Algorithmic challenges & opportunities for HEP MCs
facilitating progress & new collaborations

Steffen Schumann
Institut für Theoretische Physik, Universität Göttingen

Graz 22/09/22
HEP MCs – construction paradigms & challenges

- forward sim of (B)SM hypotheses
  \sim \text{systematic expansion QCD/EW}
  \sim \text{quantifiable uncertainties}
- supplemented by NP effects
  \sim \text{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
  \sim \text{search for bottlenecks in workflow, improve algorithms}
  \sim \text{facilitate physics progress & new scientific collaborations}

[G. Aad et al. [ATLAS], Phys. Lett. B 816 (2021), 136204]
HEP MCs – construction paradigms & challenges

\[
\sigma_{pp \rightarrow X_n} = \sum_{ab} \int dxa dxb \, d\Phi_n \, f_a(x_a, \mu_F^2) f_b(x_b, \mu_F^2) \left| M_{ab \rightarrow X_n} \right|^2 \Theta_n(p_1, \ldots, p_n)
\]

\(\rightarrow\) multi-modal, wildly fluctuating target distribution

\(\rightarrow\) real- & virtual corrections, IR subtractions

\(\rightarrow\) subject to non-trivial acceptance cuts \(\Theta_n\)

\(\rightarrow\) Monte-Carlo phase space sampling \([\text{dim}[\Phi_n] = 3n - 4]\)
Novel Sampling Algorithms
Phase-space sampling challenges

when expensive integrands meet poor efficiencies

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

huge computational resources needed for experimental samples

\[ \sim \] need for speed-ups, reduction of negative weights, resampling etc

- [ATLAS arXiv:2205.02597 [hep-ex]]
- [Höche, Prestel and Schulz, Phys. Rev. D 100 (2019) no.1, 014024]
Phase-space sampling challenges

Variance reduction – importance sampling in a nutshell

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

\[
I = \int_V d^d x \ f(x) = \int_U d^d y \ \frac{f(x)}{g(x)} \bigg|_{x \equiv x(y)} \quad \text{with} \quad \left| \frac{\partial y(x)}{\partial x} \right| = g(x)
\]

\( \leftarrow \) reduce variance of MC estimate through suitable proposal \( g(x) \)
\( \leftarrow \) multi-modal target use multi-channel \( g(x) = \sum_i \beta_i g_i(x) \) with \( \sum_i \beta_i = 1 \)

\[
I = \int_V d^d x \ f(x) = \sum_i \int_V d^d x \ \beta_i g_i(x) \ \frac{f(x)}{g(x)} = \sum_i \int_{U_i} d^d y_i \ \beta_i \ \frac{f(x)}{g(x)} \bigg|_{x \equiv x(y_i)}
\]

- ME generators construct channels, mapping out prominent features
- adaptive methods to optimize channel weights \( \beta_i \)
Neural Network Importance Sampling

**improve sampling through Normalizing Flows**


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

\[ \begin{align*}
[0, 1]^d & \xrightarrow{\text{Vegas/NN}} [0, 1]^d \xrightarrow{\text{Channel Mapping}} \mathcal{M}^n
\end{align*} \]

- bijective maps, called coupling layers \( C \) (simple Jacobian)

\[
\begin{align*}
  x^A & \rightarrow y^A := x^A \\
  x^B & \rightarrow y^B := C(x^B; m(x^A))
\end{align*}
\]

\[
J = \begin{vmatrix}
\text{diag}(1) & 0 \\
\frac{\partial C}{\partial m} \frac{\partial m}{\partial x_A} & \frac{\partial C}{\partial x_B}
\end{vmatrix}
= \left| \frac{\partial C}{\partial x_B} \right|
\]

- \( m \) piecewise polynomial function, with NN adopted parameters

\[ \sim \text{very expressive non-linear variable transformations (non-factorisable)} \]
Neural Importance Sampling

improve sampling through Normalizing Flows – cont’d

\[ \Gamma_{t \rightarrow b e^- \nu_e} \]

\[ \sigma_{e^+ e^- \rightarrow t[be^+ \nu_e] \bar{t}[\bar{b} e^- \bar{\nu}_e]} \]

- 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]

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

\[
= \int_{0}^{1} \prod_{j=1}^{N-1} \frac{U^{\nu - (L+1)D/2}}{F^{\nu - LD/2}} = \int_{0}^{1} \prod_{j=1}^{N-1} dx_j \mathcal{I}(\vec{x})
\]

∽ analytic continuation in complex plane, closed integration contour

\[
0 = \oint_{c} \prod_{j=1}^{N} d\vec{z}_j \mathcal{I}(\vec{z}) = \int_{0}^{1} \prod_{j=1}^{N} dx_j \mathcal{I}(\vec{x}) + \int_{\gamma} \prod_{j=1}^{N} d\vec{z}_j \mathcal{I}(\vec{z})
\]

⇔ \[
\int_{0}^{1} \prod_{j=1}^{N} dx_j \mathcal{I}(\vec{x}) = - \int_{\gamma} \prod_{j=1}^{N} d\vec{z}_j \mathcal{I}(\vec{z}) = \int_{0}^{1} \prod_{j=1}^{N} dx_j \det \left( \frac{\partial \vec{z}(\vec{x})}{\partial \vec{x}} \right) \mathcal{I}(\vec{z}(\vec{x}))
\]
Neural Importance Sampling – spin offs

numerical evaluation of multi-loop Feynman integrals


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

1. Contour deformation: used if multi-scale integral

\[
\int_{\gamma} \prod_{j=1}^{N} dy_j I(y) \\
\frac{\partial I(y)}{\partial y_j} = 0
\]

2. \(\lambda\)-glob: optimization of \(\lambda_j\) parameters

\[
\int_{\gamma} \prod_{j=1}^{N} dz_j I(z) \\
\frac{\partial I(z)}{\partial y_j} = 0
\]

3. Normalizing flow: remapping of reals

\[
\int_{\gamma} \prod_{j=1}^{N} dx_j I'(x) \\
\frac{\partial I'(x)}{\partial y_j} = 0
\]

- achieve reduction of numerical uncertainties
- relevant for high-precision LHC calculations
Neural Importance Sampling – spin offs

numerical evaluation of multi-loop Feynman integrals


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

\[ D_0 = e^{3\epsilon \gamma E} \int \frac{d^d l_1 d^d l_2 d^d l_3}{\pi^{3d/2}} \frac{(q^2)^{5-3d/2}}{\ell_1^2 \ell_2^2 \ell_3^2 (\ell_{13} - q)^2 (\ell_{23} - q)^2} \]

- i-flow can outperform Vegas
  \( \sim \) higher dimensions, complexity
  \( \sim \) approaches precision target earlier
- branch out in other communities
Other sampling ideas

Exploring phase space with Nested Sampling

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

Unweighting efficiency $gg \rightarrow Ng$
Surrogate Models
Surrogate Unweighting

Employ fast ME × PS surrogates for event unweighting

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

\[
d\sigma_{ab\rightarrow n}\big|_{p_a, p_b, \{p_i\}} = f_a(x_a, \mu_F) f_b(x_b, \mu_F) \left| \mathcal{M}_{ab\rightarrow n} \right|^2 J_{\Phi_n} \left| d\Phi_n \right|_{p_a, p_b, \{p_i\}}
\]

approximate by surrogate, e.g. from NN

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

```plaintext
while true do
    generate phase-space point \( u \);
    calculate approximate event weight \( s \);
    generate uniform random number \( R_1 \in [0, 1) \);
    # first unweighting step
    if \( s > R_1 \cdot w_{\text{max}} \) then
        calculate exact event weight \( w \);
        determine ratio \( x = w/s \);
        generate uniform random number \( R_2 \in [0, 1) \);
        # second unweighting step
        if \( x > R_2 \cdot x_{\text{max}} \) then
            return \( u \) and \( \tilde{w} = \max(1, s/w_{\text{max}}) \cdot \max(1, x/x_{\text{max}}) \)
        end
    end
end
```
Surrogate Unweighting

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

- $gg\to e^- e^+ gdd\bar{d}$

\[ N = 1 \text{M events} \]

\[ x_{\text{max}}^{\text{med}} = 72.9 \]

\[ x_{\text{max}}^\text{med} = 26.6 \]

- $N_{\text{eff}} := \frac{\left( \sum_i \tilde{w} \right)^2}{\sum_i \tilde{w}^2} = \alpha N$

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

\[ \frac{N(0,1)}{\sqrt{\Delta_{\text{Full}}^2 + \Delta_{\text{Surrogate}}^2}} \]

\[ \Rightarrow \text{fully compatible samples} \]
Surrogate Unweighting

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

<table>
<thead>
<tr>
<th></th>
<th>$gg \to t\bar{t}ggg$</th>
<th>$ug \to t\bar{t}ggu$</th>
<th>$uu \to t\bar{t}guu$</th>
<th>$u\bar{u} \to t\bar{t}gd\bar{d}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\epsilon_{\text{full}}$</td>
<td>1.1e−2</td>
<td>7.3e−3</td>
<td>6.8e−3</td>
<td>6.6e−4</td>
</tr>
<tr>
<td>$\epsilon_{1\text{st,surr}}$</td>
<td>8.7e−3</td>
<td>5.8e−3</td>
<td>4.7e−3</td>
<td>3.6e−4</td>
</tr>
<tr>
<td>$\langle t_{\text{full}} \rangle / \langle t_{\text{surr}} \rangle$</td>
<td>39312</td>
<td>2417</td>
<td>199</td>
<td>64</td>
</tr>
<tr>
<td>$\chi_{\text{max}}^{\text{p.m.}}$</td>
<td>52.03</td>
<td>32.52</td>
<td>69.76</td>
<td>326.19</td>
</tr>
<tr>
<td>$\epsilon_{2\text{nd,surr}}^{\text{p.m.}}$</td>
<td>2.4e−2</td>
<td>3.8e−2</td>
<td>2.1e−2</td>
<td>5.6e−3</td>
</tr>
<tr>
<td>$\alpha^{\text{p.m.}}$</td>
<td>0.9989</td>
<td>0.9984</td>
<td>0.9994</td>
<td>0.9981</td>
</tr>
<tr>
<td>$f_{\text{eff}}^{\text{p.m.}}$</td>
<td>2.21</td>
<td>4.89</td>
<td>1.47</td>
<td>0.19</td>
</tr>
<tr>
<td>$\chi_{\text{max}}^{\text{med}}$</td>
<td>30.40</td>
<td>19.14</td>
<td>27.78</td>
<td>25.34</td>
</tr>
<tr>
<td>$\epsilon_{2\text{nd,surr}}^{\text{med}}$</td>
<td>4.3e−2</td>
<td>6.4e−2</td>
<td>5.1e−2</td>
<td>7.1e−2</td>
</tr>
<tr>
<td>$\alpha^{\text{med}}$</td>
<td>0.9983</td>
<td>0.9966</td>
<td>0.9943</td>
<td>0.9321</td>
</tr>
<tr>
<td>$f_{\text{eff}}^{\text{med}}$</td>
<td>3.90</td>
<td>8.26</td>
<td>3.91</td>
<td>2.22</td>
</tr>
</tbody>
</table>
Surrogate Unweighting – dedicated ME emulation methods provide use case for improved ME emulators

- factorisation-aware ME emulator [Maître and Truong, JHEP 11 (2021), 066]
  \[ \langle |M_{n+1}|^2 \rangle = \sum_{\{ijk\}} C_{ijk} D_{ij,k} \]

- alternative approaches focused on loop-induced channels
  [Aylett-Bullock, Badger and Moodie, JHEP 08 (2021), 066]
  [Badger et al., arXiv:2206.14831 [hep-ph]]
Surrogate Unweighting – dedicated ME emulation methods

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
  \( \leadsto \) find significant improvements compared to naive NN setup
  \( \leadsto pp \rightarrow Z + 4j \) channel: \( f_{\text{med,naive}}^{\text{eff}} = 4.7 \rightarrow f_{\text{med,dipole}}^{\text{eff}} = 15 \)
Novel Tuning Strategies
Non-perturbative models

- modelling of non-perturbative phenomena corner stone of HEP MCs
- 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
- grid search not feasible, parametrize/model MC response
- Professor [Hoeth et al.] & Apprentice [Krishnamoorthy et al.] tools

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

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

[Chahal and Krauss, SciPost Phys. 13 (2022) no.2]
Non-perturbative models – tuning

Hadronization tuning using Bayesian optimization

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

- Bayesian optimization standard strategy for (expensive) black-box functions
  \[ \sim \] 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
  \[ \sim \] non-parametric, statistical interpretation/uncertainties
- promising feasibility study, closure test with Pythia Monash tune [Skands 2014]
  \[ \sim \] model global $\chi^2$-measure of MC vs. Monash

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Monash Value</th>
<th>Tune Value</th>
<th>Range Considered</th>
</tr>
</thead>
<tbody>
<tr>
<td>sigma</td>
<td>0.335</td>
<td>0.333±0.002</td>
<td>[0, 1]</td>
</tr>
<tr>
<td>bLund</td>
<td>0.98</td>
<td>1.04±0.02</td>
<td>[0, 2]</td>
</tr>
<tr>
<td>aExtraSQuark</td>
<td>0</td>
<td>0.07±0.0</td>
<td>[0, 2]</td>
</tr>
<tr>
<td>aExtraDiQuark</td>
<td>0.97</td>
<td>1.48±0.15</td>
<td>[0, 2]</td>
</tr>
<tr>
<td>rFactC</td>
<td>1.32</td>
<td>1.38±0.06</td>
<td>[0, 2]</td>
</tr>
<tr>
<td>rFactB</td>
<td>0.855</td>
<td>0.887±0.015</td>
<td>[0, 2]</td>
</tr>
</tbody>
</table>
Non-perturbative models – tuning

Machine Learning parameter responses

- learn the inverted model of a generator (here Pythia8)
  ~ 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]]
possible HEP MC development perspectives

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

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

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

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