = \frac{\langle w^2 \rangle}{\langle w \rangle^2}$$

[Nachman, JDT, PRD 2020]

The Challenge of Monte Carlo Tuning

"Event Generation"

MadGraph5_aMC@NLO, POWHEG BOX, SHERPA, HERWIG, PYTHIA, …

Matrix Element	Parton Shower	Hadronization

How do you account for modeling parameters that cannot be predicted from first principles?

$$\arg\min_{\theta} \text{Dist}\left[p_{\text{simulation}}(x|\theta) \,\middle\|\, p_{\text{validation data}}(x) \right]$$

O(10-100) parameter to optimize over

Daunting computational challenge to simulate all variants

Parametrized Classification for Simulation

Baseline sample with good phase space coverage

Samples from a range of parameter values θ

1 ↘ ↙ 0

Classifier parametrized by θ

$$\Rightarrow w(x|\theta)$$

With good generalizability, parametrized classification can interpolate parameters

$$p_{\text{baseline}}(x) \times w(x|\theta) = p_{\text{correct}}(x|\theta)$$

Or you can directly perform maximum likelihood analysis for parameter tuning

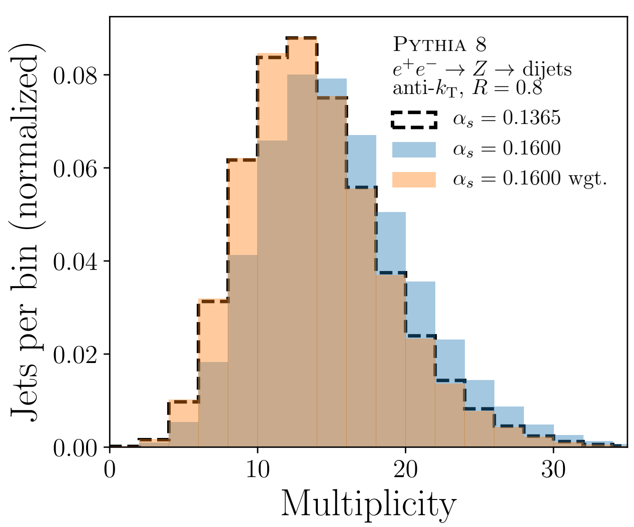
$$w(x|\theta) = \frac{p(x|\theta)}{p(x|\theta_0)}$$

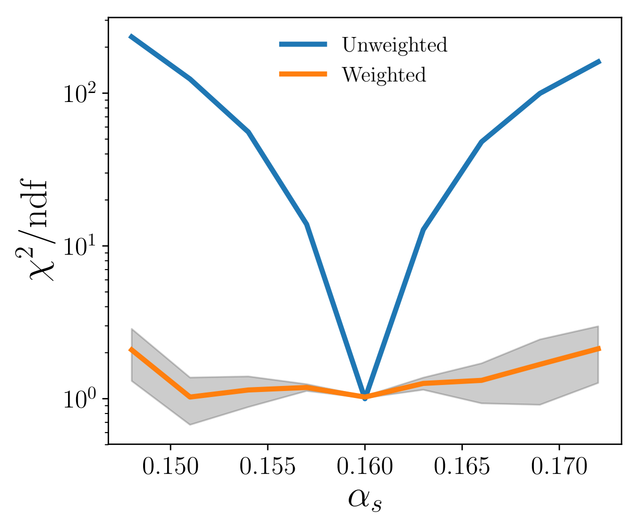
[Cranmer, Pavez, Louppe, arXiv 2015; Baldi, Cranmer, Faucett, Sadowski, Whiteson, EPJC 2016]

Parametrized Classification for Simulation

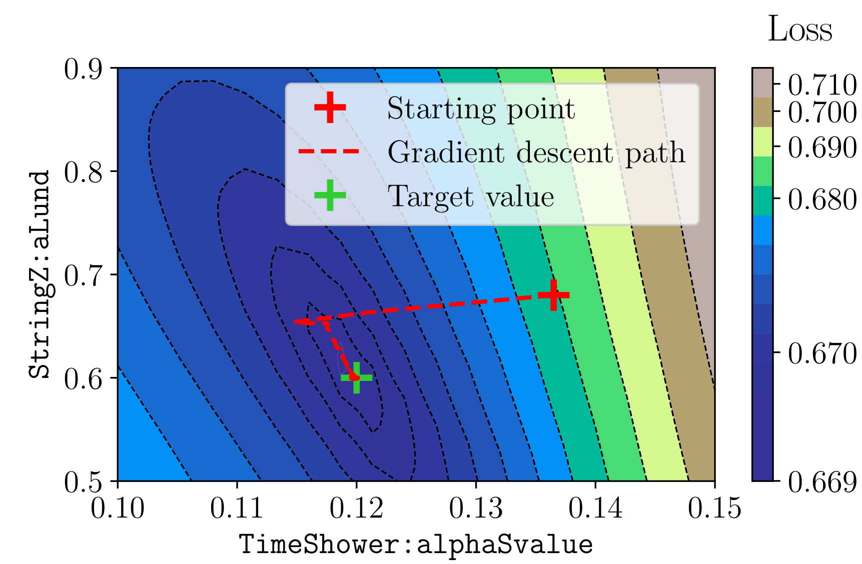
Case Study in Tuning Monte Carlo Event Generators

Reweighted simulation exhibits excellent fidelity with relatively low computational cost





Multidimensional maximum likelihood tuning



[Andreassen, Nachman, PRD 2020]

Or you can directly perform maximum likelihood analysis for parameter tuning

$$w(x|\theta) = \frac{p(x|\theta)}{p(x|\theta_0)}$$

[Cranmer, Pavez, Louppe, arXiv 2015; Baldi, Cranmer, Faucett, Sadowski, Whiteson, EPJC 2016]

The Challenge of Unfolding

$$\int dt\, p_{\mathrm{Gen}}(t)\, p_{\mathrm{Sim}}(r|t) = p(r)$$

"Truth" | *Reconstruction*

"Detector Simulation"

GEANT4, FLUKA, …

Detector effects must be deconvolved to compare theoretical calculations to experimental measurements

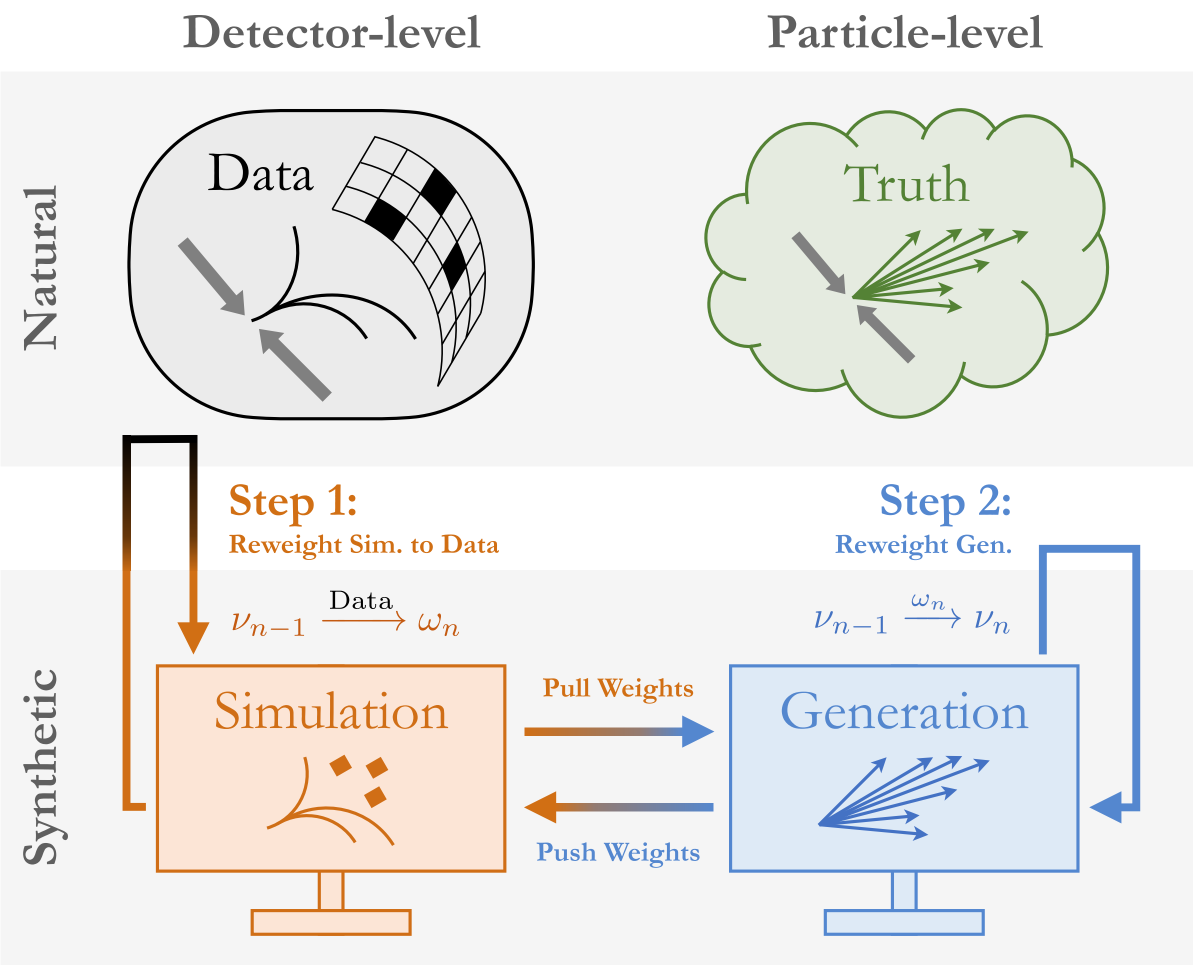
Iterative reweighting is a known solution to this problem for binned histogram data in low dimensions

Deep learning can lift these methods to act on unbinned data in high-dimensional phase space

[Richardson-Lucy deconvolution: Richardson, JOSA 1972; Lucy, AJ 1974] [Iterative Bayesian Unfolding: D'Agostini, NIM 1995]
[asymptotic equivalence to maximum likelihood reconstruction in Shepp, Vardi, IEEE 1982]

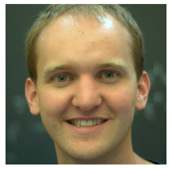
Deconvolution with OmniFold

Multi-dimensional unbinned detector corrections via iterated application of likelihood ratio trick



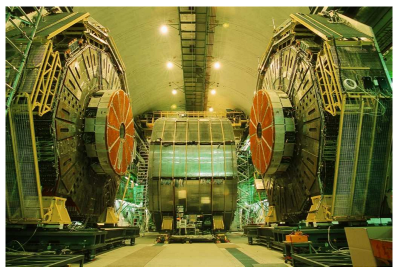
Use deep learning to compute reweighting factors

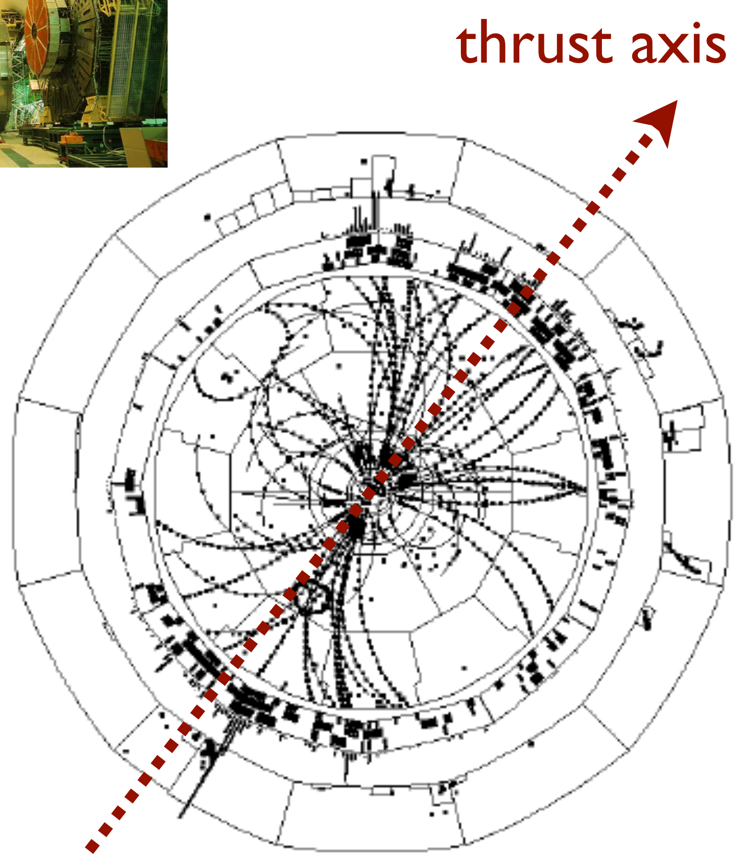
[Andreassen, Komiske, Metodiev, Nachman, JDT, PRL 2020]
[see complimentary approach in Bellagente, et al., SciPost 2020]

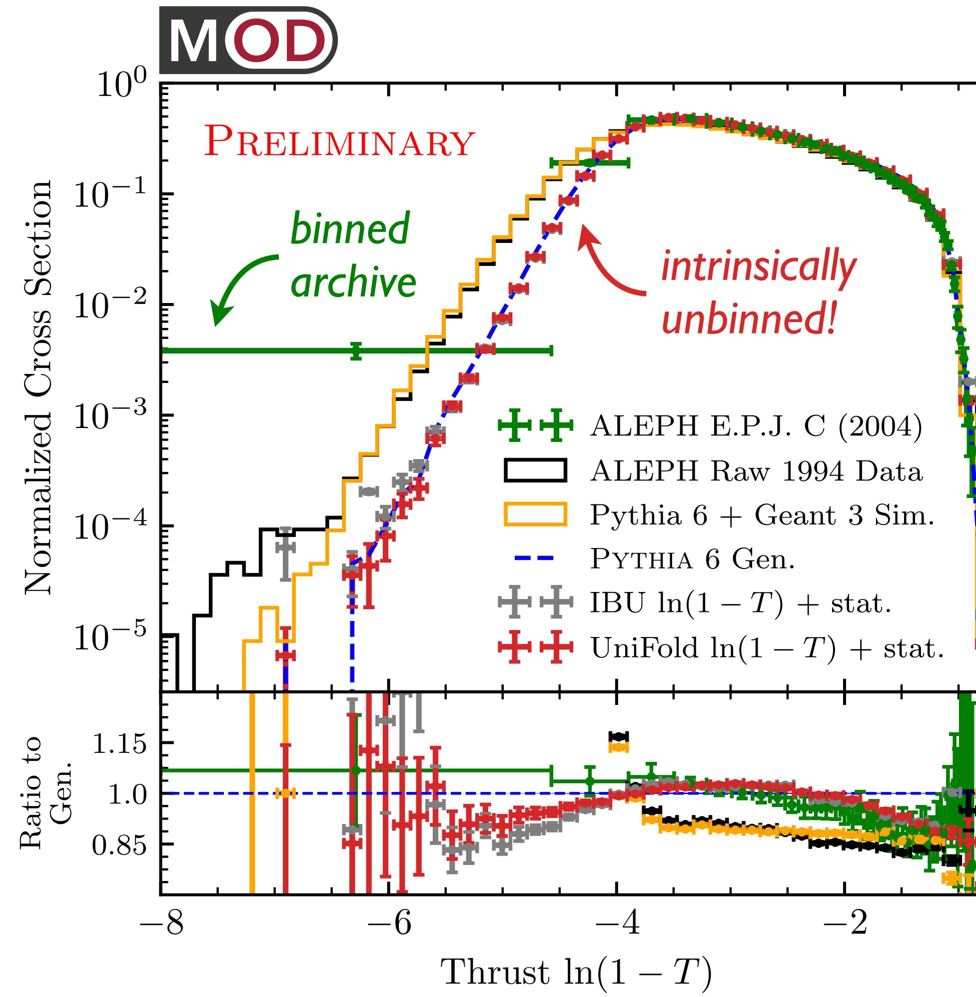


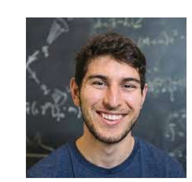

Deconvolution with OmniFold

Back to the Future with ALEPH Archival Data



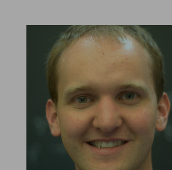
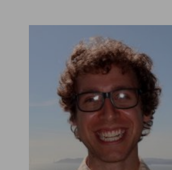




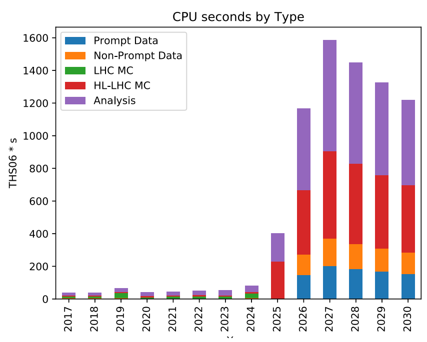


[talk by Badea, ICHEP 2020; cf. ALEPH, EPJC 2004]
[see also Badea, Baty, Chang, Innocenti, Maggi, McGinn, Peters, Sheng, JDT, Lee, PRL 2019]

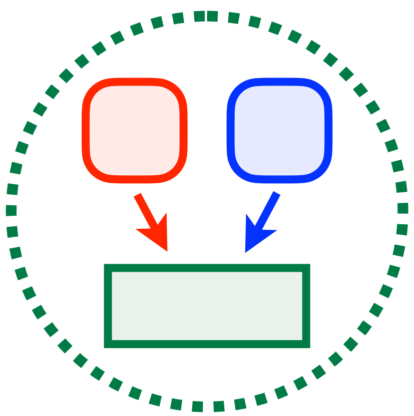
[Andreassen, Komiske, Metodiev, Nachman, JDT, PRL 2020]
[see complimentary approach in Bellagente, et al., SciPost 2020]

Deep Learning for Collider Physics Simulation



For another talk: Computational cost of HL-LHC have inspired new ideas for accelerating simulation



With large sets of i.i.d. data, you can perform weighted simulation via binary classification

Deep learning can augment "deep thinking" and address key simulation challenges in collider physics

[http://iaifi.org]