Precision Forward and Inverse Simulations for the LHC

Tilman Plehn

Universität Heidelberg

Bern, March 2022

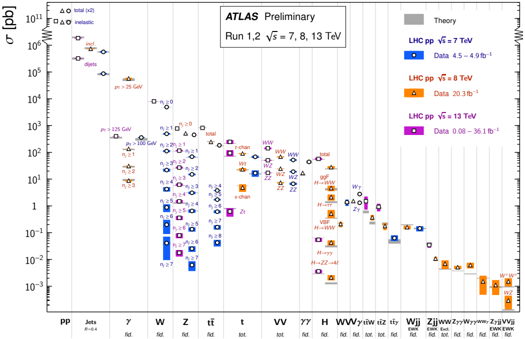
LHC goals

Fundamental questions

- particle nature of dark matter?
- origin of the Higgs mechanism? [hierarchy problem?]
- matter-antimatter asymmetry? [CP-symmetry]
- Standard Model all there is?

Rates

- many processes
- vastly different rates
- high precision
- predicted by theory



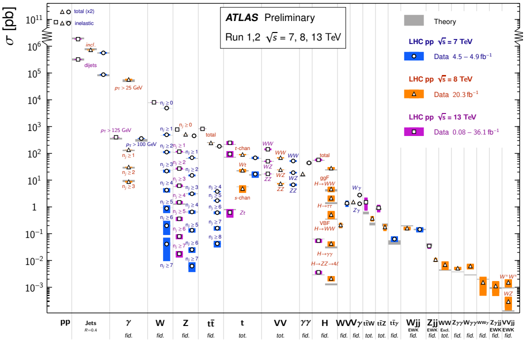
LHC goals

Fundamental questions

- particle nature of dark matter?
- origin of the Higgs mechanism? [hierarchy problem?]
- matter-antimatter asymmetry? [CP-symmetry]
- Standard Model all there is?

Rates

- many processes
- vastly different rates
- high precision
- predicted by theory
- but completely useless!

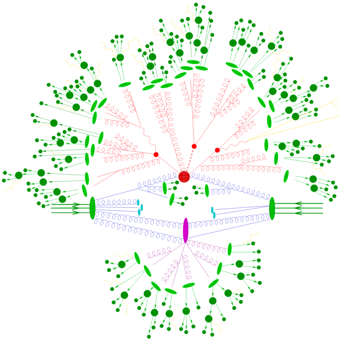


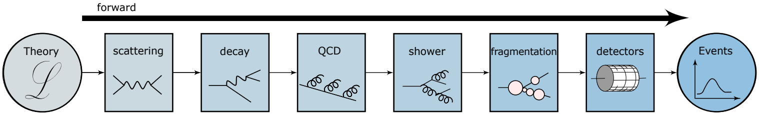
First-principle simulations

Simulation-based inference [likelihood-free inference]

- understand events completely
- Lagrangian to start
- perturbative QFT
- event generation [Sherpa, Madgraph, Pythia, Powheg]
- detector simulation

→ LHC events in virtual worlds





First-principle simulations

Simulation-based inference [likelihood-free inference]

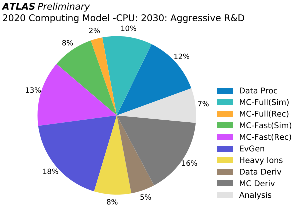
- understand events completely
- Lagrangian to start
- perturbative QFT
- event generation [Sherpa, Madgraph, Pythia, Powheg]
- detector simulation

$\to$ LHC events in virtual worlds

Forward LHC simulations

- HL-LHC: preparing for 25-fold data set
- simulated events $\sim$ expected events
- 1%-2% statistical uncertainty [NNLO/N^{3}LO]
- low-rate high-multiplicity processes
- time-dependent signal hypotheses

$\to$ Event generation limiting factor



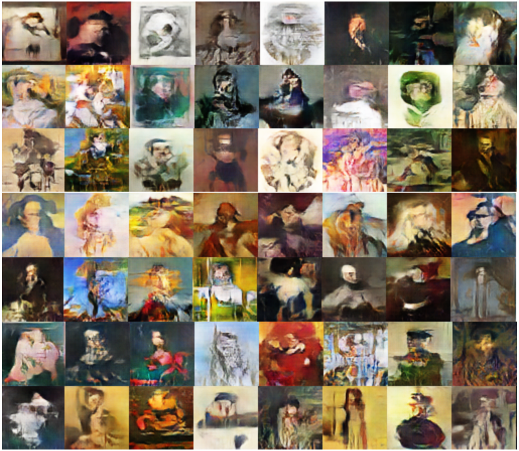
Generative networks

GANGogh [Bonafilia, Jones, Danyluk (2017)]

- neural network — learned function by minimizing loss

regression	$x \to f(x)$
classification	$x \to p(x) \in [0,1]$
generation	$x \to p_X(x)$ sampled $x \sim \mathcal{N}$

- networks to create new pieces of art
- train on 80,000 pictures
- generate portraits



Generative networks

GANGogh [Bonafilia, Jones, Danyluk (2017)]

- neural network — learned function by minimizing loss

  regression $x \to f(x)$
  classification $x \to p(x) \in [0,1]$
  generation $x \to p_X(x)$ sampled $x \sim \mathcal{N}$
- networks to create new pieces of art
- train on 80,000 pictures
- generate portraits

Edmond de Belamy [Caselles-Dupre, Fautrel, Vernier (2018)]

- trained on 15,000 portraits
- sold for $432.500

Generative networks

GANGogh [Bonafilia, Jones, Danyluk (2017)]

- neural network — learned function by minimizing loss

  regression $x \to f(x)$
  classification $x \to p(x) \in [0,1]$
  generation $x \to p_X(x)$ sampled $x \sim \mathcal{N}$
- networks to create new pieces of art
- train on 80,000 pictures
- generate portraits

Edmond de Belamy [Caselles-Dupre, Fautrel, Vernier (2018)]

- trained on 15,000 portraits
- sold for $432.500

LHC examples

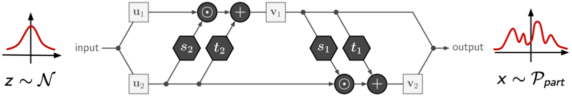
- jets [Nachman (2017), Carrazza-Dreyer (2019)...]
- events [Butter (2019), Review (2020)...]
- detector [Nachman (2017), Erdmann (2018), Kasieczka (2020), Krause-Shih (2021)...]
- unfolding [Omnifold, cGAN, cINN (2020)...]
- inference [QCD splittings (2020)...]
- compression [Rabemananjara (2021), ephemeral (2022)...]

Uncertain normalizing flows

Normalizing flows — INN [Ardizzone, Köthe]

- Gaussian latent space
- bijective mapping
- known Jacobian
- log-likelihood loss

→ Better than VAEs and GANs



Uncertain normalizing flows

Normalizing flows — INN [Ardizzone, Köthe]

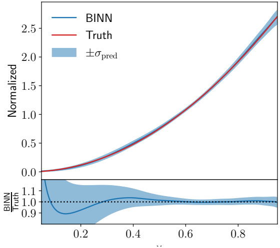
- Gaussian latent space
- bijective mapping
- known Jacobian
- log-likelihood loss

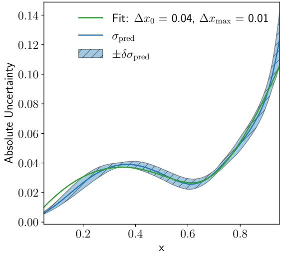
→ Better than VAEs and GANs

Bayesian INNs [Bellagente, Haußmann, Luchmann, TP]

- network weight distributions [Gal (2016)]
- sample for output [efficient ensembling]
- working for regression, classification
- events with error bars [density & uncertainty maps]
- 2D: wedge ramp, kicker ramp,...

→ INNs just fits





Precision generator

ML-event generators [Butter, Plehn..., CaloFLow: Krause & Shih]

- useful ML-playground
- training from event samples
  no momentum conservation
  no detector effects [sharper structures]

1- top-quark pairs $t\bar{t} \to 6$ jets [resonance peaks]

2- $Z_{\mu\mu} + \{1, 2, 3\}$ jets [Z-peak, variable jet number, jet-jet topology]

Precision generator

ML-event generators [Butter, Plehn..., CaloFLow: Krause & Shih]

- useful ML-playground
- training from event samples
  no momentum conservation
  no detector effects [sharper structures]

1- top-quark pairs $t\bar{t} \to 6$ jets [resonance peaks]

2- $Z_{\mu\mu} + \{1, 2, 3\}$ jets [Z-peak, variable jet number, jet-jet topology]

INN-generator [Butter, Heimel, Hummerich, Krebs, TP, Rousselot, Vent]

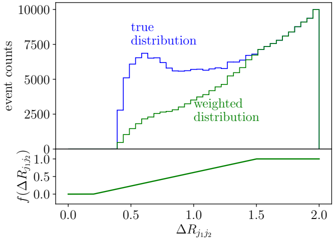
- challenging ΔR_{jj} features
- opposite of importance sampling

$$w^{(1\text{-jet})} = 1$$

$$w^{(2\text{-jet})} = f(\Delta R_{j_1,j_2})$$

$$w^{(3\text{-jet})} = f(\Delta R_{j_1,j_2}) f(\Delta R_{j_2,j_3}) f(\Delta R_{j_1,j_3})$$

$$f(\Delta R) = \frac{\Delta R - R_-}{R_+ - R_-} \qquad (\Delta R \in [R_-, R_+])$$



Precision generator

ML-event generators [Butter, Plehn..., CaloFLow: Krause & Shih]

- useful ML-playground
- training from event samples
  no momentum conservation
  no detector effects [sharper structures]

1- top-quark pairs $t\bar{t} \to 6$ jets [resonance peaks]

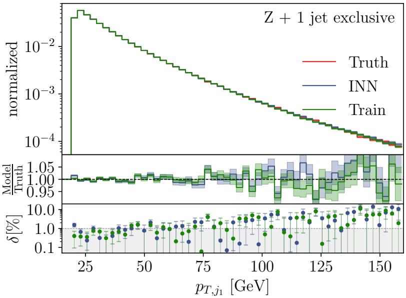
2- $Z_{\mu\mu} + \{1, 2, 3\}$ jets [Z-peak, variable jet number, jet-jet topology]

INN-generator [Butter, Heimel, Hummerich, Krebs, TP, Rousselot, Vent]

- challenging ΔR_{jj} features
- opposite of importance sampling

$$w^{(1\text{-jet})} = 1$$
$$w^{(2\text{-jet})} = f(\Delta R_{j_1, j_2})$$
$$w^{(3\text{-jet})} = f(\Delta R_{j_1, j_2}) f(\Delta R_{j_2, j_3}) f(\Delta R_{j_1, j_3})$$
$$f(\Delta R) = \frac{\Delta R - R_-}{R_+ - R_-} \qquad (\Delta R \in [R_-, R_+])$$

$\to$ Per-cent precision in reach



Precision generator

ML-event generators [Butter, Plehn..., CaloFLow: Krause & Shih]

- useful ML-playground
- training from event samples
  no momentum conservation
  no detector effects [sharper structures]

1- top-quark pairs $t\bar{t} \to 6$ jets [resonance peaks]

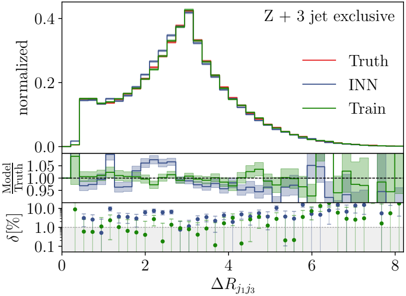
2- $Z_{\mu\mu} + \{1, 2, 3\}$ jets [Z-peak, variable jet number, jet-jet topology]

INN-generator [Butter, Heimel, Hummerich, Krebs, TP, Rousselot, Vent]

- challenging ΔR_{jj} features
- opposite of importance sampling

$$w^{(1\text{-jet})} = 1$$
$$w^{(2\text{-jet})} = f(\Delta R_{j_1,j_2})$$
$$w^{(3\text{-jet})} = f(\Delta R_{j_1,j_2}) f(\Delta R_{j_2,j_3}) f(\Delta R_{j_1,j_3})$$
$$f(\Delta R) = \frac{\Delta R - R_-}{R_+ - R_-} \qquad (\Delta R \in [R_-, R_+])$$

→ Per-cent precision in reach

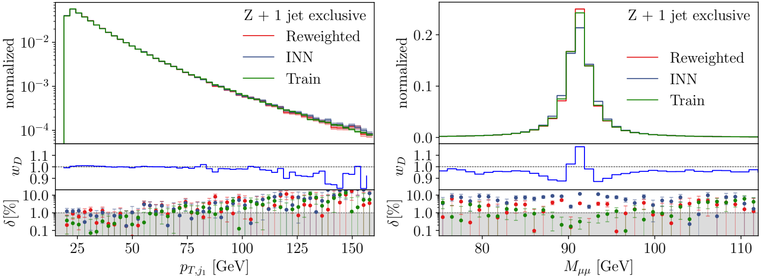


Controlled precision generator

Discriminator: training vs generated

- input $\{p_T, \eta, \phi, M, M_{\mu\mu}, \Delta R\}$
- output $D = 0(\text{generator}), 1(\text{truth})$
- decent generator training $D \approx 0.5$
- additional event weight $w_D = D/(1-D) \to 1$

$\to$ Dual use — control & reweight



Controlled precision generator

Discriminator: training vs generated

- input $\{p_T, \eta, \phi, M, M_{\mu\mu}, \Delta R\}$
- output $D = 0(\text{generator}), 1(\text{truth})$
- decent generator training $D \approx 0.5$
- additional event weight $w_D = D/(1-D) \to 1$

$\to$ Dual use — control & reweight

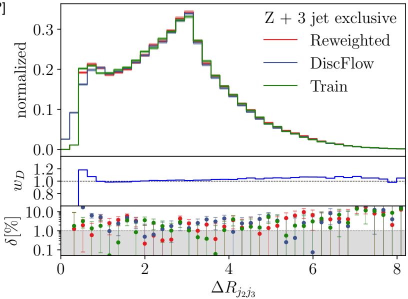
Joint DiscFlow training [GAN inspiration]

- adversarial loss unstable [Nash equilibrium??]
- coupling through weights

$$L_{\text{DiscFlow}} = -\sum_{i=1}^{B} w_D(x_i)^\alpha \log \frac{P(x_i)}{P_{\text{ref}}(x_i)}$$

$$\approx -\int dx \frac{P_{\text{ref}}^{\alpha+1}(x)}{P^\alpha(x)} \log \frac{P(x)}{P_{\text{ref}}(x)}$$

$\to$ Per-cent precision

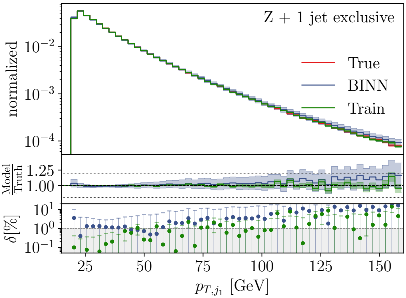


Uncertain precision generator

BINN generator

- Bayesian precision generator
- uncertainty over phase space
- low statistics challening

$\rightarrow$ Training-related error bars



Uncertain precision generator

BINN generator

- Bayesian precision generator
- uncertainty over phase space
- low statistics challening

$\rightarrow$ Training-related error bars

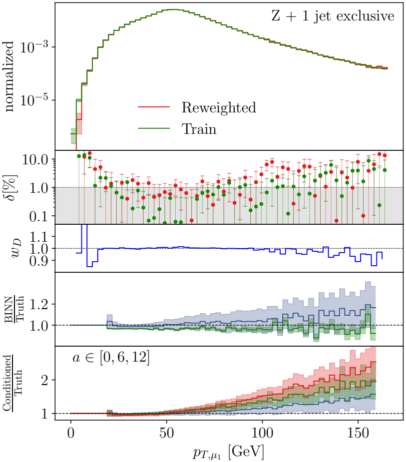
Theory uncertainties

- systematics from data augmentation
- adjust data in tails [$a = 0 \dots 30$]

$$w = 1 + a \left(\frac{p_{T,j_1} - 15 \text{ GeV}}{100 \text{ GeV}} \right)^2$$

- train conditionally on smeared a
- error bar from sampling a

$\rightarrow$ INNs for LHC standards

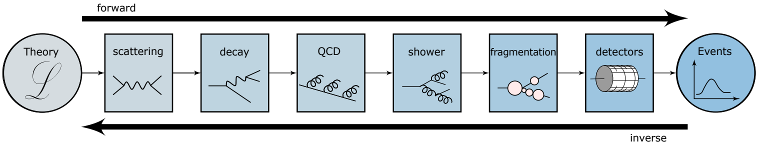


Inverse simulation

Invertible ML-simulation [see also Ben Nachman's seminar]

- detector unfolding known problem
- QCD parton from jet algorithm standard
- jet radiation combinatorics challenge
- decays established by top groups
- matrix element method the dream

$\to$ multi-dimensional, unbinned, statistical?



Inverse simulation

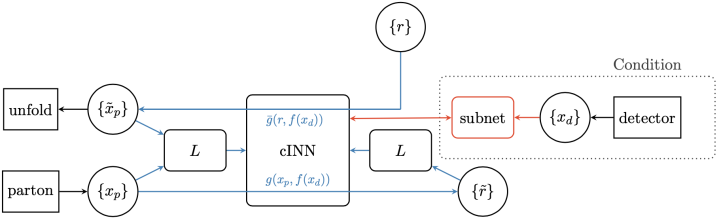
Invertible ML-simulation [see also Ben Nachman's seminar]

- detector unfolding known problem
- QCD parton from jet algorithm standard
- jet radiation combinatorics challenge
- decays established by top groups
- matrix element method the dream

$\rightarrow$ multi-dimensional, unbinned, statistical?

Conditional INN [Bellagente, Butter, Kasieczka, TP, Rousselot, Winterhalder, Ardizzone, Köthe]

- partonic events from $\{r\}$, given detector event



Inverse simulation

Invertible ML-simulation [see also Ben Nachman's seminar]

- detector unfolding known problem
- QCD parton from jet algorithm standard
- jet radiation combinatorics challenge
- decays established by top groups
- matrix element method the dream

$\rightarrow$ multi-dimensional, unbinned, statistical?

Conditional INN [Bellagente, Butter, Kasieczka, TP, Rousselot, Winterhalder, Ardizzone, Köthe]

- partonic events from $\{r\}$, given detector event
- maximum likelihood loss

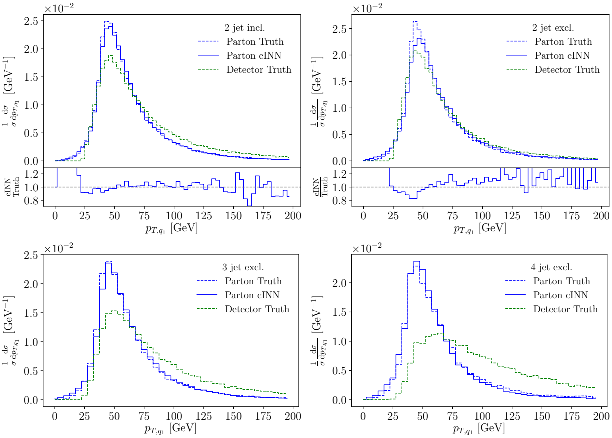
$$\begin{aligned} L &= -\left\langle \log p(\theta|x_p, x_d) \right\rangle_{x_p, x_d} \\ &= -\left\langle \log p(g(x_p, x_d)) + \log \left| \frac{\partial g(x_p, x_d)}{\partial x_p} \right| \right\rangle_{x_p, x_d} - \log p(\theta) + \text{const.} \end{aligned}$$

$\rightarrow$ Stable and statistically calibrated

Inverting to hard process

Undo QCD jet radiation

- nasty jet combinatorics, little information
- detector level: $pp \to ZW$+jets [variable number of objects]
- hard process given, ME vs PS jets from network



Inverting to hard process

Undo QCD jet radiation

- nasty jet combinatorics, little information
- detector level: $pp \to ZW$+jets [variable number of objects]
- hard process given, ME vs PS jets from network

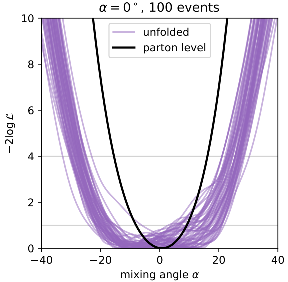
Matrix element method [Butter, Heimel, Martini, Peitzsch, TP (soon)]

- parameter likelihood from parton-level events

$$\mathcal{L}(\theta) = \prod_{i=1}^{N} p(\vec{x}^{(i)}|\theta) = \prod_{i=1}^{N} \frac{1}{\sigma_{\text{fid}}(\theta)} \int d^m z \, \frac{d^m\sigma(\theta)}{dz_1 \dots dz_m} \, T(\vec{x}^{(i)}, \vec{z})$$

$$T(\vec{x}, \vec{z}) = p_{\text{INN}}(\vec{z}|\vec{x})\epsilon(\vec{x}) \qquad \Rightarrow \qquad \mathcal{L}(\theta) = \prod_{i=1}^{N} \frac{\epsilon(\vec{x}^{(i)})}{\sigma_{\text{fid}}(\theta)} \int d^m z \, \frac{d^m\sigma(\theta)}{dz_1 \dots dz_m} \, p_{\text{INN}}(\vec{z}|\vec{x}^{(i)})$$

$$= \prod_{i=1}^{N} \frac{\epsilon(\vec{x}^{(i)})}{\sigma_{\text{fid}}(\theta)} \left\langle \frac{d^m\sigma(\theta)}{dz_1 \dots dz_m} \right\rangle_{\vec{z} \sim p_{\text{INN}}}$$



Inverting to QCD

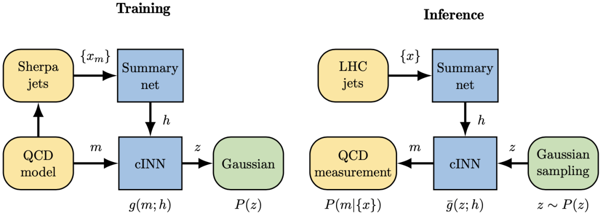
cINN for inference [Bieringer, Butter, Heimel, Höche, Köthe, TP, Radev]

- condition jets with QCD parameters
  train model parameters $\to$ Gaussian latent space
  test Gaussian sampling $\to$ parameter measurement
- beyond C_A vs C_F [Kluth etal]

$$P_{qq} = C_F \left[D_{qq} \frac{2z(1-y)}{1-z(1-y)} + F_{qq}(1-z) + C_{qq} yz(1-z) \right]$$

$$P_{gg} = 2C_A \left[D_{gg} \left(\frac{z(1-y)}{1-z(1-y)} + \frac{(1-z)(1-y)}{1-(1-z)(1-y)} \right) + F_{gg} z(1-z) + C_{gg} yz(1-z) \right]$$

$$P_{gq} = T_R \left[F_{qq} \left(z^2 + (1-z)^2 \right) + C_{gq} yz(1-z) \right]$$



Inverting to QCD

cINN for inference [Bieringer, Butter, Heimel, Höche, Köthe, TP, Radev]

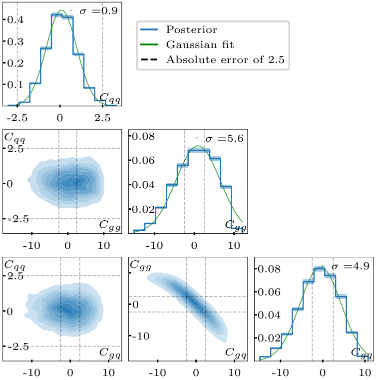
- condition jets with QCD parameters
  train model parameters $\to$ Gaussian latent space
  test Gaussian sampling $\to$ parameter measurement
- beyond C_A vs C_F [Kluth etal]

$$P_{qq} = C_F \left[D_{qq} \frac{2z(1-y)}{1-z(1-y)} + F_{qq}(1-z) + C_{qq} yz(1-z) \right]$$

$$P_{gg} = 2C_A \left[D_{gg} \left(\frac{z(1-y)}{1-z(1-y)} + \frac{(1-z)(1-y)}{1-(1-z)(1-y)} \right) + F_{gg} z(1-z) + C_{gg} yz(1-z) \right]$$

$$P_{gq} = T_R \left[F_{qq} \left(z^2 + (1-z)^2 \right) + C_{gq} yz(1-z) \right]$$

- idealized shower [Sherpa]
- More ML-opportunities...



ML for LHC Theory

ML-applications in LHC analysis and theory

- just another numerical tool for a numerical field
- driven by money from data science, medical research
- goals are...

  ...improve established tasks
  ...develop new tools for established tasks
  ...transform through new ideas
- particle physics as exciting as our ideas

$\rightarrow$ Opportunity to make a difference!

Contents

Machine Learning and LHC Event Generation

Anja Butter[1,2], Tilman Plehn[1], Steffen Schumann[3] (Editors),
Simon Badger[4], Sascha Caron[5, 6] Kyle Cranmer[7,8], Francesco Armando Di Bello[9],
Etienne Dreyer[10], Stefano Forte[11], Sanmay Ganguly[12], Dorival Gonçalves[13], Eilam Gross[10],
Theo Heimel[1], Gudrun Heinrich[14], Lukas Heinrich[15], Alexander Held[16], Stefan Höche[17],
Jessica N. Howard[18], Philip Ilten[19], Joshua Isaacson[17], Timo Janßen[3], Stephen Jones[20],
Marumi Kado[9,21], Michael Kagan[22], Gregor Kasieczka[23], Felix Kling[24], Sabine Kraml[25],
Claudius Krause[26], Frank Krauss[20], Kevin Kröninger[27], Rahool Kumar Barman[13],
Michel Luchmann[1], Vitaly Magerya[14], Daniel Maitre[20], Bogdan Malaescu[2],
Fabio Maltoni[28,29], Till Martini[30], Olivier Mattelaer[28], Benjamin Nachman[31,32],
Sebastian Pitz[1], Juan Rojo[33,34], Matthew Schwartz[35], David Shih[25], Frank Siegert[36],
Roy Stegeman[11], Bob Stienen[5], Jesse Thaler[37], Rob Verheyen[38], Daniel Whiteson[18],
Ramon Winterhalder[28], and Jure Zupan[19]

Abstract

First-principle simulations are at the heart of the high-energy physics research program. They link the vast data output of multi-purpose detectors with fundamental theory predictions and interpretation. This review illustrates a wide range of applications of modern machine learning to event generation and simulation-based inference, including conceptional developments driven by the specific requirements of particle physics. New ideas and tools developed at the interface of particle physics and machine learning will improve the speed and precision of forward simulations, handle the complexity of collision data, and enhance inference as an inverse simulation problem.

Submitted to the Proceedings of the US Community Study
on the Future of Particle Physics (Snowmass)