Variance reduction with Normalizing Flows

Rene Poncelet

LHCP 2026

18th– 22nd May 2026

THE HENRYK NIEWODNICZAŃSKI
INSTITUTE OF NUCLEAR PHYSICS
POLISH ACADEMY OF SCIENCES

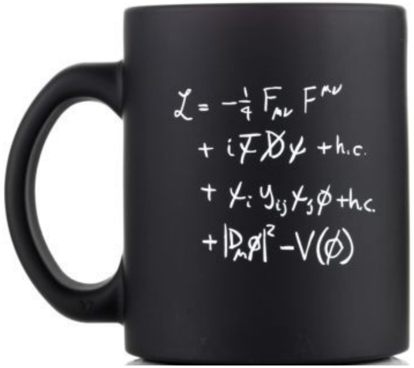
Funded by the European Union

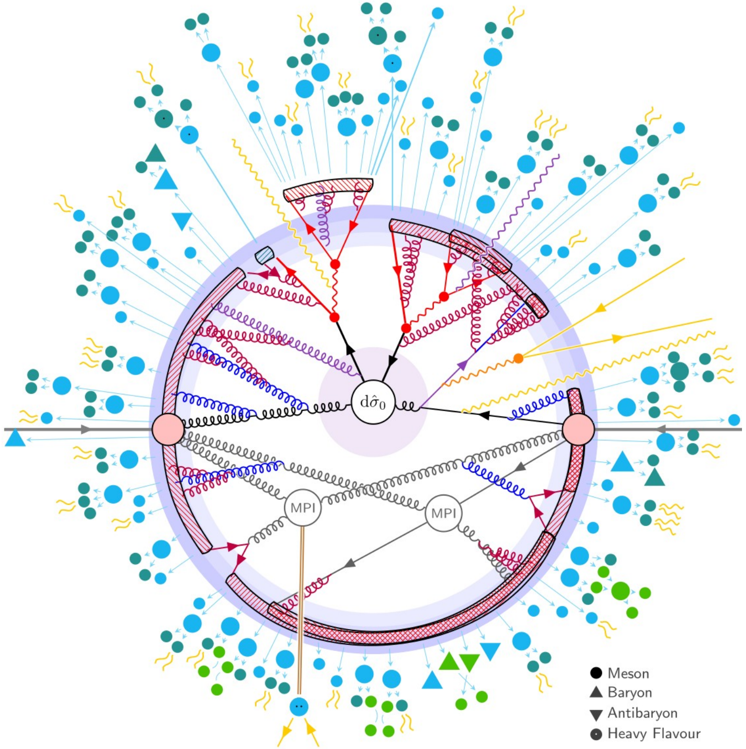
European Research Council
Established by the European Commission

NARODOWE CENTRUM NAUKI

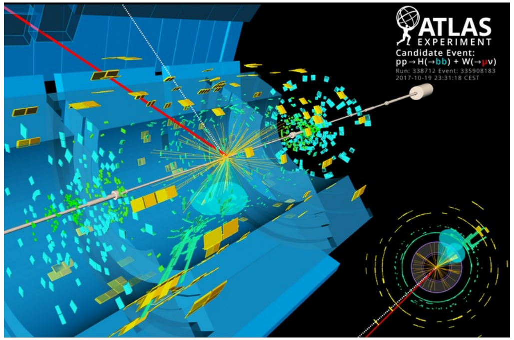
Monte Carlo Event Simulations

… the **bridge** between “theory world” and “real world”



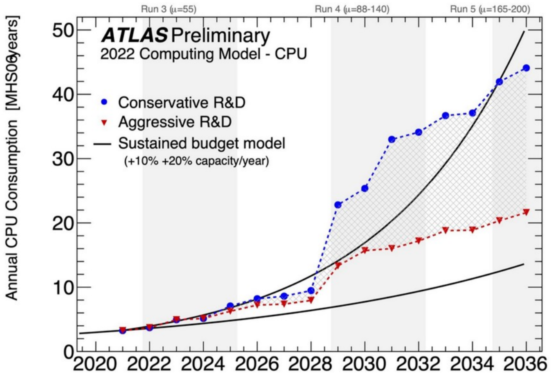


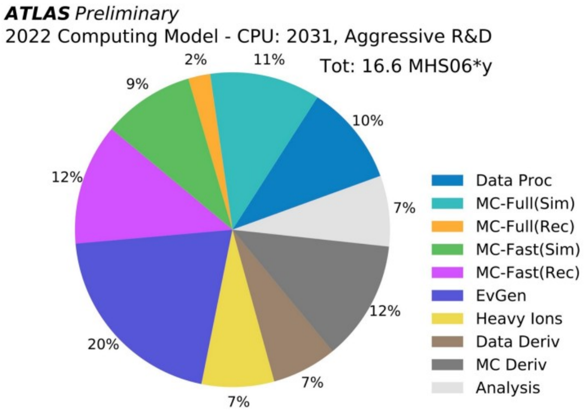
[Pythia 8.3]



The HL-LHC Computing Problem

[CERN-LHCC-2022-005]





Generating events is/will be computationally expensive

1) **negative events**

→ decrease in effective sample size

→ cumbersome down stream in analysis etc.

2) **expensive higher point ME unweighting**

bites twice:

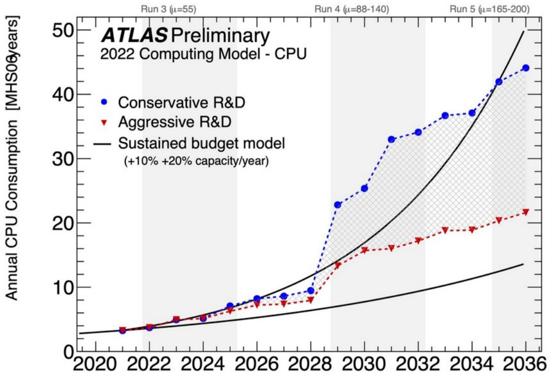
→ higher-point matrix elements are expensive

→ complex phase space reduces sample efficiency

The HL-LHC Computing Problem

Generating events is/will be computationally expensive
1) **negative events**
2) **expensive higher point ME unweighting**

[CERN-LHCC-2022-005]



Exploit modern architectures → CPU + GPU

- MEs: PEPPER [Bothmann, Childers, Giele, Höche, Isaacson Knobbe'23]
- Madgraph on GPUs [Valassi, Childers, Hageböck, Massaro, Mattelaer, Nichols, Optolowicz, Roiser, Teig, Wettersten'25]

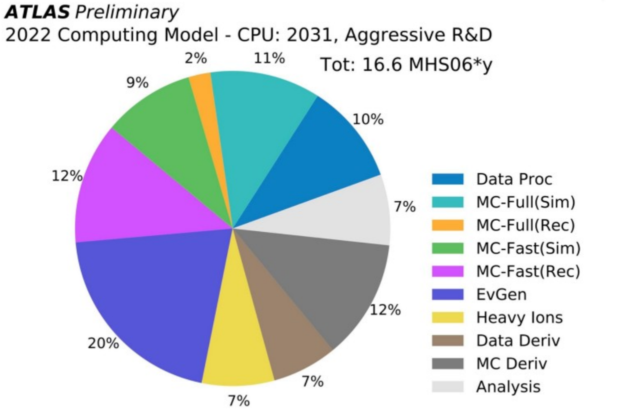
Ensure positivity of event weights

- Positive matching: ESME [PanScales'25], KrK [Pratixan, Siodmok, Whitehead '24]
- Reweighting methods: cell-resampling [J. Andersen, A. Maier'21+Maître'23] Arcane [Shyamsundar'25]

Machine Learning Methods

[De Crescenzo, Mariño, Elmer, Heimel, Plehn, Winterhalder, Zaro '26][Bothmann, Janßen, Knobbe, Schmitzer, Sinz'26][Jansen, Poncelet, Schumann '25]

- Neural Importance Sampling
- Surrogates to emulating MEs



Theory picture of hadron collision events

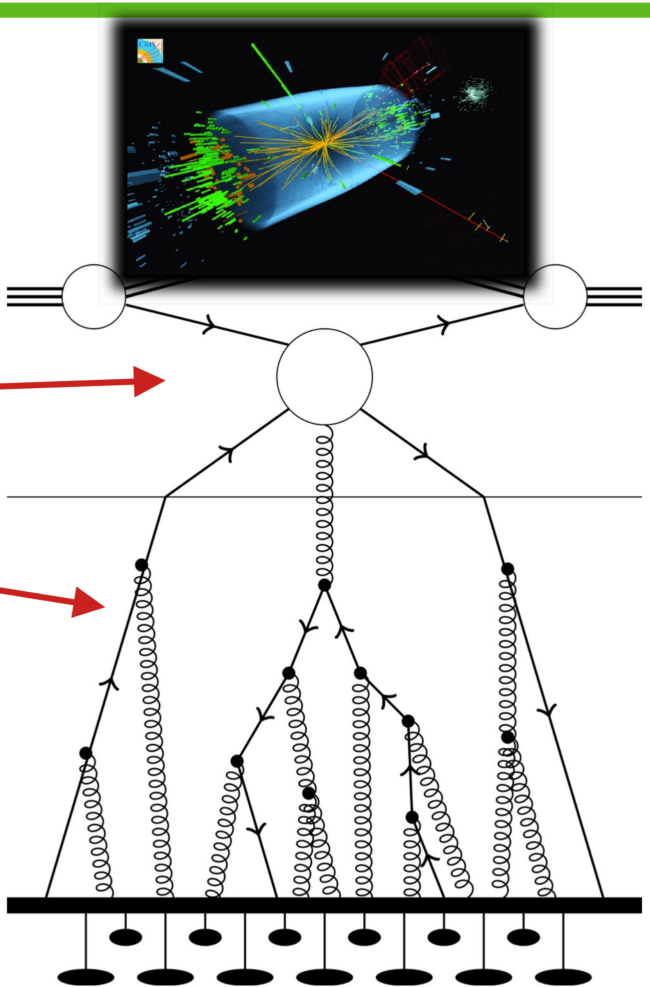
Guiding principle: factorization
"What you see depends on the energy scale"

In Quantum Chromodynamics (QCD):

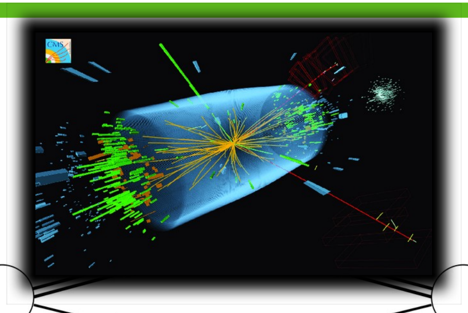
$Q \gg \Lambda_{\mathrm{QCD}}$ **Fixed-order perturbation theory**
scattering of individual partons

$Q \gtrsim \Lambda_{\mathrm{QCD}}$ **Parton-shower/Resummation**
all-order bridge between perturbative and non-perturbative physics

$Q \sim \Lambda_{\mathrm{QCD}}$ **"Hadronization"/MPI/...**
non-perturbative physics



Theory picture of hadron collision events



Guiding principle: factorization
"What you see depends on the energy scale"

In Quantum Chromodynamics (QCD):

$Q \gg \Lambda_{\mathrm{QCD}}$	**Fixed-order perturbation theory** scattering of individual partons
$Q \gtrsim \Lambda_{\mathrm{QCD}}$	**Parton-shower/Resummation** all-order bridge between perturbative and non-perturbative physics
$Q \sim \Lambda_{\mathrm{QCD}}$	**"Hadronization"/MPI/…** non-perturbative physics

Focus here:

- Higher-order cross sections
- "Unweighted" (matched) events are the seeds of parton-shower evolution

Neural importance sampling & surrogates

Variance reduction through importance sampling:

$$\hat{I} = \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x}_i)\,, \quad \delta\hat{I} = \sqrt{\frac{1}{N-1}\left(\frac{1}{N}\sum_{i=1}^{N} f^2(\mathbf{x}_i) - \hat{I}^2\right)}$$

$$I = \int_{\mathbf{H}(\mathbf{x})\in\Omega} \mathrm{d}\mathbf{H}\,\frac{f(\mathbf{x})}{h(\mathbf{x})} \qquad h(\mathbf{x}) = \left|\det\left(\frac{\partial \mathbf{H}(\mathbf{x})}{\partial \mathbf{x}}\right)\right|$$

Task: find $h(\mathbf{x})$adaptive MC techniques: VEGAS [Lepage'78], Parni [Hameren'14]

Neural importance sampling

Normalising Flows in Sherpa [Bothmann'20], Madgraph [Heimel'22'25],STRIPPER [Jansen'25]

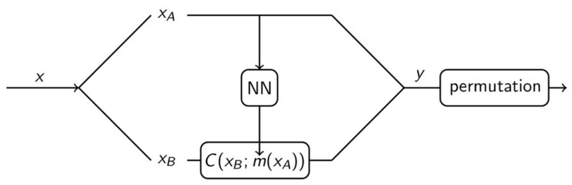
Surrogates

If f(x) is very expensive → finding a surrogate improves performance
→ can be unbiased by unweighting against real f(x)

- QCD factorization aware NN ME emulators [Maitre'21] [Janssen'23]
- two-stage unweighting to correct surrogates [Danziger'22]
- one-loop emulators [Aylett-Bullock'21][Badger'23] [Maitre'23]

Unweighting with Normalizing Flows

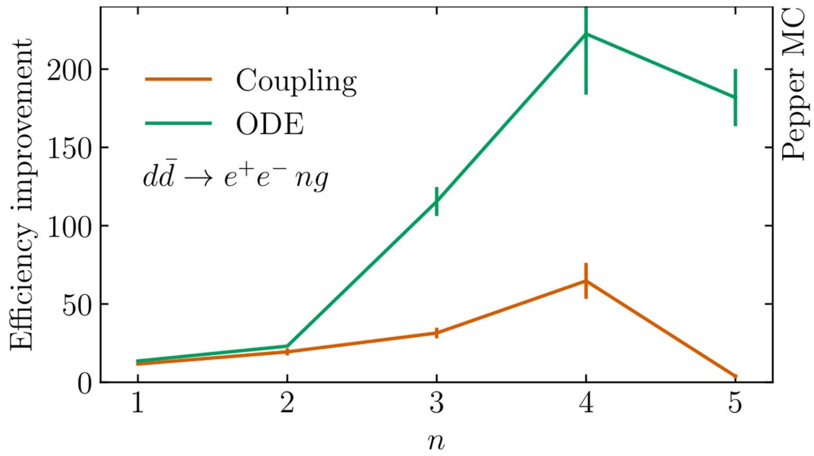
Coupling Layer Flows → series of simple bijections

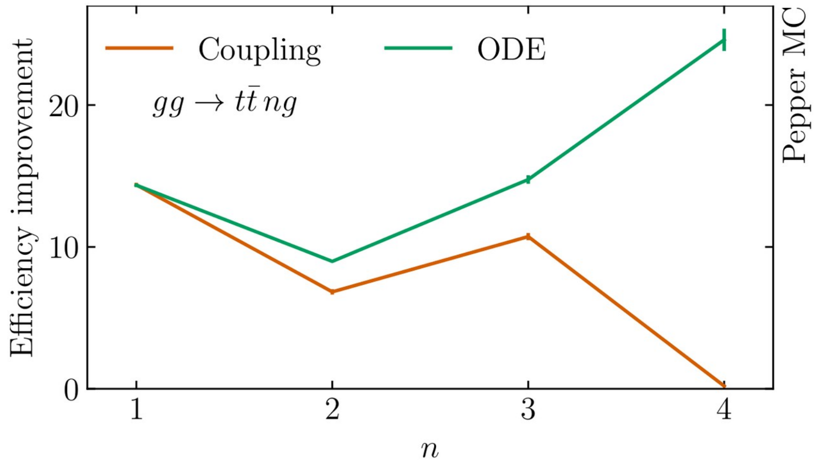


Continuos Normalising Flows $\frac{\mathrm{d}\psi_t(x)}{\mathrm{d}t} = v_t(\psi_t(x))$

Substantial unweighting efficiency improvements

$$\epsilon_\theta = \frac{\langle w_\theta \rangle}{w_{\max,\theta}} \quad \text{with} \quad w_\theta(x) := \frac{p_0(x)}{q_\theta(x)} .$$



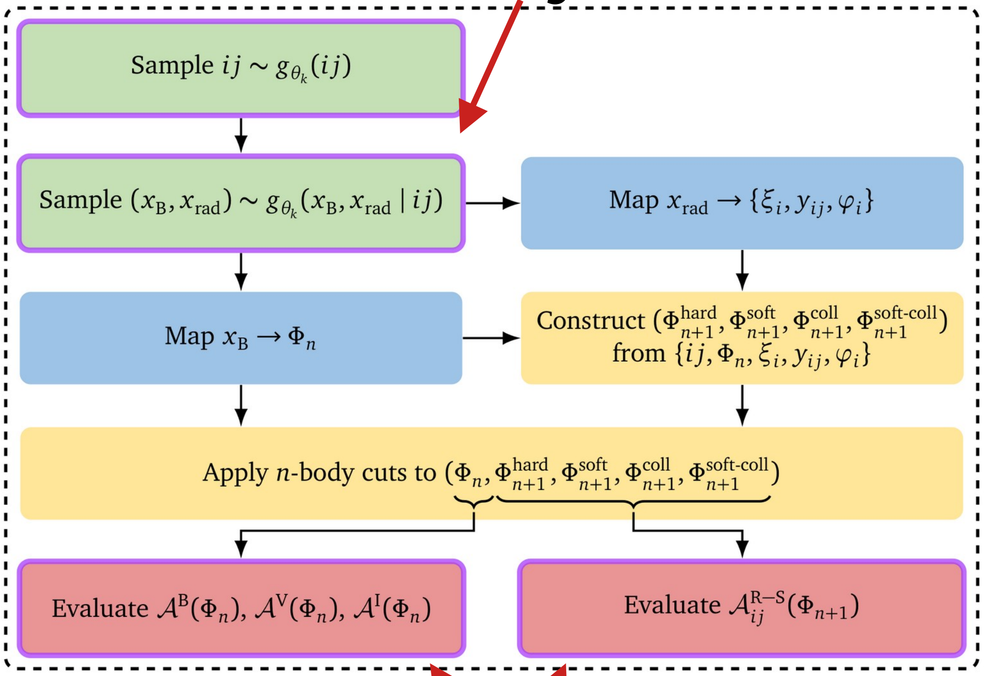


[Bothmann, Janßen, Knobbe, Schmitzer, Sinz, 2604.03511]

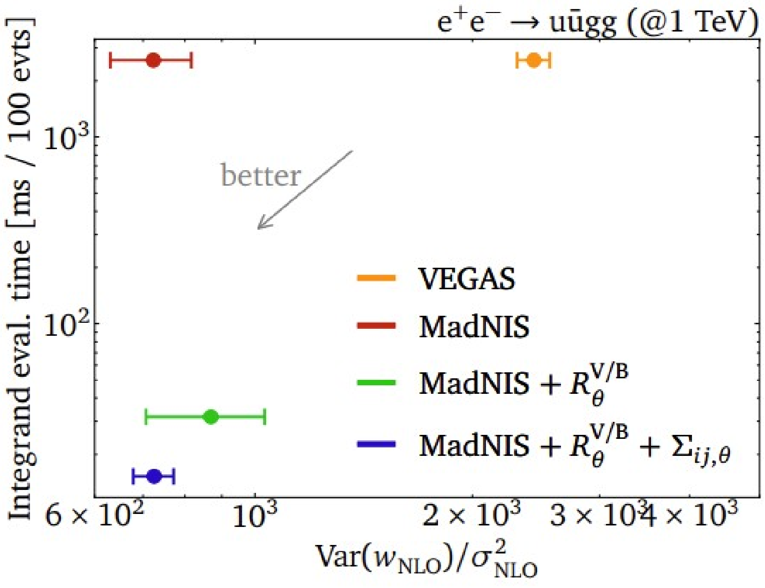
Surrogates + Normalising flows

Madnis@NLO [De Crescenzo, Mariño, Elmer, Heimel, Plehn, Winterhalder, Zaro 2603.22407]

Normalizing Flows



Surrogates

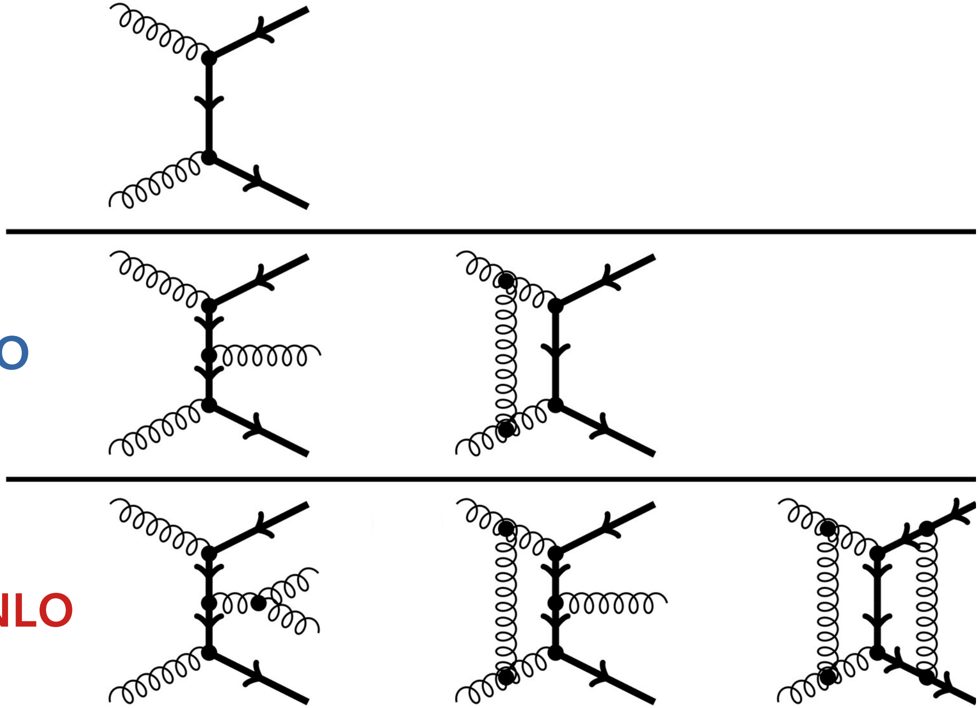


NNLO cross section contributions

LO

NLO

NNLO



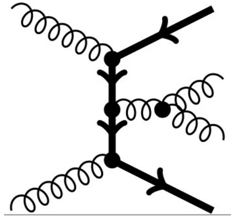
IR-finite cross section

NNLO subtraction schemes

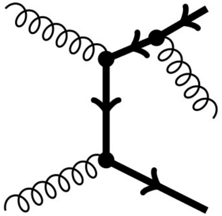
qT-slicing [Catani'07]
N-jettiness slicing [Gaunt'15/Boughezal'15]
Antenna [Gehrmann'05-'08],
Colorful [DelDuca'05-'15]
Sector-improved residue subtraction [Czakon'10-'14,'19]
Projection-to-Born [Cacciari'15]
Nested collinear [Caola'17]
Local Analytic [Magnea'18]
Geometric [Herzog'18]
Unsubtraction [Aguilera-Verdugo'19]

Anatomy of a fixed-order real emission event

Matrix elements | Kinematics + weight | Observable

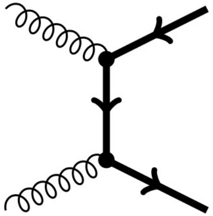


$$\Phi_{n+2},\ w_{n+2} \longrightarrow \mathcal{O}(\Phi_{n+2})$$

$-$  $\otimes \mathcal{S}_1$

$$\Phi_{n+1},\ w_{n+1} \longrightarrow \mathcal{O}(\Phi_{n+1})$$

generic soft/splitting functions

$-$  $\otimes \mathcal{S}_2$

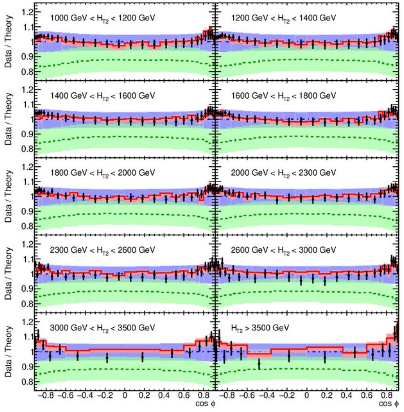
$$\Phi_{n},\ w_{n} \longrightarrow \mathcal{O}(\Phi_{n})$$

$$\boxed{w_{n+2} + w_{n+1} + w_n = \text{finite}}$$

In all infrared limits

Computational costs of fixed-order cross sections

- Normally evaluated as **weighted** MC integrals
  (optimisations for efficient integration like VEGAS implied)
  → histogram cross sections
  (if you change bin edges you start from scratch)
- Can be computationally very challenging
- Negative subtraction source of large **variance/MC error**
  [ATLAS 2301.09351]



→ O(100 MCPUh)

Costs for typical NNLO calc:

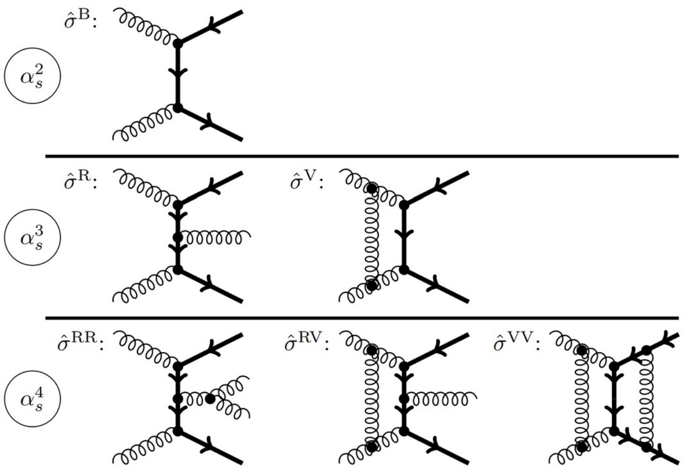
Process class	Core-hours
$pp \to V$	~hours
$pp \to VV$	O(1k)
$pp \to V + j$	O(>10k)
$pp \to jj$	O(>100k)
$pp \to jjj$	O(>1M)

How to make this more efficient?
→ can ML help?

Application to NNLO QCD cross sections

Benchmarks

- To make life simple: optimization goal is total cross section
  → focusing on **gluonic top-pair production**
  (small number of channels)



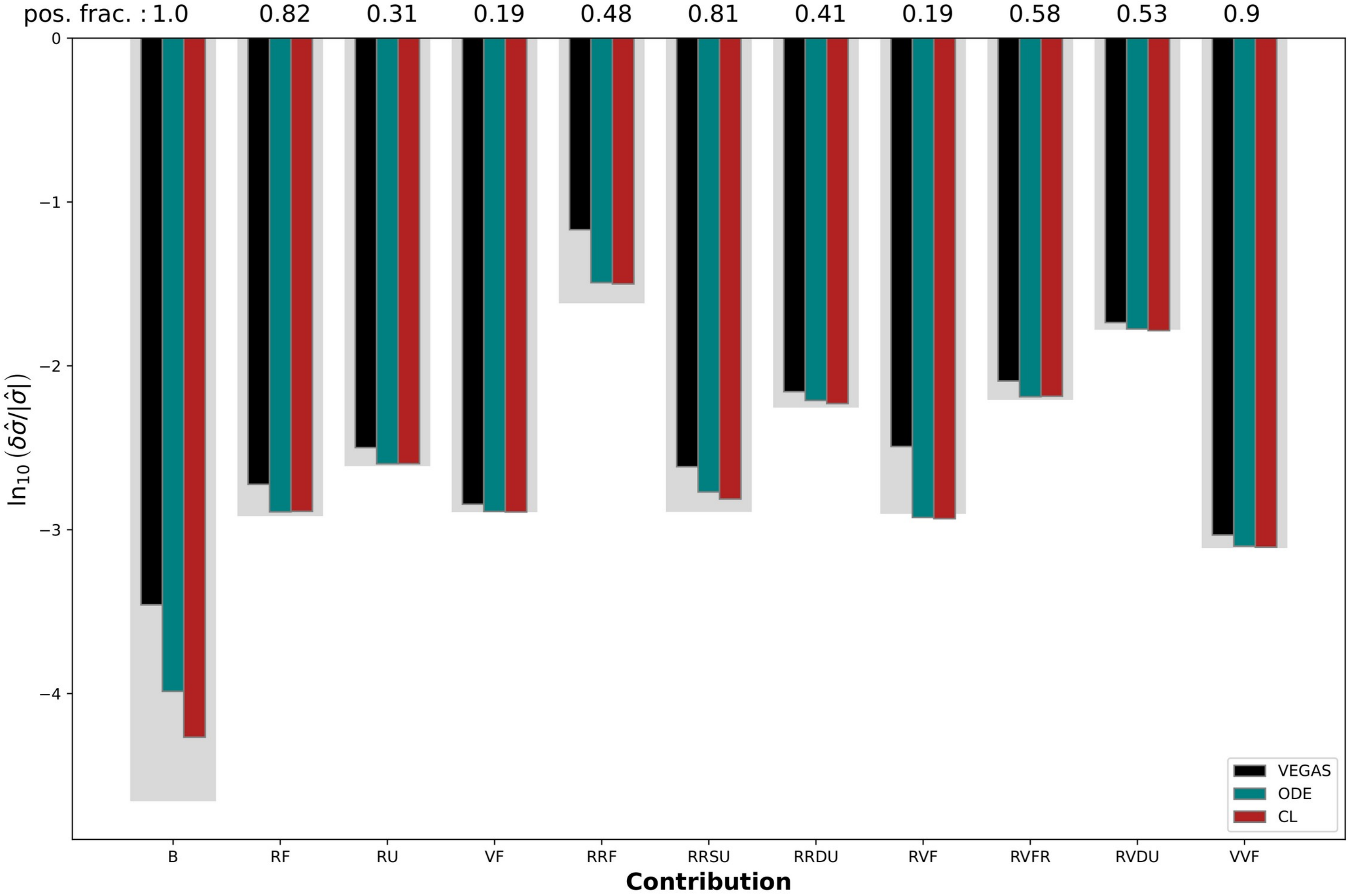
- Three integrators:
  VEGAS (reference for the current default), **CL, ODE**
  → different implementation of discrete parameters:
  continuous and embedded (flows only)

- Benchmark quantities:
  - Weight variance / MC error $$\delta\hat{I} = \sqrt{\frac{1}{N-1}\left(\frac{1}{N}\sum_{i=1}^{N} f^2(\mathbf{x}_i) - \hat{I}^2\right)}$$
  - Unweighting efficiency $$\approx \frac{\langle w\rangle}{w_{\max}}$$

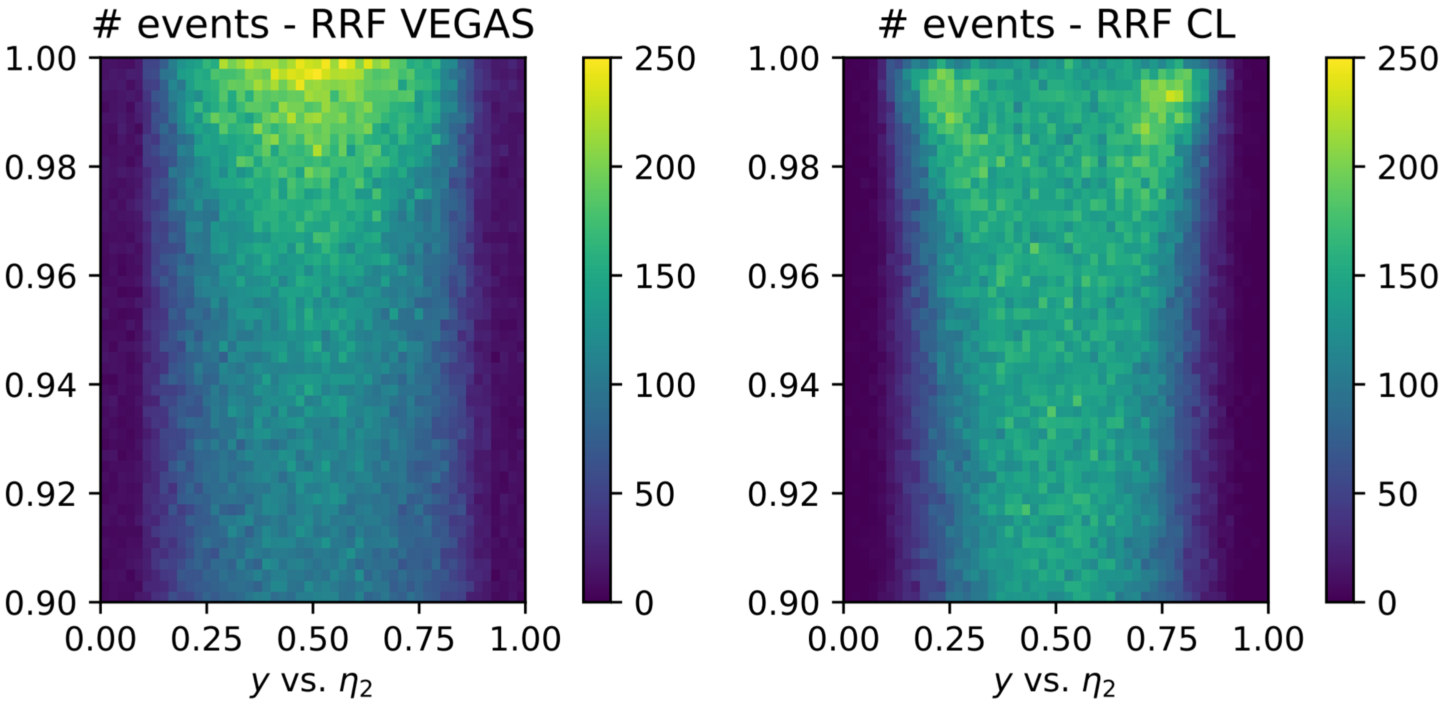
NNLO cross section results

Frozen integrators
1M points

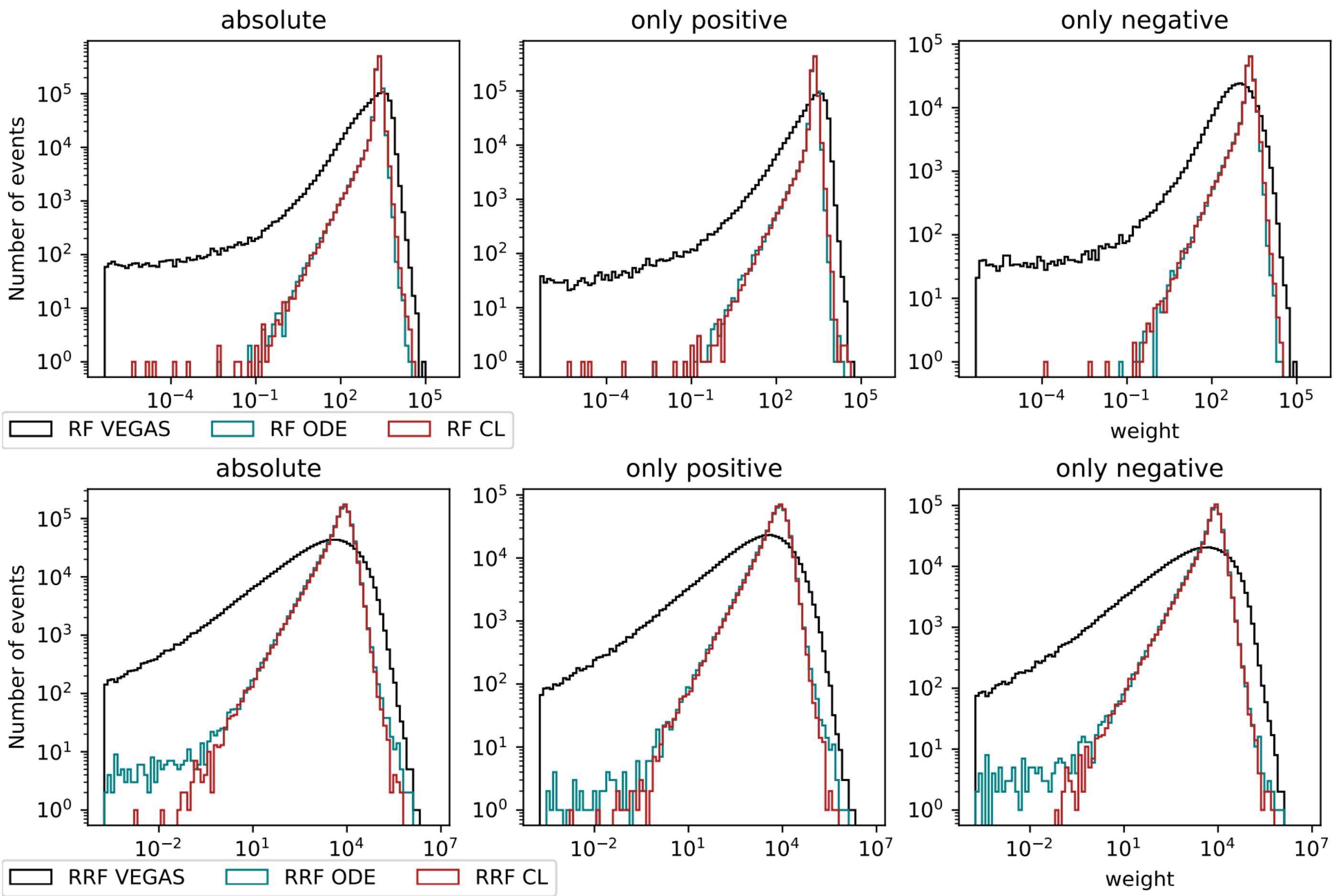
Normalizing Flows appear not to be a game changer**⁉**



Non-factorizing phase space features



Weight distributions



This is a **log-plot**

→ suggests that the variance with NFs are **much better**

Non-positive definite integrands

- Non-definite integrands introduce new challenges

  → cancellation between +/- parts increase the variance

- Consider extreme case: $|f(x)|/h(x) = w = \text{const.}$

  MC estimate:

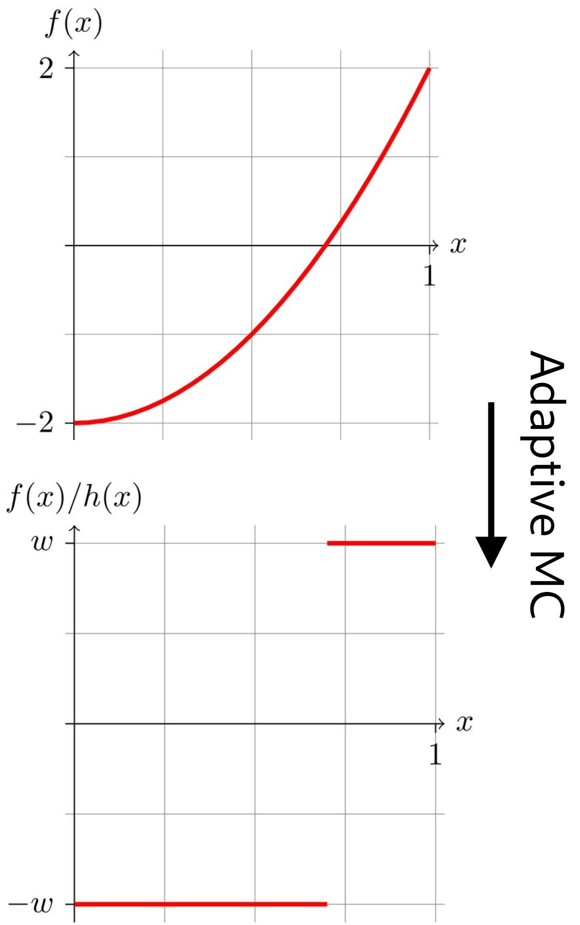
$$\hat{I} = w\frac{N_+ - N_-}{N} \equiv w(2\alpha - 1) \qquad \alpha = N_+/N$$

- Lower bound on variance:

$$\boxed{\mathrm{Var}(\hat{I}) = w^2 - w^2(2\alpha-1)^2 = w^2(4\alpha(1-\alpha))}$$

→ relative uncertainty: $\frac{\delta\hat{I}}{\hat{I}} = \frac{1}{\sqrt{N-1}}\frac{\sqrt{\alpha(1-\alpha)}}{\alpha - \frac{1}{2}}$

Rephrased: at some point it doesn't matter any more how good your adaptive MC is...

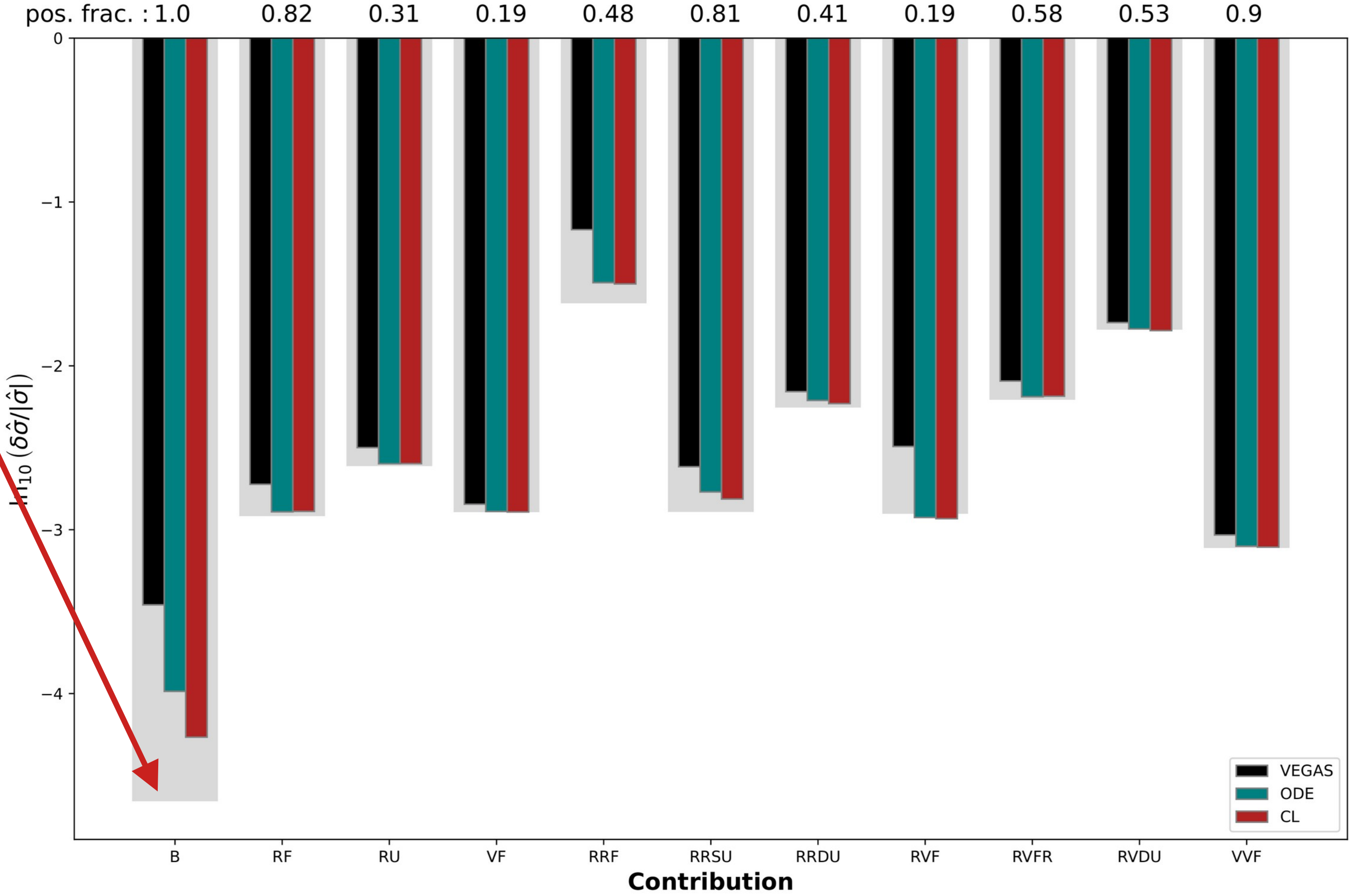


NNLO cross section results → ‘optimal’ uncertainty

Grey boxes
‘Optimal’ uncertainty!

Observation #1:
Saturation of lower bounds both with NF & VEGAS
(for NNLO → NF slightly better)

side remark/interpretation:
1) compliment PS parametrization and/or
2) too simple a problem



Stratification of signed integrands

There are ways around:

1) Add a large constant

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

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

"two independent integrals"

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

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

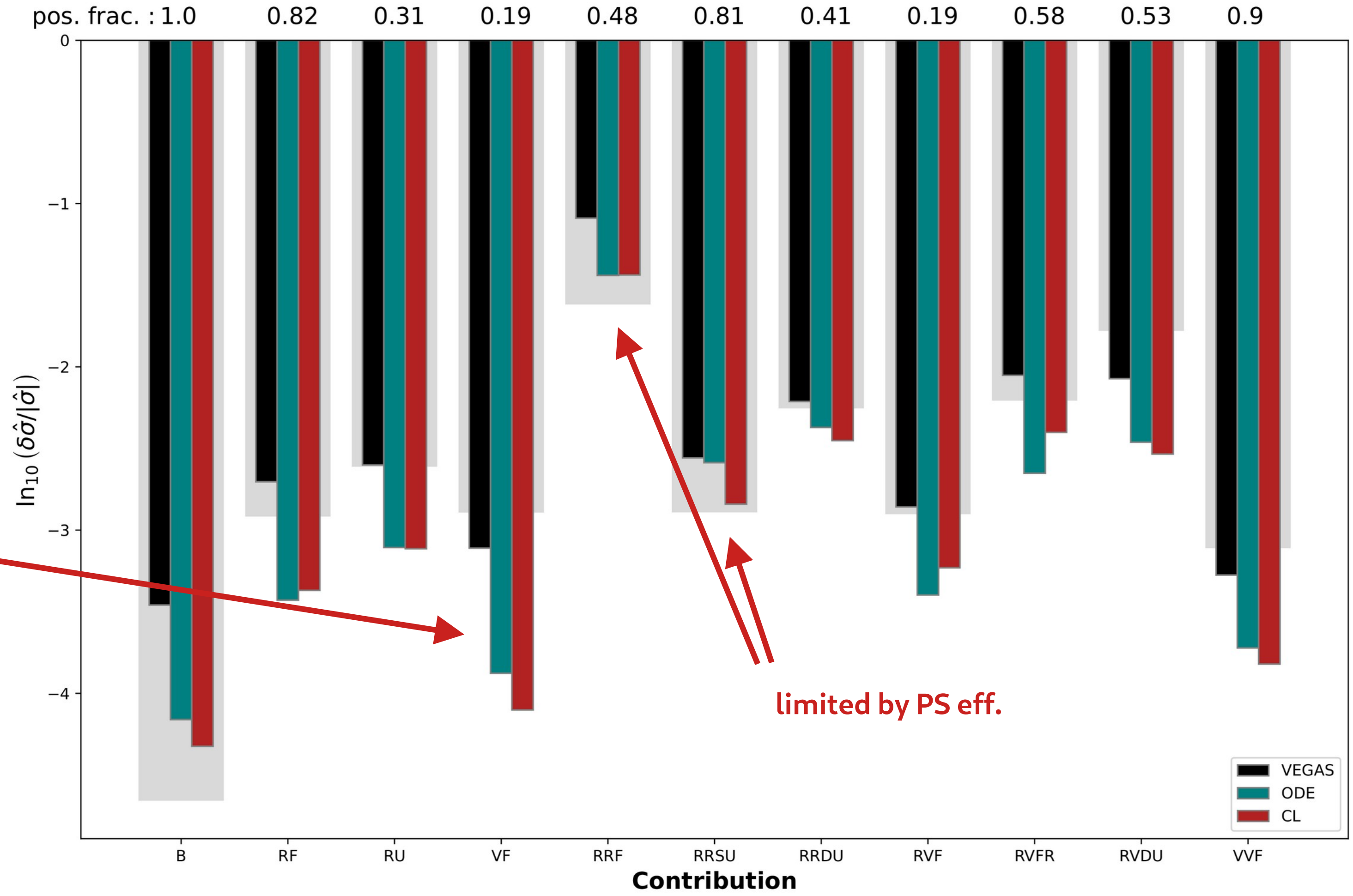
$$\mathrm{Var}(\hat{I}_\pm) = \frac{1}{N_\pm}\sum_{i=1}^{N_\pm}\left(\frac{f_\pm(\mathbf{x}_i)}{h_\pm(\mathbf{x}_i)}\right)^2 - \hat{I}_\pm^2$$

+ The total variance is now bounded by the individual variances

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

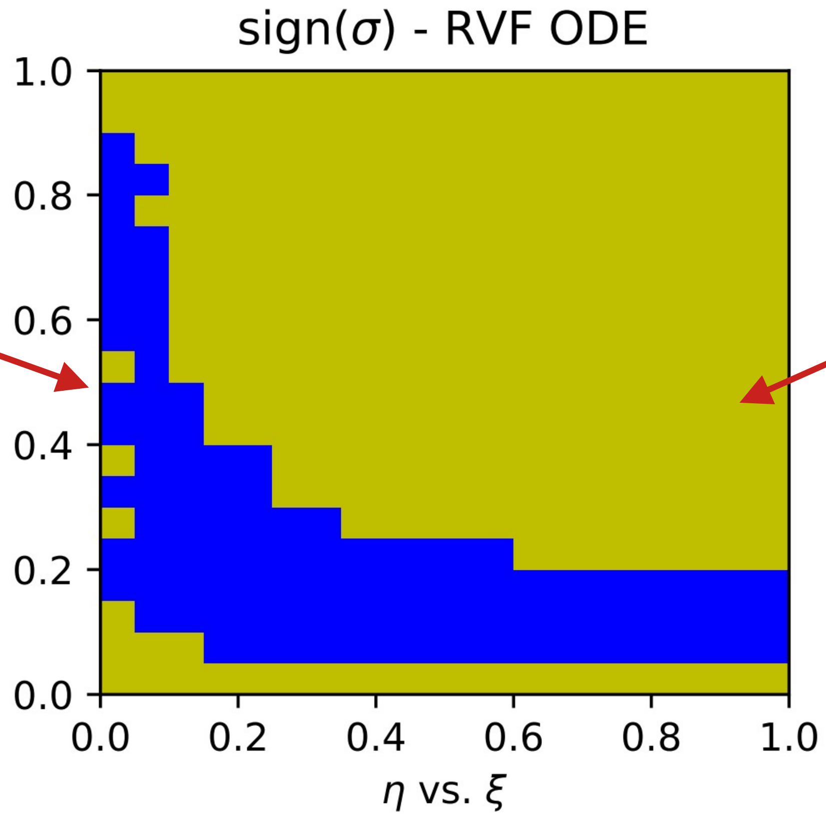
Relative uncertainties with positive/negative split

Observation #2:
Splitting can give a significant performance gains for flow based integrators



Non-trivial positive/negative structure

Negative



Positive

non-factorizing
→ VEGAS not sufficient

Folding in GPU/CPU costs

Final result is sum over contributions:

$$\delta\sigma_{\text{target}} = \delta\hat{\sigma}_{\text{tot}}(\mathbf{n}) = \sqrt{\sum_i (\delta\hat{\sigma}^i(n_i))^2} = \sqrt{\sum_i \frac{\text{Var}(\hat{\sigma}^i)}{n_i - 1}}$$

Minimize the total computational cost: $C = \sum_i C_i n_i$

(→ dominated in the end by the double real contributions)

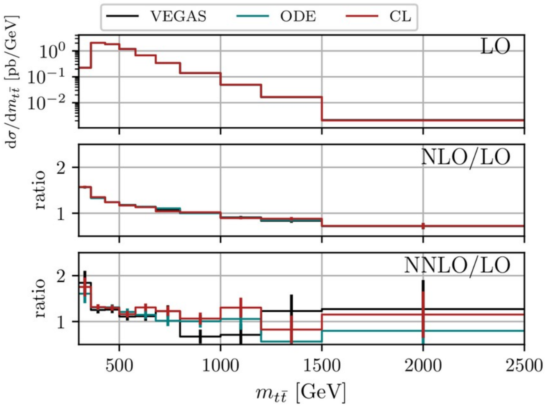
Optimal performance gains (infinity cheap NF/expensive integrand):

- **Absolute integrand training: ~2**
- **Pos/neg split: ~8**

[In our setup: 1-1.5h GPU vs. 10h CPU]

↓	↓
Flow dominated by disk I/O	Integrand dominated by ME

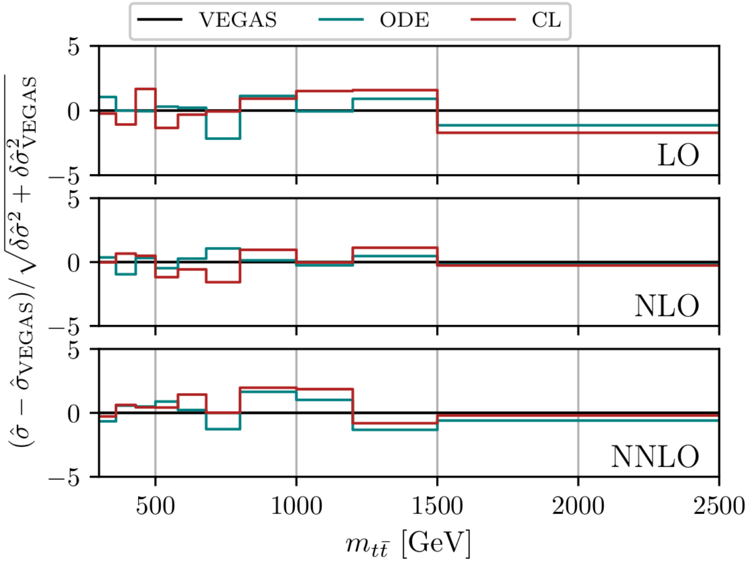
Differential distributions

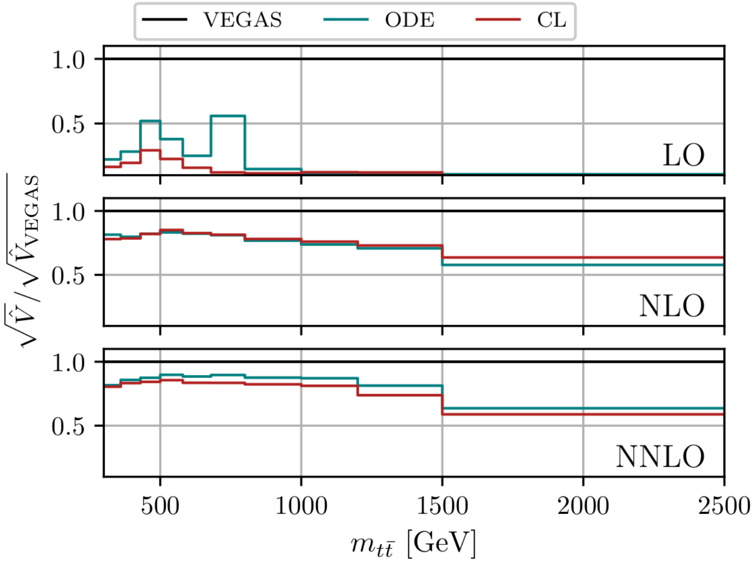


← Steeply falling spectrum

Validation
→ method works differentially

Performance gain limited
by non-definiteness





Summary

- Normalizing flows as a tool for variance reduction
- Investigated the usage of flow-based methods for NNLO QCD phase space sampling
  → Reduction of weight distributions
  → Pos/Neg cancellation bottleneck in many cases → lower bound as for unweighted samples

- Pos/Neg splitting allows **to go beyond this bound**

- Performance gains of O(10) with pos/neg splitting + normalising flow can be achieved
  → **non-factorising model is crucial**!

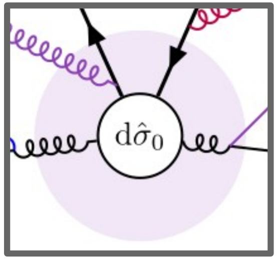
Outlook (wip):

→ fully fledged integration in STRIPPER (& HighTEA)

→ go beyond "simple" NNLO QCD top-pair

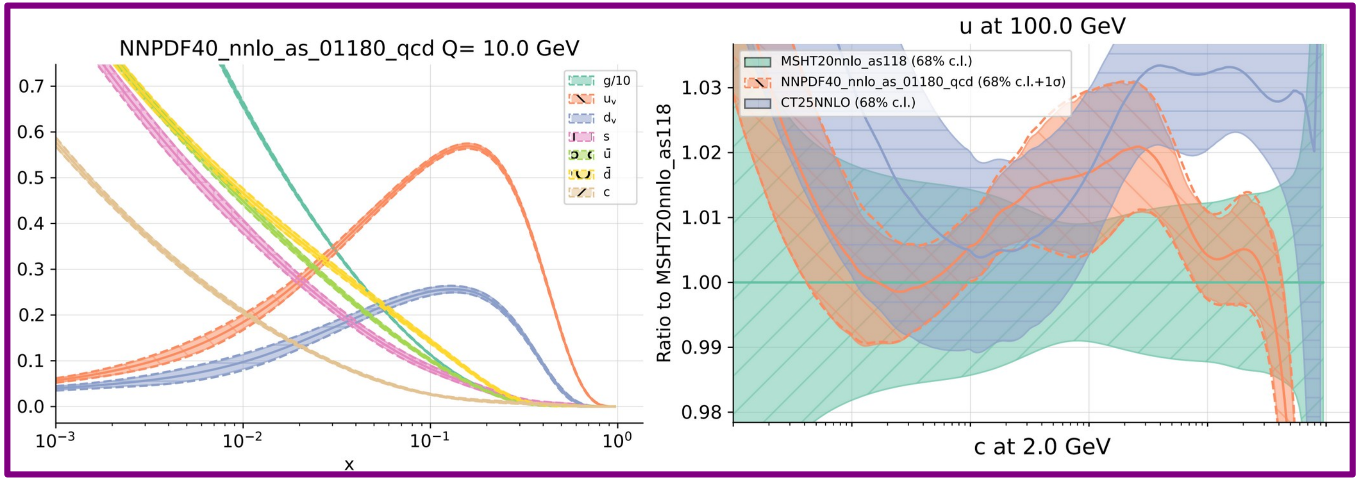
→ differential pos/neg splitting

Fixed-order QCD in collinear factorization

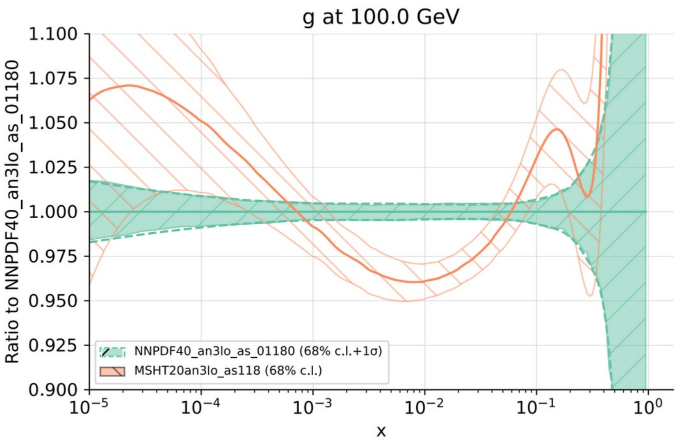


$$\sigma_{h_1 h_2 \to X} = \sum_{ij} \int_0^1 \int_0^1 \underline{\mathrm{d}x_1 \mathrm{d}x_2 \phi_{i,h_1}}(x_1, \mu_F^2)\phi_{j,h_2}(x_2, \mu_F^2)\hat{\sigma}_{ij\to X}(\alpha_s(\mu_R^2), \mu_R^2, \mu_F^2)$$

Collinear (unpolarised) parton distribution functions



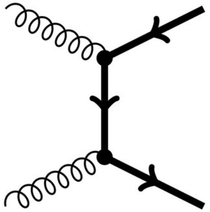
First steps towards N3LO ...

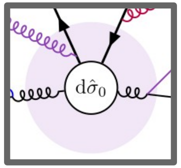


[from Juan's DIS talk last week]

Fixed-order QCD in collinear factorization

LO



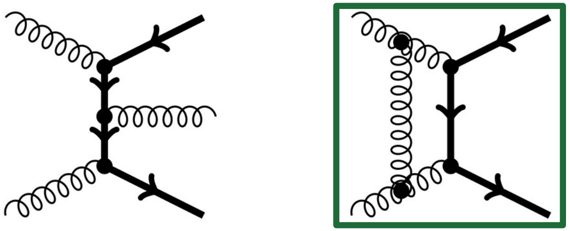


$$\sigma_{h_1h_2\to X} = \sum_{ij}\int_0^1\int_0^1 \mathrm{d}x_1\mathrm{d}x_2\phi_{i,h_1}(x_1,\mu_F^2)\phi_{j,h_2}(x_2,\mu_F^2)\hat{\sigma}_{ij\to X}(\alpha_s(\mu_R^2),\mu_R^2,\mu_F^2)$$

partonic cross section

$$\hat{\sigma}_{ab\to X} = \hat{\sigma}^{(0)}_{ab\to X} + \hat{\sigma}^{(1)}_{ab\to X} + \hat{\sigma}^{(2)}_{ab\to X} + \mathcal{O}(\alpha_s^3)$$

NLO



$$\hat{\sigma}^{(i)} \ni \frac{1}{\mathcal{F}}\int |\mathcal{M}_n|^2 \mathrm{d}\Phi_n$$

IR-finite cross section

NNLO

Automated fixed-order @ NLO

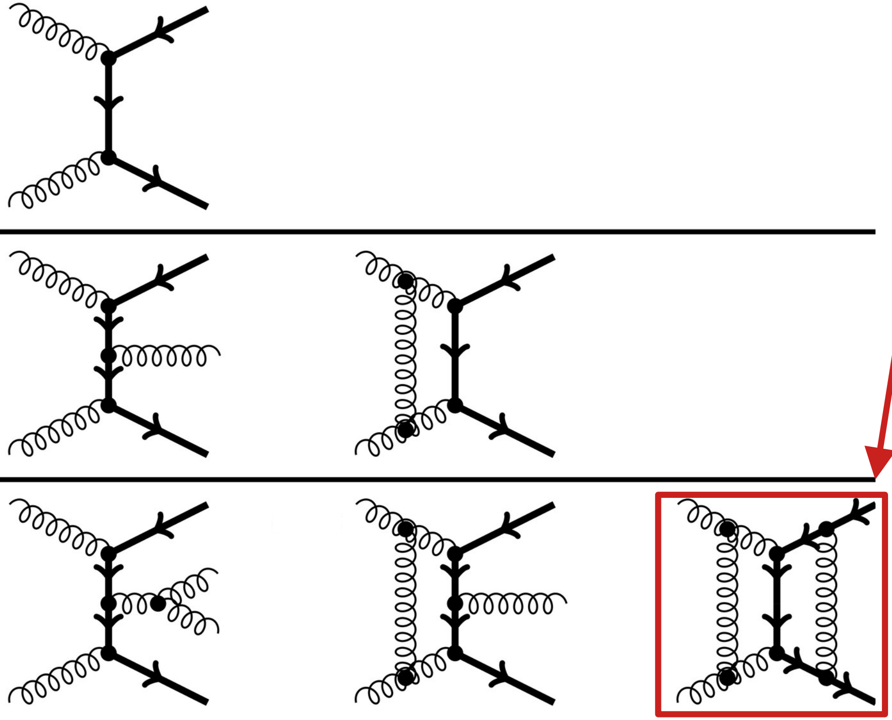
- oneloop-matrix elements
  - Openloops, Recola, Helac, MadGraph, GoSam
- local QCD/EW subtraction
  - FKS: MadGraph5_aMC@NLO, Powheg
  - CS dipoles: Sherpa, Herwig

NNLO QCD challenges: two-loop amplitudes

LO

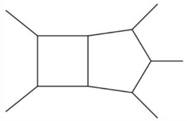
NLO

NNLO

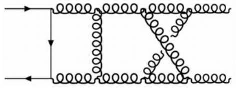


How to compute
multi-scale two-loop QCD amplitudes?
→ **fast growing complexity:**
rational coef. and **special functions**
→ deeper understanding of the analytical properties
→ refinement of computational tools

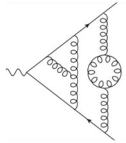
Two-loop 5-point
[Abreu, Agarwal, Badger, Buccioni, Chawdhry, Chicherin,Czakon, de Laurenties, Febres-Cordero, Gambuti, Gehrmann, Henn, Lo Presti, Manteuffel, Ma, Mitov, Page, Peraro, Poncelet, Schabiner Sotnikov, Tancredi, Zhang,...]



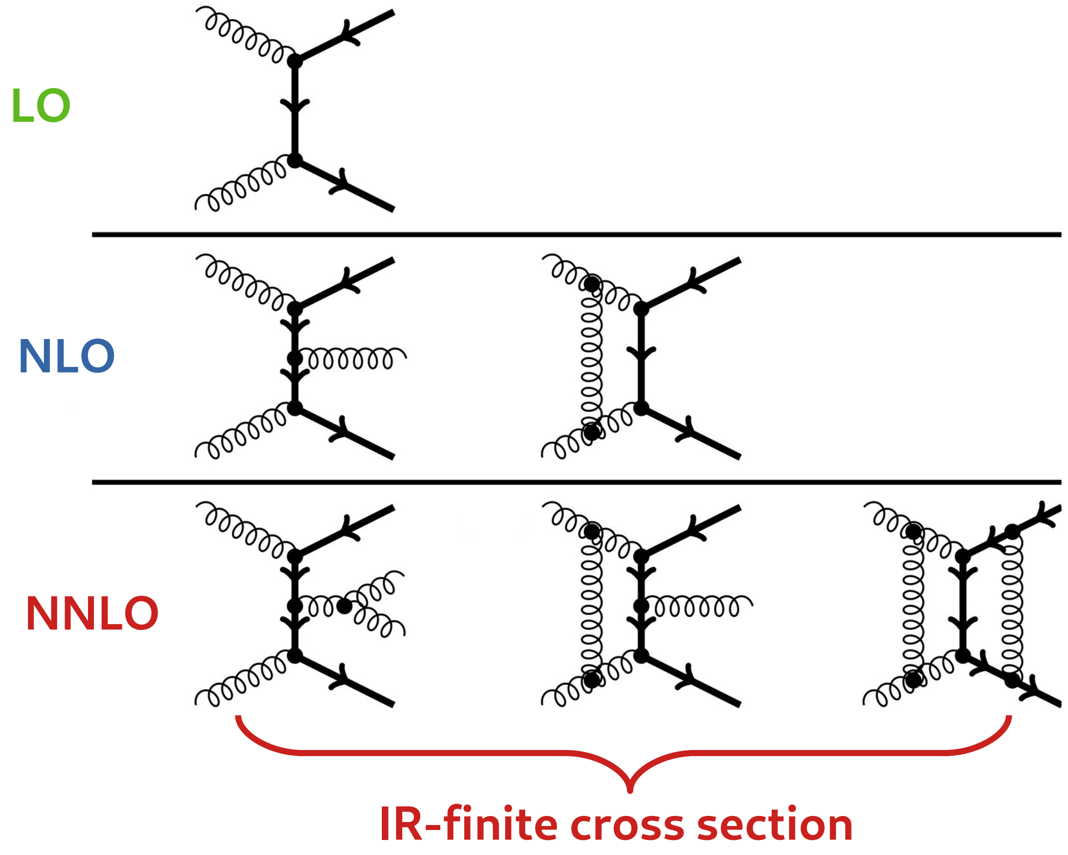
Three-loop 4-point
[Bargiela, Dobadilla, Canko, Caola, Jakubcik, Gambuti, Gehrmann, Henn, Lim, Mella, Mistleberger, Wasser, Manteuffel, Syrrakow, Smirnov, Trancredi, ...]



Four-loop 3-point
[Henn, Lee, Manteuffel, Schabinger, Smirnov, Steinhauser,...]



NNLO QCD challenges: subtraction



How to achieve **infrared (IR) finite differential** cross sections at NNLO QCD?

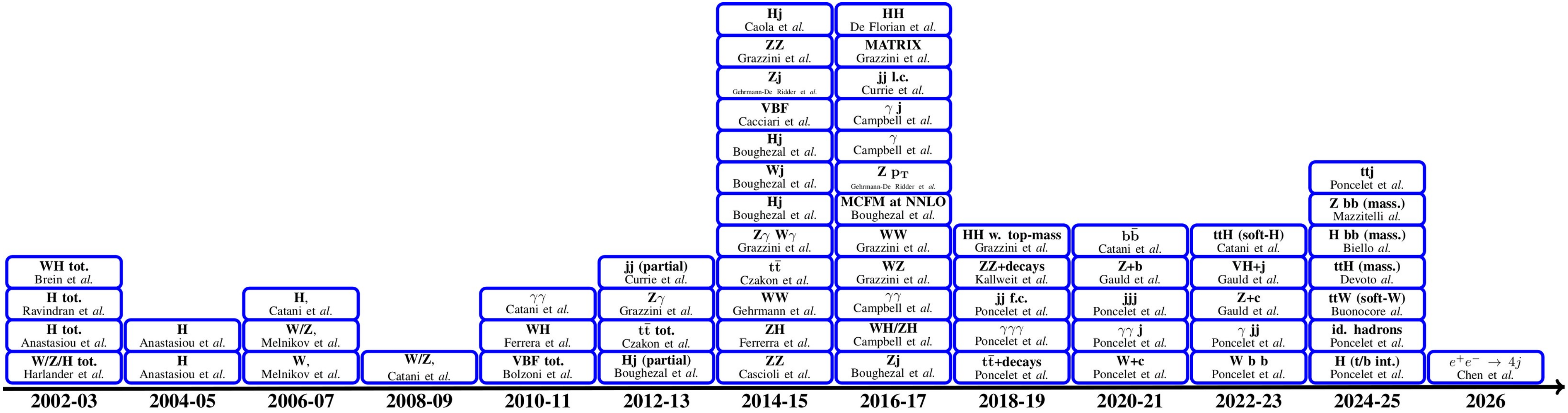
~20 years to solve this problem

qT-slicing [Catani'07], N-jettiness slicing [Gaunt'15/Boughezal'15], Antenna [Gehrmann'05-'08], Colorful [DelDuca'05-'15], Projetction [Cacciari'15], Geometric [Herzog'18], Unsubtraction [Aguilera-Verdugo'19], Nested collinear [Caola'17], Local Analytic [Magnea'18] Sector-improved residue subtraction [Czakon'10-'14,'19]

Automation of NNLO QCD subtraction:

- qT-slicing (massive/color singlet) in **MATRIX** & **MCFM** (public)
- Antenna subtraction in **NNLOJet** (public)
- Sector-improved residue subtraction in **Stripper** (private impl.)
- color singlets with nested-collinear in **history** (public)

Status of NNLO QCD cross section computations



All 2 → 2 SM processes, 2 → 3+ only **limited by**

→ the available two-loop matrix elements

→ **numerical/computational challenges**

Training and model parameters

Training target based on Kullback–Leibler (KL) divergence:

$$D_{\mathrm{KL}}(p \parallel q_\theta) = \sum_{i=1}^{N} p(\mathbf{y}_i) \log\left(\frac{p(\mathbf{y}_i)}{q_\theta(\mathbf{y}_i)}\right)$$

1) Generate sample & evaluate target function
   (1M points)

2) Do NN optimization step with gradient descend (ADAM)

3) Repeat 1)
   (10 times)

Size of neural networks depends on the dimension of the problem.

→ we investigated 4 to 13 dimensional problems + discrete parameters (conditional NNs):

~ 100k – 10M parameters for CL flows

~ 1M parameter for ODE flows

The integrands

Matrix elements + measurement functions

Structure of partonic contributions:

different partonic channels $\hat{\sigma}^i_{ab} = \sum_{j \in \mathcal{C}^i_{ab}(ab \to X)} \hat{\sigma}^i_{ab,j}$ $\Rightarrow$ $\hat{\sigma}^i_{ab,j} = N_j \int \mathrm{d}\Phi_j \mathcal{F}_{ab,j}(\Phi_j)$

Subtraction for real-emission contributions
based on sector-decomposition + residue subtraction:

$$\hat{\sigma}^{(1)} \ni \int \mathrm{d}\Phi_j \sum_{kl} \mathcal{S}_{kl} |\mathcal{M}_j(\Phi_j)|^2 F(\Phi_j)\,. \qquad \hat{\sigma}^{(i)} \ni \int_{[0,1]^m} \mathrm{d}^m\chi \int_{[0,1]^n} \mathrm{d}^n\mathbf{x} \frac{f^j_{\{k\}}(\chi,\mathbf{x}) - \sum f^j_{\{k\}}(\chi|_{\to 0},\mathbf{x})}{\prod_i^m \chi_i}$$

But for the integration problem (in case of total cross sections) it is enough to consider:

$$\boxed{\hat{\sigma}^{(i)} \ni \int_{[0,1]^n} \mathrm{d}\mathbf{y}\, g^j_{\{k\}}(\mathbf{y})}$$

$$\sum_{\{k\},h} \int_{[0,1]^n} \mathrm{d}\mathbf{y}\, g^j_{\{k\},h}(\mathbf{y}) = \sum_l \int_{[0,1]^n} \mathrm{d}\mathbf{y}\, g^j_l(\mathbf{y})$$

Sum of
Sectors, helicities, channels

Coupling Layer Normalizing Flow

Based on the i-flow paper: 2001.05486

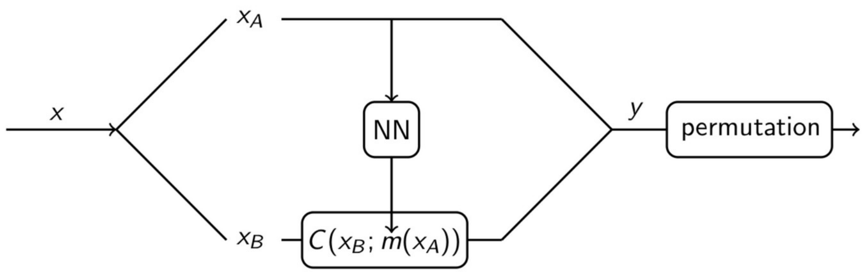
Series of bijections:

$$\vec{x}_K = c_K(c_{K-1}(\cdots c_2(c_1(\vec{x}))))$$

Distribution:

$$g_K(\vec{x}_K) = g_0(\vec{x}_0) \prod_{k=1}^{K} \left| \frac{\partial c_k(\vec{x}_{k-1})}{\partial \vec{x}_{k-1}} \right|^{-1}, \quad \text{where} \quad \begin{cases} \vec{x}_0 = \vec{x} \\ \vec{x}_k = c_k(\vec{x}_{k-1}) \end{cases}$$

Structure of a single coupling layer:



$$\begin{aligned} x'_A &= x_A, & A &\in [1, d]\,, \\ x'_B &= C(x_B; m(\vec{x}_A)), & B &\in [d+1, D]\,. \end{aligned}$$

The inverse map

$$x_A = x'_A \, ,$$
$$x_B = C^{-1}(x'_B; m(\vec{x}'_A)) = C^{-1}(x'_B; m(\vec{x}_A))$$

Inverse Jacobian: $\left|\frac{\partial c(\vec{x})}{\partial \vec{x}}\right|^{-1} = \left|\begin{pmatrix} \vec{1} & 0 \\ \frac{\partial C}{\partial m}\frac{\partial m}{\partial \vec{x}_A} & \frac{\partial C}{\partial \vec{x}_B} \end{pmatrix}\right|^{-1} = \left|\frac{\partial C(\vec{x}_B; m(\vec{x}_A))}{\partial \vec{x}_B}\right|^{-1}$

Derivative of the NN

Big advantage: $\frac{\partial m}{\partial x_A}$ not needed!
→ Performance

How many coupling layers you need? D for $D \leq 5$ & $2\log_2 D$ for $D > 5$

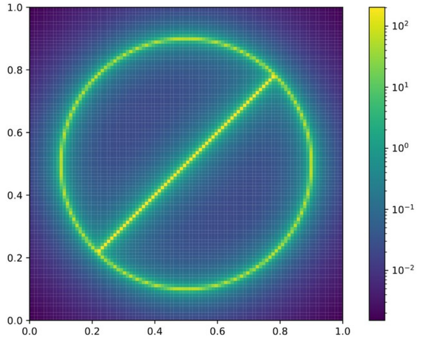
Example: Masking to capture all correlations for D=12

	Dimension	0	1	2	3	4	5	6	7	8	9	10	11
2x	Transformation 1	0	1	0	1	0	1	0	1	0	1	0	1
	Transformation 2	0	0	1	1	0	0	1	1	0	0	1	1
	Transformation 3	0	0	0	0	1	1	1	1	0	0	0	0
	Transformation 4	0	0	0	0	0	0	0	0	1	1	1	1

A toy example

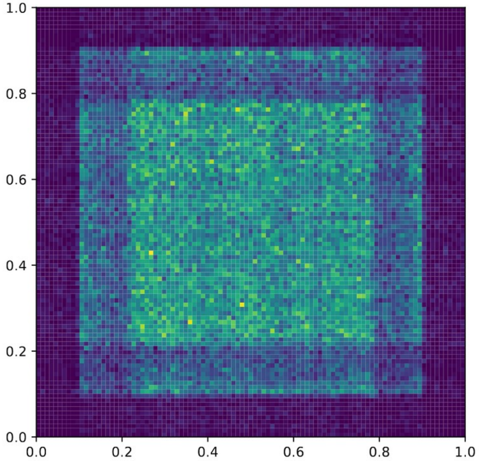
Multi-modular function ("stop-sign"):

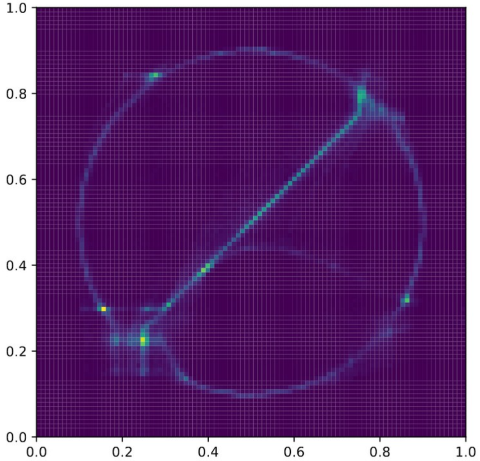
$$f(x,y) = \frac{1}{2\pi^2} \cdot \frac{\Delta r}{\left(\sqrt{(x-x_0)^2+(y-y_0)^2} - r_0\right)^2 + (\Delta r)^2} \cdot \frac{1}{\sqrt{(x-x_0)^2+(y-y_0)^2}}$$
$$+\frac{1}{2\pi r_0} \cdot \frac{\Delta r}{((y-y_0)-(x-x_0))^2+(\Delta r)^2} \cdot \Theta\left(r_0 - \sqrt{(x-x_0)^2+(y-y_0)^2}\right).$$



Sampling densities:

VEGAS





Coupling-Layer Flow

Continuous Flows

"time"-dependent probability density function: $\log q_t(\mathbf{y}_t) = \log q_0(\mathbf{y}_0) - \log\left|\det \frac{\partial \phi_t}{\partial \mathbf{y}_0}\right|$

constructed from $\frac{\mathrm{d}}{\mathrm{d}t}\phi_t(\mathbf{y}) = v_t(\phi_t(\mathbf{y}))\,, \quad \text{with} \quad \phi_0(\mathbf{y}) = \mathbf{y}$

+continuity equation (preserve prob.)

vector-field is given by a trainable NN

The mapped point is the solution of a simple ODE: $\mathbf{y}_1 = \phi_1(\mathbf{y}_0) = \int_0^1 v_t(\phi_t(\mathbf{y}_0))\,\mathrm{d}t$

Jacobian by inverse ODE:

$$\frac{\mathrm{d}}{\mathrm{d}s}\begin{bmatrix}\phi_{1-s}(\mathbf{y}) \\ f(1-s)\end{bmatrix} = \begin{bmatrix}-v_{1-s}(\phi_{1-s}(\mathbf{y})) \\ -\operatorname{div}\big(v_{1-s}(\phi_{1-s}(\mathbf{y}))\big)\end{bmatrix} \qquad \begin{bmatrix}\phi_1(\mathbf{y}) \\ f(1)\end{bmatrix} = \begin{bmatrix}\mathbf{y}_1 \\ 0\end{bmatrix}$$

$$\rightarrow \quad \log q_1(\mathbf{y_1}) = \log q_0(\mathbf{y_0}) - f(0)$$

Results LO and NLO

Contribution	$\sigma_{\mathrm{B}} \cdot 10^{-2}$	$\sigma_{\mathrm{RF}} \cdot 10^{-2}$	$\sigma_{\mathrm{RU}} \cdot 10^{-2}$	$\sigma_{\mathrm{VF}} \cdot 10^{-1}$
δ^{opt}	0.0001	0.008	0.01	0.006
CPU cost [a.u.]	1	6	1.5	1.3
VEGAS				
$\hat{\sigma} \pm \delta\hat{\sigma}$	5.3278 ± 0.002	6.967 ± 0.01	-5.378 ± 0.02	-4.487 ± 0.006
$\hat{\sigma}^+ \pm \delta\hat{\sigma}^+$	5.3284 ± 0.002	8.955 ± 0.01	4.432 ± 0.006	1.403 ± 0.0008
$\hat{\sigma}^- \pm \delta\hat{\sigma}^-$	$-0.0006446 \pm 5e-07$	-1.981 ± 0.004	-9.8098 ± 0.008	-5.8939 ± 0.003
$\Sigma^{\pm}$	5.3278 ± 0.002	6.973 ± 0.01	-5.378 ± 0.01	-4.4909 ± 0.003
ϵ_{Φ}^{+}	0.99	0.728	0.639	0.808
ϵ_{Φ}^{-}	0.834	0.384	0.852	0.95
$\epsilon^+ (\epsilon^+_{0.1\%})$	0.13 (0.43)	0.048 (0.098)	0.02 (0.048)	0.082 (0.21)
$\epsilon^- (\epsilon^-_{0.1\%})$	0.016 (0.066)	0.013 (0.021)	0.049 (0.17)	0.12 (0.27)
ODE Flow				
$\hat{\sigma} \pm \delta\hat{\sigma}$	5.3279 ± 0.0005	6.98 ± 0.009	-5.394 ± 0.01	-4.483 ± 0.006
$\hat{\sigma}^+ \pm \delta\hat{\sigma}^+$	5.32872 ± 0.0004	8.9494 ± 0.002	4.4185 ± 0.002	1.4018 ± 0.0001
$\hat{\sigma}^- \pm \delta\hat{\sigma}^-$	$-0.00064495 \pm 1e-07$	-1.9844 ± 0.0006	-9.802 ± 0.003	-5.89315 ± 0.0005
$\Sigma^{\pm}$	5.32808 ± 0.0004	6.965 ± 0.003	-5.3835 ± 0.004	-4.4914 ± 0.0006
ϵ_{Φ}^{+}	1.0	0.991	0.992	1.0
ϵ_{Φ}^{-}	0.997	0.99	0.987	0.999
$\epsilon^+ (\epsilon^+_{0.1\%})$	0.33 (0.7)	0.025 (0.3)	0.0059 (0.099)	0.11 (0.56)
$\epsilon^- (\epsilon^-_{0.1\%})$	0.055 (0.36)	0.028 (0.17)	0.02 (0.16)	0.12 (0.73)
Coupling Layer Flow				
$\hat{\sigma} \pm \delta\hat{\sigma}$	5.32795 ± 0.0003	6.972 ± 0.009	-5.39 ± 0.01	-4.492 ± 0.006
$\hat{\sigma}^+ \pm \delta\hat{\sigma}^+$	5.32807 ± 0.0003	8.949 ± 0.002	4.4101 ± 0.002	$1.40155 \pm 9e-05$
$\hat{\sigma}^- \pm \delta\hat{\sigma}^-$	$-0.000644883 \pm 3e-08$	-1.9821 ± 0.0007	-9.8 ± 0.002	-5.89183 ± 0.0003
$\Sigma^{\pm}$	5.32742 ± 0.0003	6.9669 ± 0.003	-5.3899 ± 0.004	-4.49028 ± 0.0004
ϵ_{Φ}^{+}	1.0	0.989	0.988	1.0
ϵ_{Φ}^{-}	1.0	0.99	0.994	1.0
$\epsilon^+ (\epsilon^+_{0.1\%})$	0.53 (0.81)	0.028 (0.24)	0.0082 (0.046)	0.17 (0.63)
$\epsilon^- (\epsilon^-_{0.1\%})$	0.11 (0.79)	0.0074 (0.06)	0.009 (0.19)	0.4 (0.79)

Results NNLO

Contribution	$\sigma_{\mathrm{RRF}} \cdot 10^{-2}$	$\sigma_{\mathrm{RRSU}} \cdot 10^{-3}$	$\sigma_{\mathrm{RRDU}} \cdot 10^{-3}$	$\sigma_{\mathrm{RVF}} \cdot 10^{-2}$	$\sigma_{\mathrm{RVFR}} \cdot 10^{-2}$	$\sigma_{\mathrm{RVDU}} \cdot 10^{-3}$	$\sigma_{\mathrm{VVF}} \cdot 10^{-0}$
δ^{opt}	0.0079	0.0036	0.013	0.0081	0.0018	0.0045	0.011
CPU cost [a.u.]	53	26	8	1542	2.7	2.7	13
VEGAS							
$\hat{\sigma} \pm \delta\hat{\sigma}$	-0.3 ± 0.02	2.835 ± 0.0069	-2.461 ± 0.017	-6.808 ± 0.022	0.2864 ± 0.0023	0.267 ± 0.0049	14.768 ± 0.014
$\hat{\sigma}^+ \pm \delta\hat{\sigma}^+$	3.823 ± 0.013	3.718 ± 0.0056	5.639 ± 0.0065	1.669 ± 0.0026	1.051 ± 0.0014	2.381 ± 0.0011	16.711 ± 0.0067
$\hat{\sigma}^- \pm \delta\hat{\sigma}^-$	-4.135 ± 0.013	-0.8813 ± 0.0022	-8.078 ± 0.0085	-8.4539 ± 0.0068	-0.7635 ± 0.0012	-2.1114 ± 0.0012	-1.955 ± 0.0011
$\Sigma^{\pm}$	-0.312 ± 0.025	2.836 ± 0.0079	-2.439 ± 0.015	-6.785 ± 0.0094	0.2871 ± 0.0025	0.2696 ± 0.0023	14.756 ± 0.0078
ϵ_{Φ}^{+}	0.54	0.732	0.803	0.55	0.593	0.946	0.956
ϵ_{Φ}^{-}	0.513	0.402	0.8	0.871	0.501	0.916	0.816
$\epsilon^{+}(\epsilon^{+}_{0.1\%})$	0.004 (0.0052)	0.0073 (0.017)	0.0018 (0.022)	0.013 (0.024)	0.025 (0.06)	0.029 (0.22)	0.015 (0.1)
$\epsilon^{-}(\epsilon^{-}_{0.1\%})$	0.0041 (0.0053)	0.0035 (0.01)	0.0037 (0.019)	0.014 (0.038)	0.021 (0.05)	0.011 (0.11)	0.024 (0.17)
ODE Flow							
$\hat{\sigma} \pm \delta\hat{\sigma}$	-0.333 ± 0.011	2.823 ± 0.0048	-2.421 ± 0.015	-6.784 ± 0.0081	0.2893 ± 0.0019	0.271 ± 0.0046	14.758 ± 0.012
$\hat{\sigma}^+ \pm \delta\hat{\sigma}^+$	3.812 ± 0.0063	3.7056 ± 0.0028	5.6439 ± 0.0036	1.6764 ± 0.00085	1.0495 ± 0.00038	2.3806 ± 0.00043	16.72 ± 0.0026
$\hat{\sigma}^- \pm \delta\hat{\sigma}^-$	-4.142 ± 0.0057	-0.8848 ± 0.0045	-8.071 ± 0.0068	-8.4578 ± 0.0019	-0.76271 ± 0.00026	-2.1109 ± 0.0005	-1.9554 ± 0.00023
$\Sigma^{\pm}$	-0.33 ± 0.012	2.821 ± 0.0073	-2.427 ± 0.01	-6.7814 ± 0.0027	0.2868 ± 0.00064	0.2697 ± 0.00093	14.765 ± 0.0028
ϵ_{Φ}^{+}	0.855	0.945	0.977	0.988	0.981	0.998	0.999
ϵ_{Φ}^{-}	0.853	0.834	0.982	0.991	0.983	0.998	0.998
$\epsilon^{+}(\epsilon^{+}_{0.1\%})$	0.0013 (0.0027)	0.0028 (0.022)	0.003 (0.023)	0.0029 (0.067)	0.0081 (0.099)	0.03 (0.27)	0.0086 (0.49)
$\epsilon^{-}(\epsilon^{-}_{0.1\%})$	0.0019 (0.0053)	0.00025 (0.00025)	0.0018 (0.0096)	0.051 (0.25)	0.017 (0.095)	0.017 (0.18)	0.17 (0.49)
Coupling Layer Flow							
$\hat{\sigma} \pm \delta\hat{\sigma}$	-0.317 ± 0.01	2.829 ± 0.0044	-2.414 ± 0.014	-6.778 ± 0.0079	0.285 ± 0.0019	0.276 ± 0.0045	14.771 ± 0.012
$\hat{\sigma}^+ \pm \delta\hat{\sigma}^+$	3.814 ± 0.0054	3.7038 ± 0.0025	5.6425 ± 0.0039	1.676 ± 0.0024	1.05 ± 0.00059	2.3799 ± 0.00043	16.718 ± 0.0021
$\hat{\sigma}^- \pm \delta\hat{\sigma}^-$	-4.134 ± 0.0063	-0.8849 ± 0.0015	-8.0756 ± 0.0047	-8.456 ± 0.0026	-0.76256 ± 0.00055	-2.1091 ± 0.00036	-1.95479 ± 0.00011
$\Sigma^{\pm}$	-0.32 ± 0.012	2.819 ± 0.0041	-2.433 ± 0.0086	-6.7803 ± 0.005	0.2874 ± 0.0011	0.2708 ± 0.00079	14.763 ± 0.0022
ϵ_{Φ}^{+}	0.838	0.956	0.985	0.987	0.988	0.998	1.0
ϵ_{Φ}^{-}	0.852	0.836	0.988	0.991	0.985	0.999	1.0
$\epsilon^{+}(\epsilon^{+}_{0.1\%})$	0.0022 (0.0056)	0.0096 (0.022)	0.0029 (0.011)	0.00076 (0.00076)	0.0024 (0.035)	0.01 (0.22)	0.02 (0.31)
$\epsilon^{-}(\epsilon^{-}_{0.1\%})$	0.0034 (0.006)	0.0012 (0.0028)	0.0044 (0.011)	0.0054 (0.13)	0.0017 (0.018)	0.022 (0.22)	0.1 (0.68)