Negative weight workshop summary

Yannick Ulrich
(he/him)

U Liverpool

Reminder: what are neg. weight & why do we care?

Event generators may produce events with $w_i < 0$ b/c

- pert. exactness $\rightsquigarrow \int d\sigma \stackrel{!}{=} \sigma_{NLO}$ often involves CT
  $\rightsquigarrow w < 0$

- in showers, if shower overestimates Real matrix element, $w_i < 0$ to counter that

Why is this a problem?

If $r \cdot N$ out of N events are neg., statistics needs increase by $\frac{1}{(1-2r)^2}$

solutions: - (any of the following)

- fast simulations

KrKNLO (James Whitehead)

finite part of PDF renorm. has scheme dependence @ NLO and beyond

⇝ change this to avoid negative weights

$$w \to w\left[1+\frac{\alpha}{\pi}\left(\frac{V}{B}+\frac{I}{B}+\Delta^{FS}\right)\right]$$

↑ large enough

KrkNLO

KrkNLO matching for colour-singlet processes

Pratixan Sarmah,[a] Andrzej Siódmok[a] and James Whitehead[a,b]

[a] Jagiellonian University, ul. prof. Stanisława Łojasiewicza 11, Kraków 30-348, Poland
[b] Institute of Nuclear Physics, Polish Academy of Sciences, ul. Radzikowskiego 152, Kraków 31-342, Poland

E-mail: pratixan.sarmah@doctoral.uj.edu.pl, andrzej.siodmok@uj.edu.pl, james.whitehead@uj.edu.pl

Abstract: Matched calculations combining perturbative QCD with parton showers are an indispensable tool for LHC physics. Two methods for NLO matching are in widespread use: Mc@Nlo and Powheg. We describe an alternative, KrkNLO, reformulated to be easily applicable to any colour-singlet process. The primary distinguishing characteristic of KrkNLO is its use of an alternative factorisation scheme, the 'Krk' scheme, to achieve NLO accuracy. We describe the general implementation of KrkNLO in Herwig 7, using diphoton production as a test process. We systematically compare its predictions to those produced by Mc@Nlo with several different choices of shower scale, both truncated to one-emission and with the shower running to completion, and to Atlas data from LHC Run 2.

Keywords: Higher-Order Perturbative Calculations; Parton Shower

ArXiv ePrint: 2409.16417

JHEP01(2025)062

Key ideas:

→ change PDF factorisation scheme ('Krk' scheme: exploit ambiguity!)

→ matching becomes multiplicative

→ no subtraction: weights all positive

based on work with Staszek Jadach, Andrzej Siódmok, Pratixan Sarmah

ESME in PanScales (Alex Karlberg)

idea: can't avoid negative weights & discarding changes N^kLO exactness

⇝ push problem to $N^{k+1}LO$

A Sudakov alogorithm

We want to transform the problem, such that we are guaranteed that negative weights are only associated with terms beyond $\mathcal{O}(\alpha_s)$.

Exponentiated subtraction

- Start with normalisation $n_B = 1$ and the max value for k_t (ordering variable) Run a loop while $k_t > k_{t,min}$:
  - Sample $\ln k_t$ from $e^{-M/B_0 \ln 1/k_t}$, where $M = \max(R, C)$,
  - generate random number $0 < r < 1$
  - if $r < \frac{|R-C|}{M}$ then
    → if $R > C$: $n_B \to n_B + 1$
    → if $C > R$: $n_B \to n_B - 1$
- return n_B ↖ only addresses problems of over estimate

$$\langle n_b \rangle = 1 + \int [R(\Phi) - C(\Phi)]/B_0 \, d\Phi_{rad}$$

ARCANE (Prasanth Shyamsundar)

define $F(V) = \int dL\, F(V, L) > 0$ but $F(V, L) \not> 0$

obs. property ↗ (V) ↖ latent property eg. history (L)

ARCANE reweighting

- Under ARCANE reweighting, the density of (V, L) is modified as:

$$F^{\text{(arcane)}}(V, L) = F(V, L) + G(V, L)$$

$$\text{where } G \text{ satisfies} \quad \int \mathrm{d}L\, G(V, L) = 0.$$

- This way $F^{\text{(arcane)}}(V) \overset{\text{exactly}}{=} F(V)$.
- G redistributes the contributions of different (V, L)-s to the same V, to try and get $F^{\text{(arcane)}}(V, L)$ to have the same sign as $F(V)$.

cmp. redistr. across history

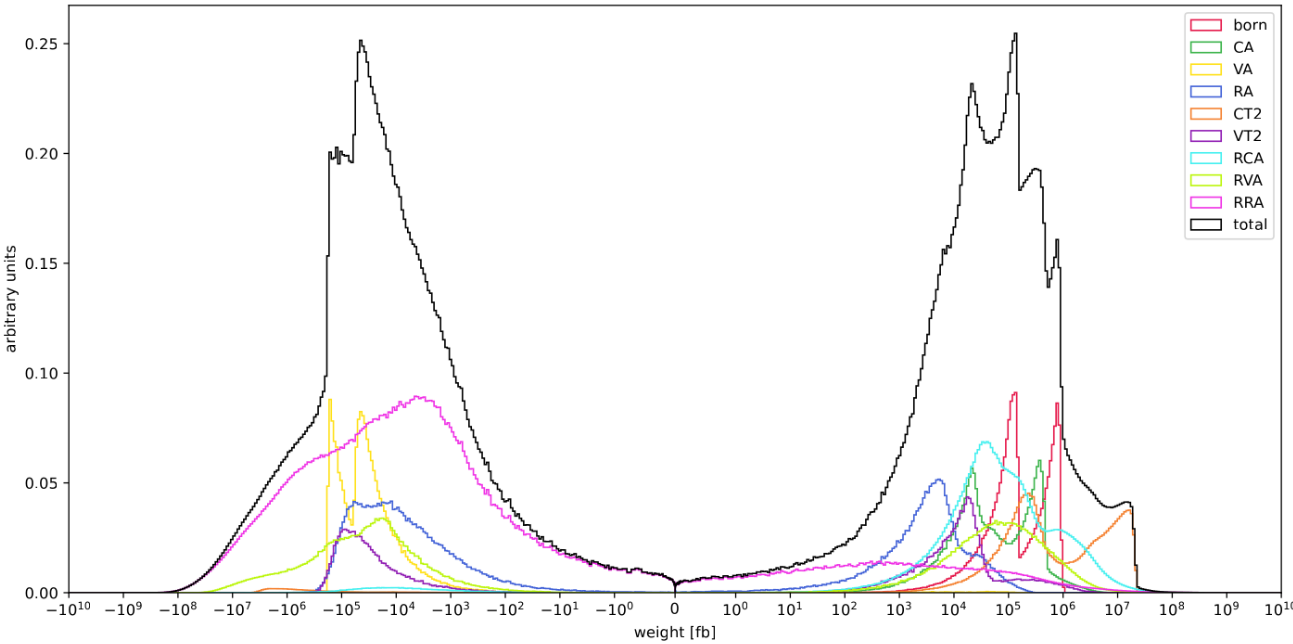
Two questions:

1) How does one construct the redistribution function G?
2) How does one sample events from $F^{\text{(arcane)}}(V, L)$.

MATRIX event generation (Oleksandr Zenaiev)

PDF fits ~ event generation with MATRIX for $pp \to t\bar{t}$

$t\bar{t}$ @ NNLO in MATRIX: weights

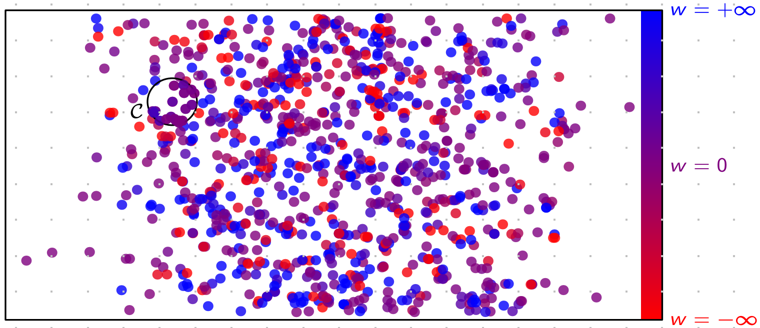


- Largest (positive and negative) weights: RRA $gg \to t\bar{t}gg$ and CT2 $gg \to t\bar{t}$
- These are most time consuming contributions (also RVA $gg \to t\bar{t}g$ which is slow itself)

Cell resampling: a strategy to remove neg weights
(Andreas Maier)

Cell Resampling

[Andersen, Gütschow, Maier, Prestel 2020; Andersen, Maier 2021]



delicate point!

Cell resampling:

1. Choose seed event with negative weight for cell $\mathcal{C}$
2. Iteratively add nearest event to cell until $\sum_{i \in \mathcal{C}} w_i \geq 0$ or radius exceeds $r_{\max}$
3. Redistribute weights, e. g. average over cell: $w_i \to w = \frac{\sum_{j \in \mathcal{C}} w_j}{\text{\# events in } \mathcal{C}}$
4. R e p e a t

... in QED using McMule ($\gamma\mu$)

$r \times N$ events have $w < 0$

Note $\alpha_{\text{QED}} \ll \alpha_s \Rightarrow$ need many more events

Order	ξ_c	N	r	$[w_{\min}, w_{\max}]/\langle w\rangle$	feature
LO	n/a	99.7M	0.0%	$[0.03, 7.7]$	
NLO	0.1	220M	0.5%	$[-3.6, 3.6] \times 10^7$	
			0.1%	$[-1.3, 19] \times 10^3$	cres
		51M	0	$[+0.0, 1.6] \times 10^4$	subs
	1.0	216M	3.5%	$[-2.1, 2.1] \times 10^6$	
			0.3%	$[-7.0, 26] \times 10^3$	cres
		61M	0	$[+0.0, 2.5] \times 10^4$	subs
NNLO	0.1	21.1G	0.7%	$[-8.6, 8.6] \times 10^7$	
		3.6G	0.1%	$[-6.9, 6.9] \times 10^5$	cres
	1.0	19.8G	9.2%	$[-2.3, 2.3] \times 10^8$	
		1.2G	0.5%	$[-3.3, 2.4] \times 10^4$	cres

significant reduction in r

non-pos. integrands with normalising flow (Rene Poncelet)

treat pos. & neg. part separately for generation by learning them

Stratification of signed integrands

There are ways around:

1) Add a large constant

2) Stratification: $f(\mathbf{x}) = f_+(\mathbf{x}) + f_-(\mathbf{x})\,, \quad \text{with} \quad f_\pm(\mathbf{x}) = \Theta(\pm f(\mathbf{x}))f(\mathbf{x})$

learn those in two networks

$$I = \int_{\mathbf{H}_+(\mathbf{x})\in\Omega} \mathrm{d}\mathbf{H}_+ \frac{f_+(\mathbf{x})}{h_+(\mathbf{x})} + \int_{\mathbf{H}_-(\mathbf{x})\in\Omega} \mathrm{d}\mathbf{H}_- \frac{f_-(\mathbf{x})}{h_-(\mathbf{x})}$$

"two independent integrals"

$$\hat{I}_{\text{strat}} = \hat{I}_+ + \hat{I}_- = \frac{1}{N_+}\sum_{i=1}^{N_+}\frac{f_+(\mathbf{x}_i)}{h_+(\mathbf{x}_i)} + \frac{1}{N_-}\sum_{i=1}^{N_-}\frac{f_-(\mathbf{x}_i)}{h_-(\mathbf{x}_i)}$$

$$\delta\hat{I}_{\text{strat}} = \sqrt{\frac{1}{N-1}\left[\frac{N}{N_+}\text{Var}(\hat{I}_+) + \frac{N}{N_-}\text{Var}(\hat{I}_-)\right]}$$

$$\text{Var}(\hat{I}_\pm) = \frac{1}{N_\pm}\sum_{i=1}^{N_\pm}\left(\frac{f_\pm(\mathbf{x}_i)}{h_\pm(\mathbf{x}_i)}\right)^2 - \hat{I}_\pm^2$$

+ The total variance is now bounded by the individual variances

- The mappings are more complicated (need high phase space efficiency)

this is where normalising flow comes in

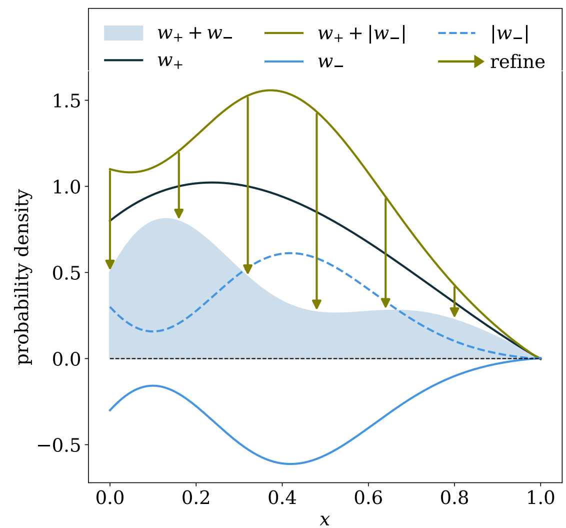
Neural refinement of weights (Dennis Noll)

Use NN to distinguish pos. & neg. event

⇝ use this to reweight

Weight Transformation: Refiner NEW!

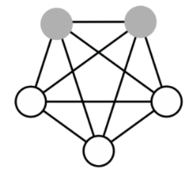
- Learn residual weight spectrum
- Train classification between:
  - Samples with positive weights w+
  - Samples with negative weights |w-|
- Application:
  - One refinement factor r per phase space x
  - $w_{new} = r \cdot |w_{old}|$



QCD & information theory (Ben Assi)

measure log moments of eg. thrust

QCD Theory meets Information Theory



Boltzman machine: Generative model samples according to stat dist

$$P(s) \sim e^{-E/T}, \quad E = -\sum_{i<j} w_{ij} s_i s_j - \sum_i \theta_i s_i$$

Boltzman factor: Maximum entropy solution subject to constraint on average energy

Prior — Sets Partition Function

$$p(x) = q(x) \exp\left[-\beta_0 - \beta_1 f_1(x) - \beta_2 f_2(x) - \dots \right]$$

Lagrange Multipler — Constrained Quantity

Sudakov structure: Plucking a random event-shape observable distribution

$$r(\tau)_{\mathrm{LL,f.c.}} = \frac{-2\alpha_s C_F}{\pi} \frac{\ln\tau}{\tau} \exp\left[-\frac{\alpha_s C_F}{\pi} \ln^2 \tau \right]$$

Sudakov factor

ML practitioner: Sudakov = Boltzmann factor and cusp AD = Lagrange multiplier that enforces constraint on the 2nd log moment of distribution

log moments not previously measured or calculated before in QCD!

BA, Höche, Lee, Thaler 2501.17219

Resampling for event generators

(Simon Plätzer)

(sorry missed the talk)

Conclusion

- This is a very important problem which needs fixing!
- Many ideas, some easier to implement some harder

BUT

do we need an event generator <u>here</u>?

⇝ simplified analysis