Efficient generation of unweighted events with matrix element surrogates

ICHEP 2024 @ Prague

Timo Janßen

Campus-Institut Data Science (CIDAS), Georg-August-Universität Göttingen

with Katharina Danziger, Tim Herrmann, Daniel Maître, Steffen Schumann, Frank Siegert, Henry Truong

19 July 2024

GEORG-AUGUST-UNIVERSITÄT GÖTTINGEN IN PUBLICA COMMODA SEIT 1737

TECHNISCHE UNIVERSITÄT DRESDEN

KISS

GEFÖRDERT VOM Bundesministerium für Bildung und Forschung

- Problem: unweighted event generation is resource intensive
- becomes worse at higher partonic multiplicity (→ HL-LHC)
- 💡 save resources by using NN matrix element surrogates
- 🚀 for $ug \to t\bar{t}gggu$ ($t\bar{t}$ + 4 jets) a 4-layer NN is 200.000 times faster than a full colour-summed calculation!
- Problem solved?

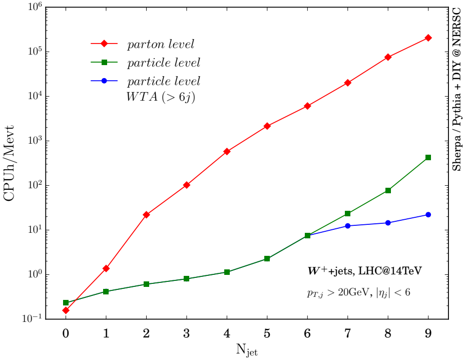
  → No! Results are biased due to approximation errors!

Questions

How to get rid of the bias?
How much can we actually gain?

Motivation

multi-jet merged calculation

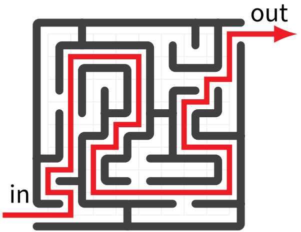


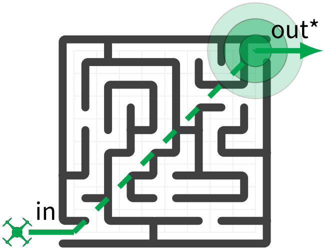
- computing time for unweighted events scales exponentially with multiplicity
- two factors contribute:
  (1) matrix elements become more expensive
  (2) unweighting efficiency goes down (curse of dimensionality)
- to deal with (2) need to improve phase space sampler, e.g. using normalizing flows

But what about the ME evaluation time?

- we have to evaluate the ME for each trial event in unweighting
- if unw. eff. is small, ME evaluation time is a bottleneck

Idea





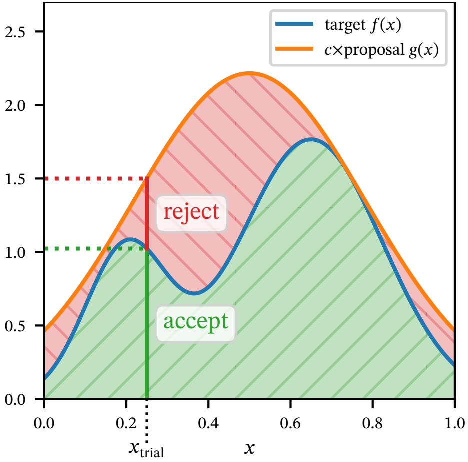
Goal:

increase the number of unweighted events per second

Idea:

- reduce event generation time by reducing the number of calls to the ME
  → use a fast & accurate surrogate (NN)
- correct all errors from the approximation in a 2nd unweighting step
  → method is unbiased by design

Unweighted event generation explained



- Monte Carlo sampled events come with weights w
- we often want to work with unweighted events

  → produce unit-weight events by unweighting

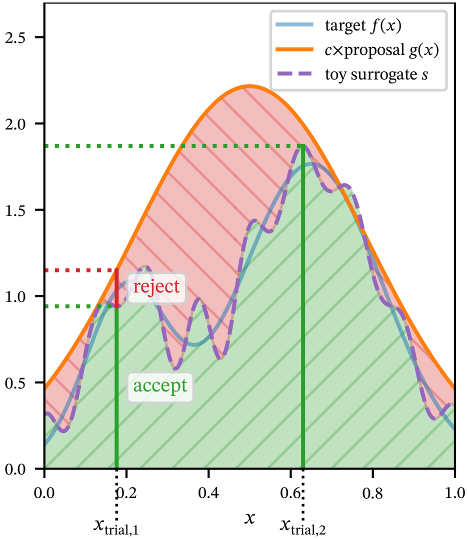
  → deliberately reduce sample size while retaining most of its statistical power
- accept events with probability $p_{\text{accept}} = \frac{f(x)}{cg(x)}$, where $c \geq w_{\text{max}}$

unweighting efficiency

$$\eta = \frac{\langle w \rangle}{c}$$

→ ratio between #accepted and #trials

Unweighting against a surrogate function

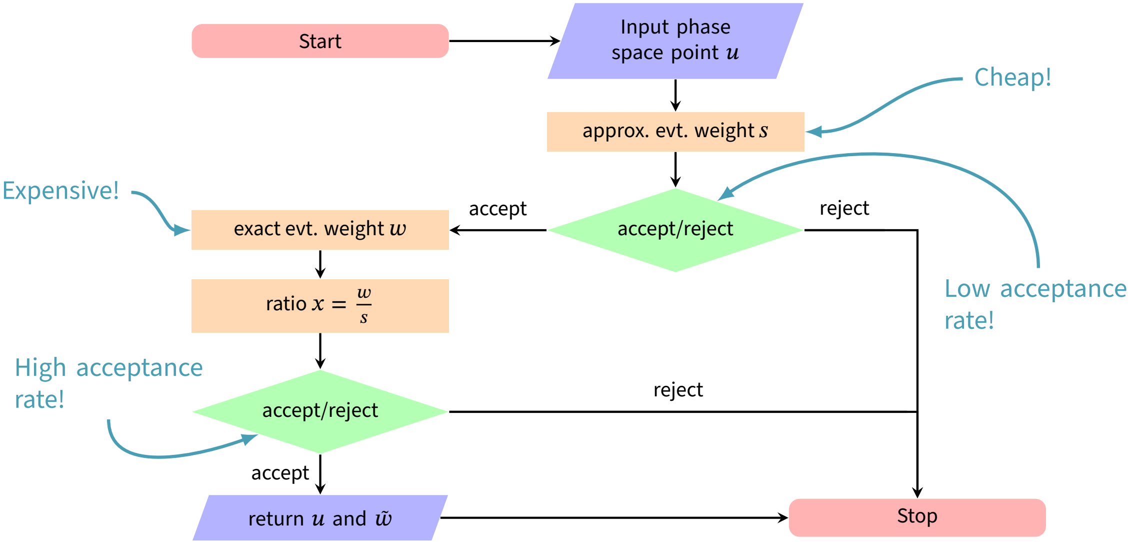


- replace target $f(x)$ by surrogate $s(x)$
- accept/reject probabilities are slightly wrong

  → events get correction weights $x = \frac{w}{s} = \frac{f}{gs}$
- 2nd unweighting step leads to fully unweighted event sample
- total efficiency: $\eta_1 \cdot \eta_2 = \frac{\langle s \rangle}{s_{\text{max}}} \cdot \frac{\langle x \rangle}{x_{\text{max}}}$
- trick: eval. true weight only for those events accepted in the 1st step!
- if $f(x)$ is expensive and $s(x)$ is a good approximation, we can save a lot of time

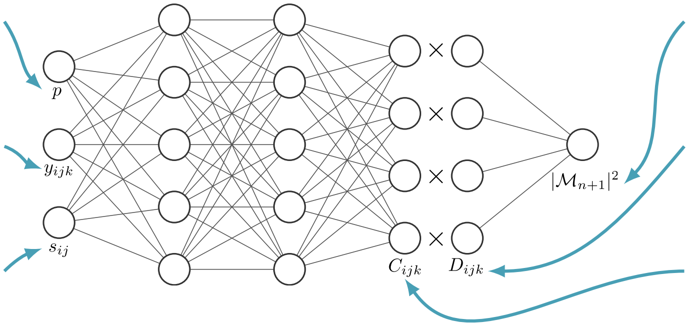
K. Danziger, TJ, S. Schumann, F. Siegert: SciPost Phys. 12, 164 (2022)

Surrogate unweighting algorithm



K. Danziger, TJ, S. Schumann, F. Siegert: SciPost Phys. 12, 164 (2022)

Factorization-aware matrix element emulation



soft/collinear factorization properties

$$|\mathcal{M}_{n+1}|^2 \to |\mathcal{M}_n|^2 \otimes \mathbf{V}_{ijk}$$

[Catani, Seymour Nucl.Phys. B485 (1997) 291-419]

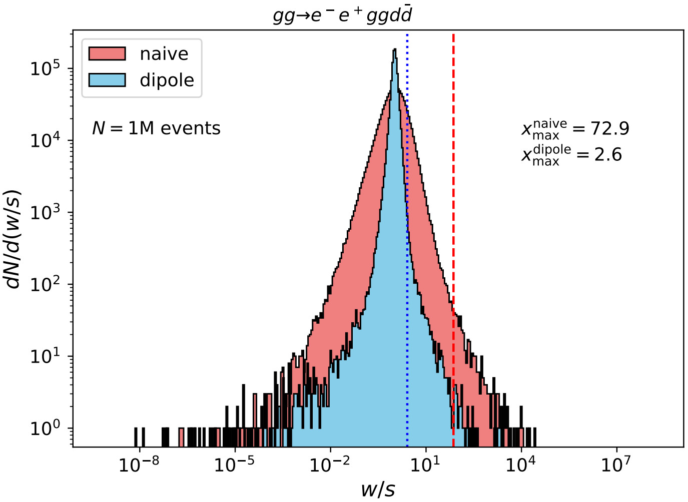
Ansatz

$$\langle|\mathcal{M}|^2\rangle \approx \sum_{\{ijk\}} C_{ijk} D_{ijk}$$

D. Maître, H. Truong: JHEP 11 (2021) 066

Factorisation-aware matrix element emulation

Comparison with naive (non-dipole) model for $Z + 4j$:

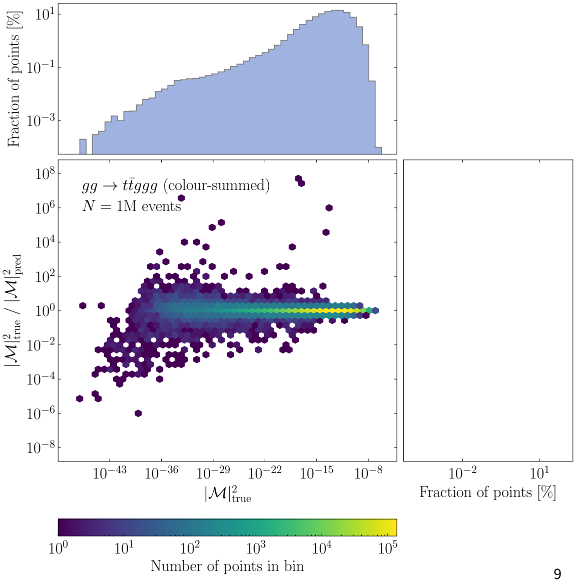


Comparison of eval time:
$\frac{t_{\text{AMEGIC}}}{t_{\text{dipole}}} \approx 388$

TJ, D. Maître, S. Schumann, F. Siegert, H. Truong: SciPost Phys. 15, 107 (2023)

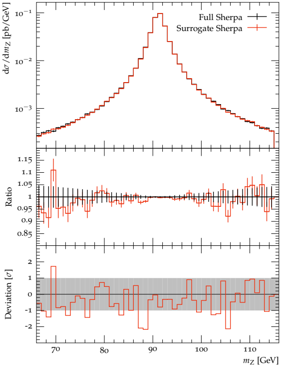
Results: distribution of weights for $t\bar{t}$ + 3jets

- MEs reproduced accurately
- especially in the regions where ME is largest (highest acceptance probabilities)
- most outliers heavily suppressed in $|\mathcal{M}|^2$
  → deal with them via partial unweighting

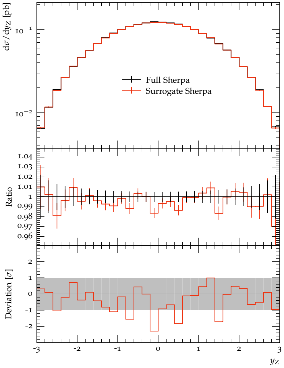


TJ, D. Maître, S. Schumann, F. Siegert, H. Truong: SciPost Phys. 15, 107 (2023)

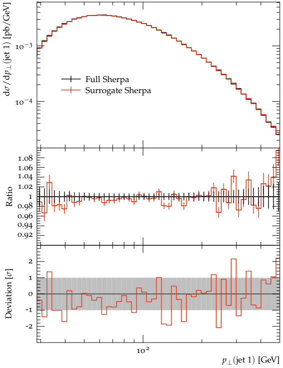
Results: validation plots



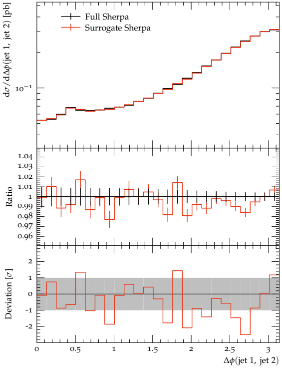
Z mass



Z rapidity



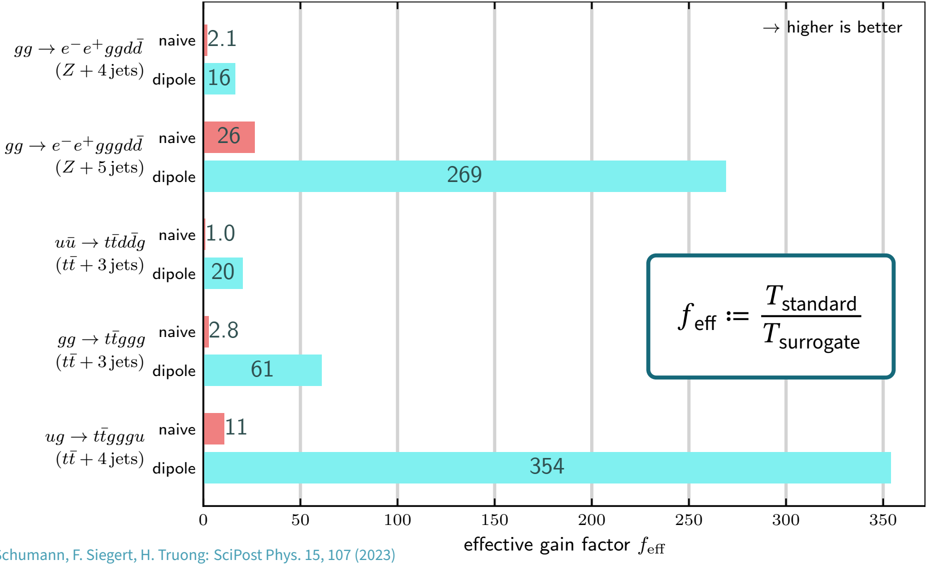
leading jet $p_\perp$



$\Delta\phi$(jet 1, jet 2)

Results: effective gain factors for LHC multi-jet processes

Using 1M training events:



TJ, D. Maître, S. Schumann, F. Siegert, H. Truong: SciPost Phys. 15, 107 (2023)

Outlook

- extend to more realistic settings:
  → deal with hadronic processes (groups of partonic channels)
  → multijet merged
- new baseline: COMIX colour-summed mode
- tune NN architecture (hyperparameter optimization)
- Is it also worthwhile at NLO?

Summary

This talk ...

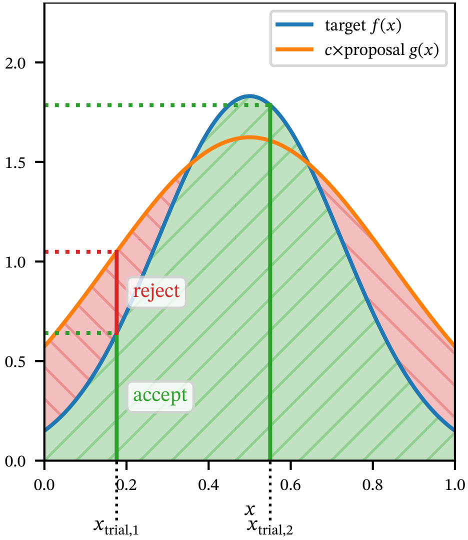
- introduced a generic & unbiased method to speed up unweighted event generation using fast & accurate surrogates
- showed that the factorisation-aware ME emulation model is very accurate for colour-summed MEs (incl. hadronic initial states & massive quarks)
- demonstrated that large gain factors can be achieved for unweighting of colour-summed MEs in relevant LHC processes

Conclusion

More physics using the same resources!

Backup Slides

Unweighting in two steps / partial unweighting



- let's allow $c < w_{\text{max}}$ so that p_{accept} >= 1 in some regions
- if accepted, events with $p_{\text{accept}} < 1$ get unit weights
- events with $p_{\text{accept}} >= 1$ are always accepted

  $\rightarrow$ these events get overweights $x = \frac{w}{s}$
- There is no bias as long as we take the overweights into account!
- can still produce fully unweighted sample by unweighting against overweights x, respecting their maximum x_{max}

Implementation details

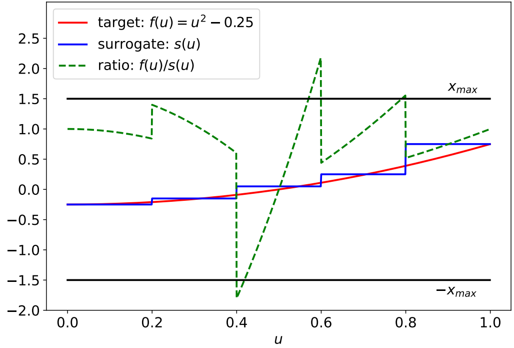
- for NN evaluation use ONNX Runtime with all possible optimisations
- two step unweighting implemented in SHERPA 2 [Bothmann et al. SciPost Phys. 7, 034 (2019)] and SHERPA 3
- CPU single threaded

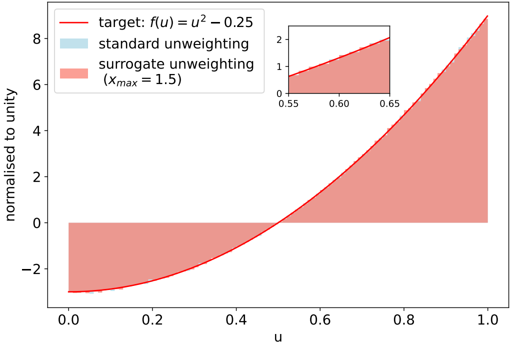
  → use in existing workflows without changes

  → apply to vectorised workflows for even better performance
- ME generator: AMEGIC [Krauss et al. JHEP 02 (2002) 044]
- we evaluate the performance for processes that are very important for the LHC: V+jets & $t\bar{t}$+jets

Going to NLO

- at NLO the weight function can become negative
- unweighting produces events with weight ± 1
- NN will happily output negative predictions but unweighting needs to be adapted
- use single maximal weight: $w_{\text{max}} = |w|_{\text{max}} > 0$
- surrogate may predict wrong sign → signed overweights





K. Danziger, TJ, S. Schumann, F. Siegert: SciPost Phys. 12, 164 (2022)