Algorithmic challenges & opportunities for HEP MCs

facilitating progress & new collaborations

Steffen Schumann

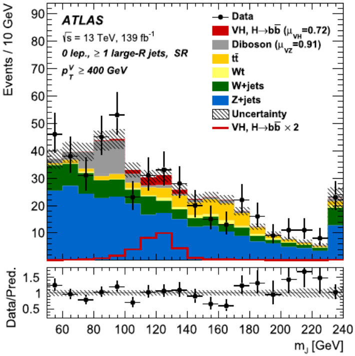
Institut für Theoretische Physik, Universität Göttingen

GEFÖRDERT VOM

Bundesministerium für Bildung und Forschung

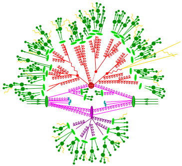
Graz 22/09/22

HEP MCs – construction paradigms & challenges



[G. Aad *et al.* [ATLAS], Phys. Lett. B **816** (2021), 136204]

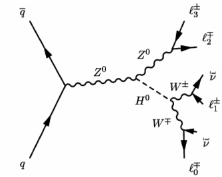
- forward sim of (B)SM hypotheses
  - ⇝ systematic expansion QCD/EW
  - ⇝ quantifiable uncertainties
- supplemented by NP effects
  - ⇝ calibrated to data

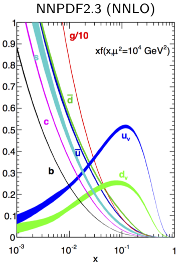


- aim for higher accuracy: towards NNLO QCD, NLO EW, NLL, etc.
- develop better NP models, interface with perturbative phases
- need to improve resource efficiency given limited computing budgets
  - ⇝ search for bottlenecks in workflow, improve algorithms
  - ⇝ facilitate physics progress & new scientific collaborations

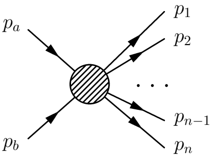
HEP MCs – construction paradigms & challenges

$$\sigma_{pp\to X_n} = \sum_{ab} \int \mathrm{d}x_a \mathrm{d}x_b \, \mathrm{d}\Phi_n \, f_a(x_a, \mu_F^2) f_b(x_b, \mu_F^2) \, |\mathcal{M}_{ab\to X_n}|^2 \, \Theta_n(p_1, \ldots, p_n)$$





- ↪ multi-modal, wildly fluctuating target distribution
- ↪ real- & virtual corrections, IR subtractions
- ↪ subject to non-trivial acceptance cuts Θ_n
- ↪ Monte-Carlo phase space sampling $[\dim[\Phi_n] = 3n - 4]$

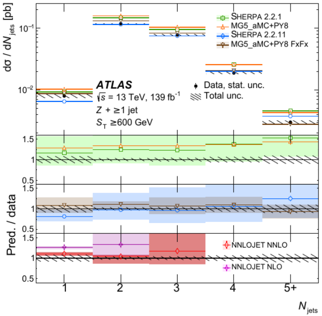


Novel Sampling Algorithms

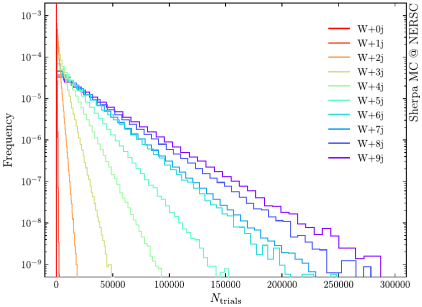
Phase-space sampling challenges

when expensive integrands meet poor efficiencies

- LHC phenomenology requires multi-particle hard processes @ (N)NLO



[ATLAS arXiv:2205.02597 [hep-ex]]



[Höche, Prestel and Schulz, Phys. Rev. D **100** (2019) no.1, 014024]

- huge computational resources needed for experimental samples

  $\rightsquigarrow$ need for speed-ups, reduction of negative weights, resampling etc

[Bothmann et al., arXiv:2209.00843 [hep-ph]], [Danziger et al., arXiv:2110.15211 [hep-ph]], [Andersen and Maier, Eur. Phys. J. C **82** (2022) no.5, 433], [Andersen et al., Eur. Phys. J. C **80** (2020) no.11, 1007], [Frederix et al., JHEP **07** (2020), 238], [Nachman and Thaler, Phys. Rev. D **102** (2020) no.7, 076004], ...

Phase-space sampling challenges

Variance reduction – importance sampling in a nut-shell

- consider generic integral over target function $f(x)$, $x \in V \subseteq \mathbb{R}^d$
- choose variable mapping $y : V \to U \subseteq \mathbb{R}^d$

$$I = \int_V \mathrm{d}^d x\, f(x) = \int_U \mathrm{d}^d y\, \left.\frac{f(x)}{g(x)}\right|_{x \equiv x(y)} \qquad \text{with} \quad \left|\frac{\partial y(x)}{\partial x}\right| = g(x)$$

$\hookrightarrow$ reduce variance of MC estimate through suitable proposal $g(x)$

$\hookrightarrow$ multi-modal target use multi-channel $g(x) = \sum_i \beta_i g_i(x)$ with $\sum_i \beta_i = 1$

$$I = \int_V \mathrm{d}^d x\, f(x) = \sum_i \int_V \mathrm{d}^d x\, \beta_i\, g_i(x)\, \frac{f(x)}{g(x)} = \sum_i \int_{U_i} \mathrm{d}^d y_i\, \beta_i\, \left.\frac{f(x)}{g(x)}\right|_{x \equiv x(y_i)}$$

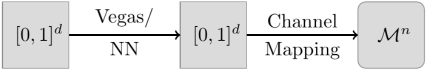
- ME generators construct channels, mapping out prominent features
- adaptive methods to optimize channel weights β_i

Neural Network Importance Sampling

improve sampling through Normalizing Flows

[Bothmann et al., SciPost Phys. **8** (2020) no.4, 069], [Gao et al., Phys. Rev. D **101** (2020) no.7, 076002]

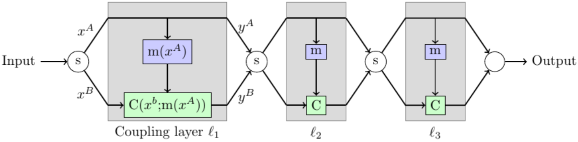
- remapping of random numbers entering phase-space channels
  ↪ just as we do it with Vegas optimisation



↪ bijective maps, called coupling layers C (simple Jacobian)

$$\left.\begin{array}{lcl} x^A & \to & y^A := x^A \\ x^B & \to & y^B := C\left(x^B; m(x^A)\right) \end{array}\right\} \; J = \left| \begin{pmatrix} \mathrm{diag}(1) & 0 \\ \frac{\partial C}{\partial m}\frac{\partial m}{\partial x_A} & \frac{\partial C}{\partial x_B} \end{pmatrix} \right| = \left| \frac{\partial C}{\partial x_B} \right|$$

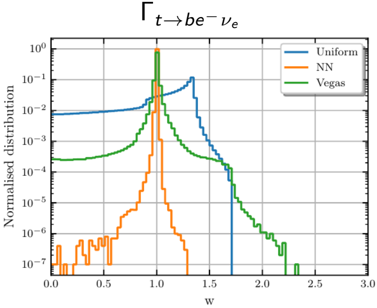
↪ m piecewise polynomial function, with NN adopted parameters

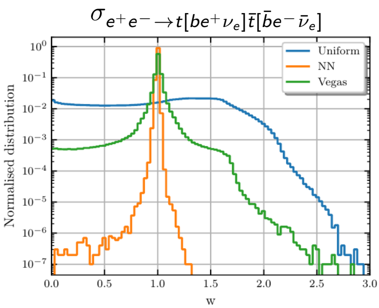


⇝ very expressive non-linear variable transformations (non-factorisable)

Neural Importance Sampling

improve sampling through Normalizing Flows – cont'd





	top decays		top-pair production		$gg \to 3g$		$gg \to 4g$	
Sample	ϵ_{uw}	E_N [GeV]	ϵ_{uw}	E_N [fb]	ϵ_{uw}	E_N [fb]	ϵ_{uw}	E_N [fb]
Uniform	59 %	0.1679(2)	35 %	1.5254(8)	3.0 %	24806(55)	2.7 %	9869(20)
Vegas	50 %	0.16782(4)	40 %	1.5251(1)	27.7 %	24813(23)	31.8 %	9868(10)
NN	84 %	0.167865(5)	78 %	1.52531(2)	64.3 %	24847(21)	48.9 %	9859(10)

- smaller impact for more complicated (multi-channel) processes
- GPU evaluation of MEs desirable [Bothmann et al., arXiv:2106.06507 [hep-ph]]

Neural Importance Sampling – spin offs

numerical evaluation of multi-loop Feynman integrals

- compute Laurent-series coefficients, based on *Sector Decomposition*
  [Heinrich, Int. J. Mod. Phys. A **23** (2008), 1457-1486]

$$G = \int_{-\infty}^{\infty} \left(\prod_{l=1}^{L} \frac{d^D k_l}{i\pi^{\frac{D}{2}}} \right) \prod_{j=1}^{N} \frac{1}{P_j^{\nu_j}(\{k\},\{p\},m_j^2)}$$

$$= \int_0^1 \prod_{j=1}^{N-1} dx_j \, x_j^{\nu_j - 1} \frac{U^{\nu-(L+1)D/2}}{F^{\nu - LD/2}} = \int_0^1 \prod_{j=1}^{N-1} dx_j \, \mathcal{I}(\vec{x})$$

$\rightsquigarrow$ analytic continuation in complex plane, closed integration contour

$$0 = \oint_{\mathcal{C}} \prod_{j=1}^{N} \mathrm{d}z_j \, \mathcal{I}(\vec{z}) = \int_0^1 \prod_{j=1}^{N} \mathrm{d}x_j \, \mathcal{I}(\vec{x}) + \int_\gamma \prod_{j=1}^{N} \mathrm{d}z_j \, \mathcal{I}(\vec{z})$$

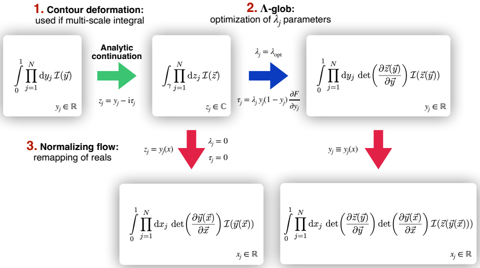
$$\Leftrightarrow \quad \int_0^1 \prod_{j=1}^{N} \mathrm{d}x_j \, \mathcal{I}(\vec{x}) = - \int_\gamma \prod_{j=1}^{N} \mathrm{d}z_j \, \mathcal{I}(\vec{z}) = \int_0^1 \prod_{j=1}^{N} \mathrm{d}x_j \; \det\left(\frac{\partial \vec{z}(\vec{x})}{\partial \vec{x}} \right) \mathcal{I}(\vec{z}(\vec{x}))$$

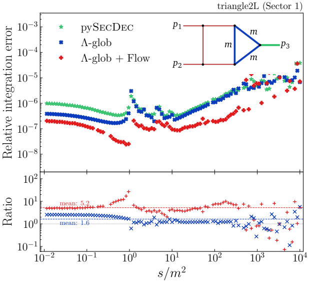
Neural Importance Sampling – spin offs

numerical evaluation of multi-loop Feynman integrals

[Winterhalder et al., "Targeting multi-loop integrals with neural networks", arXiv:2112.09145 [hep-ph]]

- consider *Sector Decomposition* via `pySecDec` with default contours
  - ⇝ search for contour deformation parameters minimizing integration error
  - ⇝ employ Normalizing Flows to remap integration variables





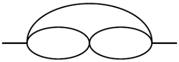
- achieve reduction of numerical uncertainties
- relevant for high-precision LHC calculations

Neural Importance Sampling – spin offs

numerical evaluation of multi-loop Feynman integrals

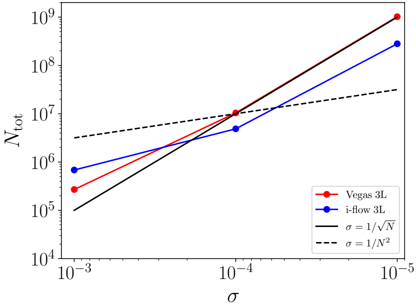
[Jinno, Kälin, Liu and Rubira: "Machine Learning Post-Minkowskian Integrals", arXiv:2209.01091 [hep-th]]

- Jinno et al. consider loop-integrals for gravitational binary dynamics
- benchmark `Vegas` and `i-flow` for known 1-, 2- & 3-loop integrals



$(\boldsymbol{D})$

$$\boldsymbol{D}_0 = e^{3\epsilon\gamma_E} \int \frac{\mathrm{d}^d\ell_1 \mathrm{d}^d\ell_2 \mathrm{d}^d\ell_3}{\pi^{3d/2}} \frac{(\boldsymbol{q}^2)^{5-3d/2}}{\boldsymbol{\ell}_1^2\, \boldsymbol{\ell}_2^2\, \boldsymbol{\ell}_3^2\, (\boldsymbol{\ell}_{13}-\boldsymbol{q})^2\, (\boldsymbol{\ell}_{23}-\boldsymbol{q})^2}$$



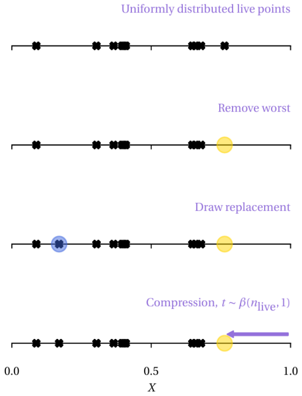
- `i-flow` can outperform `Vegas`
  - $\rightsquigarrow$ higher dimensions, complexity
  - $\rightsquigarrow$ approaches precision target earlier
- branch out in other communities

Other sampling ideas

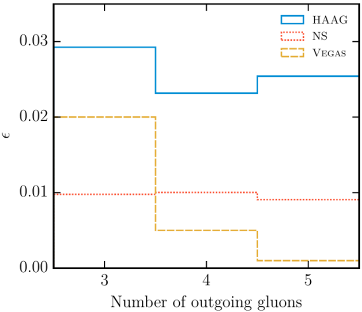
Exploring phase space with Nested Sampling

[Yallup, Janßen, Schumann and Handley, Eur. Phys. J. C **82** (2022), 8]

- transfer Bayesian inference algorithm to HEP event generation
  - $\rightsquigarrow$ applications in cosmology, statistical thermodynamics, material science
  - $\rightsquigarrow$ wide range of existing tools, *e.g.* `PolyChord` [Handley, Hobson and Lasenby]
- consider uniform prior, posterior matching target distribution (ME$\times$PS)



unweighting efficiency $gg \to Ng$



Surrogate Models

Employ fast ME×PS surrogates for event unweighting

[K. Danziger et al., SciPost Phys. **12** (2022), 164]

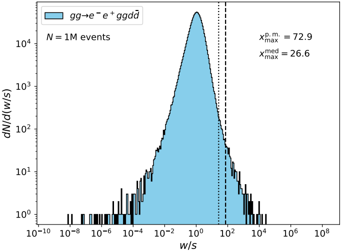
$$d\sigma_{ab\to n}\big|_{p_a,p_b,\{p_i\}} = \underbrace{f_a(x_a,\mu_F)\; f_b(x_b,\mu_F)\big|\mathcal{M}_{ab\to n}\big|^2\;|J_{\Phi_n}|}_{\text{approximate by surrogate, e.g. from NN}}\, dx_a dx_b\; d\Phi_n\big|_{p_a,p_b,\{p_i\}}$$

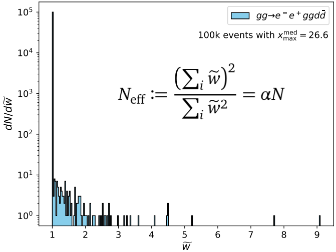
Algorithm 2: Two-stage rejection-sampling unweighting algorithm using an event-wise weight estimate.

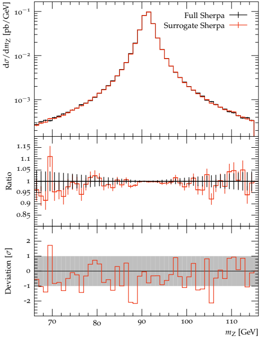
```
while true do
    generate phase-space point u;
    calculate approximate event weight s;
    generate uniform random number R_1 ∈ [0, 1);
    # first unweighting step
    if s > R_1 · w_max then
        calculate exact event weight w;
        determine ratio x = w/s;
        generate uniform random number R_2 ∈ [0, 1);
        # second unweighting step
        if x > R_2 · x_max then
            return u and w̃ = max(1, s/w_max) · max(1, x/x_max)
        end
    end
end
```

Surrogate Unweighting

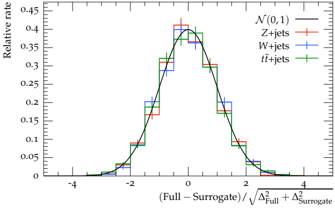
- **NN ME×PS surrogates – performance measures**







- validation for $W/Z + 4j$, $t\bar{t} + 3j$



$\hookrightarrow$ fully compatible samples

Surrogate Unweighting

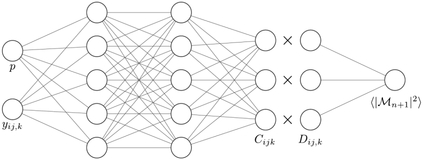
- **NN ME×PS surrogates – performance measures**
  ↪ speed-ups for $t\bar{t}+3j$ partonic channels

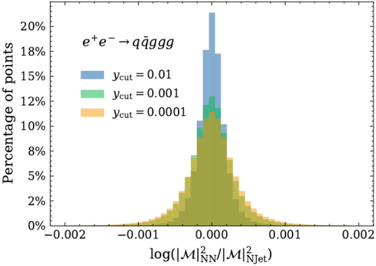
	$gg \to t\bar{t}ggg$	$ug \to t\bar{t}ggu$	$uu \to t\bar{t}guu$	$u\bar{u} \to t\bar{t}gd\bar{d}$
ϵ_{full}	1.1e−2	7.3e−3	6.8e−3	6.6e−4
$\epsilon_{\text{1st,surr}}$	8.7e−3	5.8e−3	4.7e−3	3.6e−4
$\langle t_{\text{full}}\rangle / \langle t_{\text{surr}}\rangle$	39312	2417	199	64
$x_{\text{max}}^{\text{p.m.}}$	52.03	32.52	69.76	326.19
$\epsilon_{\text{2nd,surr}}^{\text{p.m.}}$	2.4e−2	3.8e−2	2.1e−2	5.6e−3
$\alpha^{\text{p.m.}}$	0.9989	0.9984	0.9994	0.9981
$f_{\text{eff}}^{\text{p.m.}}$	2.21	4.89	1.47	0.19
$x_{\text{max}}^{\text{med}}$	30.40	19.14	27.78	25.34
$\epsilon_{\text{2nd,surr}}^{\text{med}}$	4.3e−2	6.4e−2	5.1e−2	7.1e−2
α^{med}	0.9983	0.9966	0.9943	0.9321
$f_{\text{eff}}^{\text{med}}$	3.90	8.26	3.91	2.22

Surrogate Unweighting – dedicated ME emulation methods

provide use case for improved ME emulators

- factorisation-aware ME emulator [Maître and Truong, JHEP **11** (2021), 066]
  $\rightsquigarrow$ fitting coefficients of Catani–Seymour dipoles: $\langle|\mathcal{M}_{n+1}|^2\rangle = \sum_{\{ijk\}} C_{ijk} D_{ij,k}$



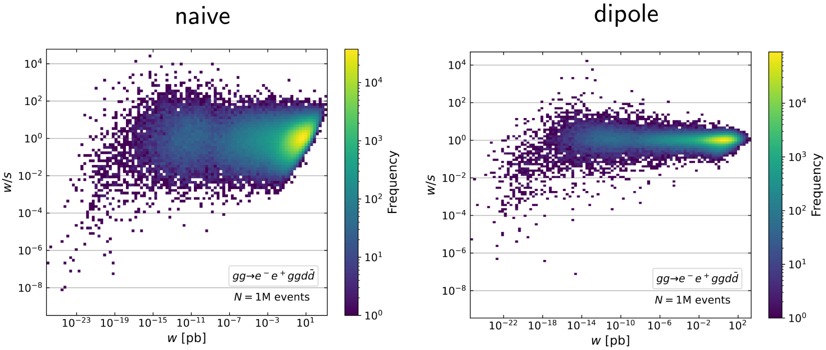


- alternative approaches focused on loop-induced channels
  [Aylett-Bullock, Badger and Moodie, JHEP **08** (2021), 066]
  [Badger et al., arXiv:2206.14831 [hep-ph]]

factorisation-aware ME emulator for hadronic processes

[Janßen, Maître, Schumann, Siegert and Truong, to appear soon]

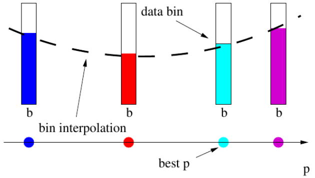
- generalise method to initial-state dipoles, apply to high multiplicities
  - $\rightsquigarrow$ find significant improvements compared to naive NN setup
  - $\rightsquigarrow$ $pp \to Z + 4j$ channel: $f_{\text{eff}}^{\text{med,naive}} = 4.7 \to f_{\text{eff}}^{\text{med,dipole}} = 15$



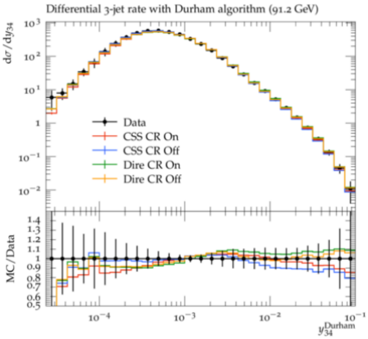
Novel Tuning Strategies

Non-perturbative models

- modelling of non-perturbative phenomena corner stone of HEP MCs
  $\rightsquigarrow$ lack of first-principle ansatz, *e.g.* Lund- & Cluster models, MPI etc
- need to calibrate $\mathcal{O}(10 - 100)$ model parameters with experimental data
- very costly to evaluate for different energies, colliders, lots of measurements
  $\rightsquigarrow$ grid search not feasible, parametrize/model MC response
  $\rightsquigarrow$ `Professor` [Hoeth et al.] & `Apprentice` [Krishnamoorthy et al.] tools



$$\chi^2 = \sum_{\text{bins}} \frac{(\text{interpolation} - \text{data})^2}{\sigma^2}$$



[Chahal and Krauss, SciPost Phys. **13** (2022) no.2]

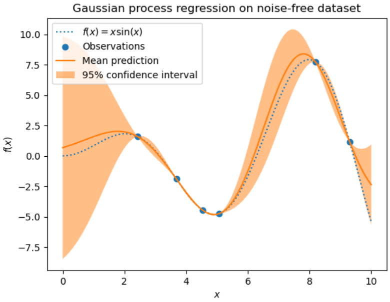
- fast turn-overs desirable, less resource intense, uncertainty estimates

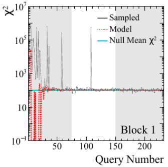
Non-perturbative models – tuning

Hadronization tuning using Bayesian optimization

[Ilten, Williams and Yang, JINST **12** (2017) no.04, P04028]

- Bayesian optimization standard strategy for (expensive) black-box functions
  $\rightsquigarrow$ ML hyperparameters optimization [Snoek et al., arxiv.1206.2944 [stat.ML]]
- suitable for problems with up to 20 parameters, need rather few evaluations
- adapt prior distribution (Gaussian process) through function evaluations
  $\rightsquigarrow$ non-parametric, statistical interpretation/uncertainties
- promising feasibility study, closure test with `Pythia` Monash tune [Skands 2014]
  $\rightsquigarrow$ model global χ^2-measure of MC vs. Monash



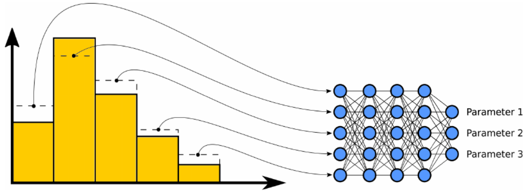


Parameter	Monash Value	Tune Value	Range Considered
sigma	0.335	$0.333^{+0.001}_{-0.002}$	$[0,1]$
bLund	0.98	$1.04^{+0.01}_{-0.02}$	$[0.2,2]$
aExtraSQuark	0	$0^{+0.07}_{-0}$	$[0,2]$
aExtraDiQuark	0.97	$1.48^{+0.15}_{-0.14}$	$[0,2]$
rFactC	1.32	1.38 ± 0.06	$[0,2]$
rFactB	0.855	0.887 ± 0.015	$[0,2]$

Non-perturbative models – tuning

Machine Learning parameter responses

- alternative approach based on Neural Networks MCNNTUNES
  [Lazzarin, Alioli and Carrazza, Comput. Phys. Commun. **263** (2021), 107908]
- learn the inverted model of a generator (here Pythia8)
  $\rightsquigarrow$ feed in (replicas of) data and infer about model parameters



Learning how to hadronize ...

- attempts to learn (simplified) hadronization process from training data
  [Bíró, Tankó-Bartalis and Barnaföldi, arXiv:2111.15655 [hep-ph]]
  [Ilten, Menzo, Youssef and Zupan, arXiv:2203.04983 [hep-ph]]
  [Ghosh, Ju, Nachman and A. Siodmok, arXiv:2203.12660 [hep-ph]]

Summary/Conclusions

possible HEP MC development perspectives

(i) development and implementation of novel sampling algorithms
- $\rightsquigarrow$ relevance for wide range of integration/sampling problems
- $\rightsquigarrow$ cross talk to many other fields, *e.g.* ML, Lattice FT, cosmo, industry, ...

(ii) fast and accurate surrogate models (partial event weights)
- $\rightsquigarrow$ resource & cost efficiency
- $\rightsquigarrow$ use for speed-ups, systematic variations, uncertainty estimations, ...
- $\rightsquigarrow$ possibly new use cases, *e.g.* in PDF/FF fitting, data analysis, ...

(iii) development of novel tuning and model calibration methods
- $\rightsquigarrow$ flexible & efficient generator tuning, tune/parameter uncertainties
- $\rightsquigarrow$ inform non-perturbative model construction, better use of HEP data
- $\rightsquigarrow$ general theme of *numerical optimization* problems: ML, statistics

... GPUs, ML, Quantum Computing algorithms, etc