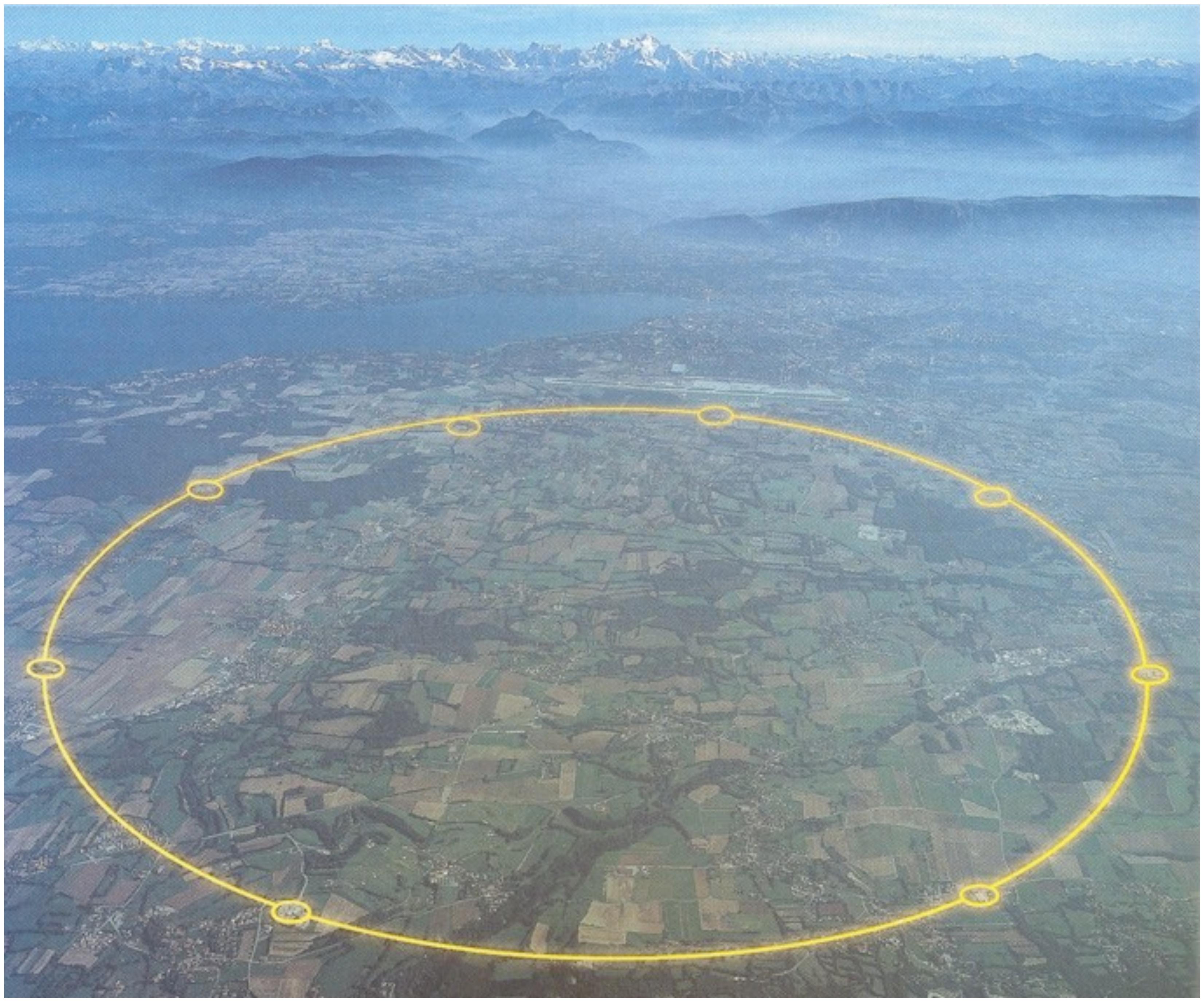
In the weeds of collider simulations: navigating negative weights

Rikkert Frederix
Lund University

Freiburg seminar, Wednesday, Feb. 7, 2024

Large Hadron Collider at CERN



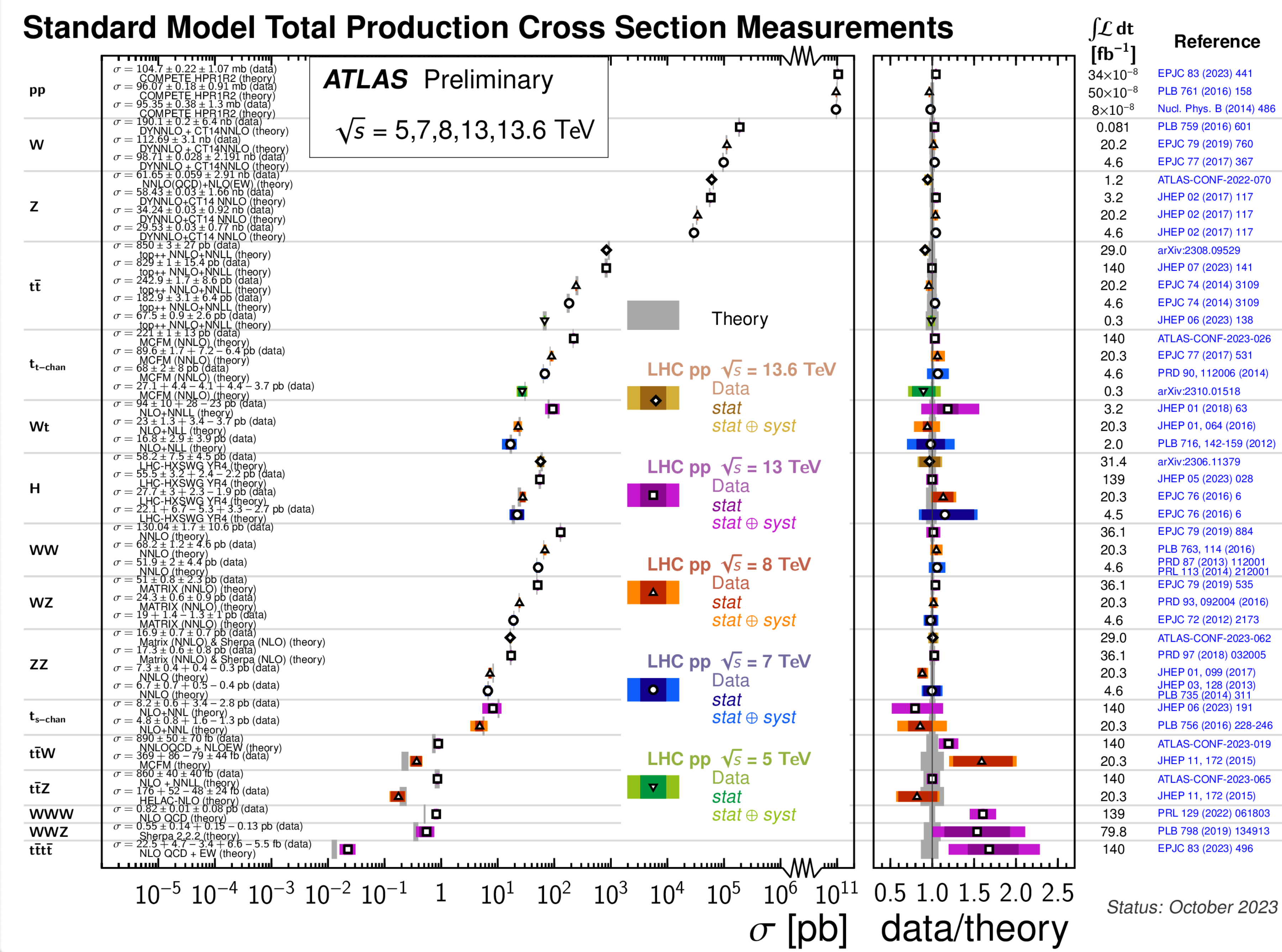
- Largest and most powerful particle accelerator in the world
- Collisions bring huge amounts of energy in a very tiny amount of space
  - $E=mc^2$
  - produces new particles
- Try to understand matter at smallest scales
- Discovery of the Higgs boson in 2012

Success of the LHC

- Searches for New Physics are relying more and more upon high-precision comparisons between theory and data
  - Large data samples, methods to reduce systematics
  - High precision computations
- We are scrutinising the Standard Model at higher and higher precision and in smaller and smaller corners of the phase-space
  - The ultimate stress-test for our predictions

Excellent agreement

- Excellent agreement between computed and measured cross sections
- for all accessible processes
- over many orders of magnitude



Bridging the gap

Experiment

Detector simulation
Pions, Kaons, ...
Reconstruction
B-tagging efficiency
...

Lagrangian
Gauge Invariance
Partons
Fixed Order Corrections
Resummation
...

Theory

An LHC collision: phenomenological picture

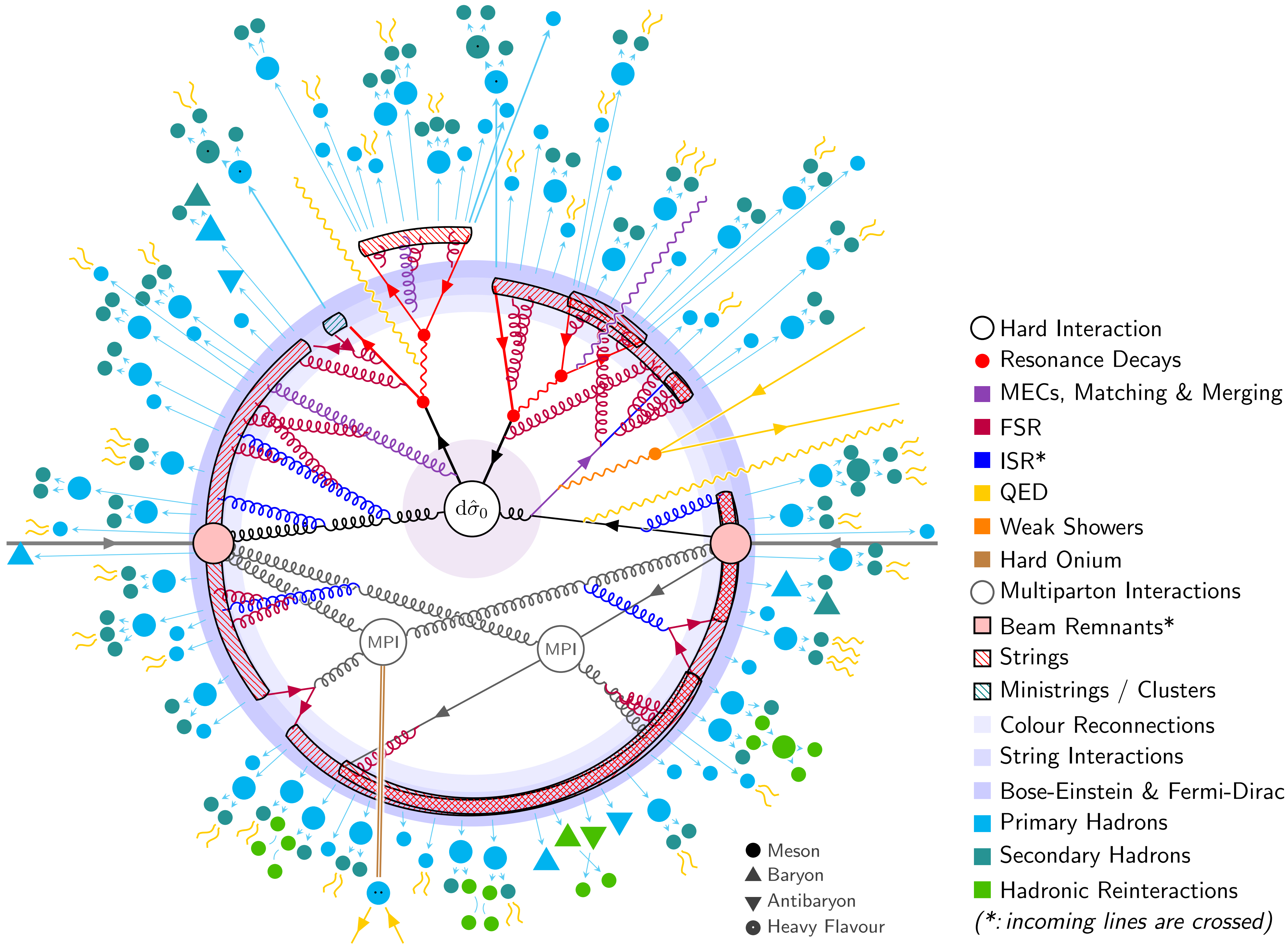


Figure taken from Bierlich et al., 2022 *(Pythia8.3 manual)*

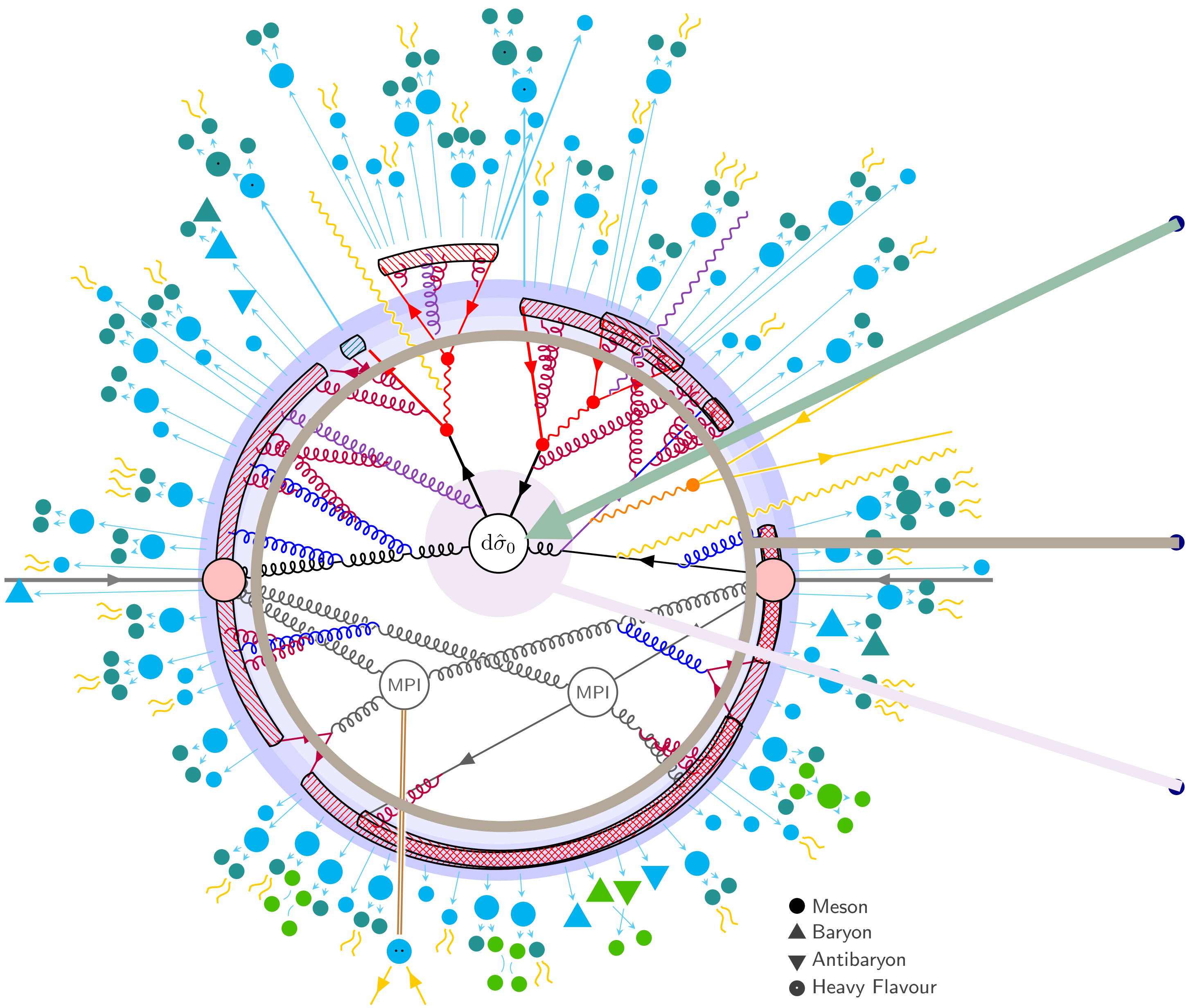
This talk

- Is not about the greatness of these simulation codes
- Is not a grant overview of their features

- It is about a little thing that soaked up an enormous amount of my time over last 5 years or so

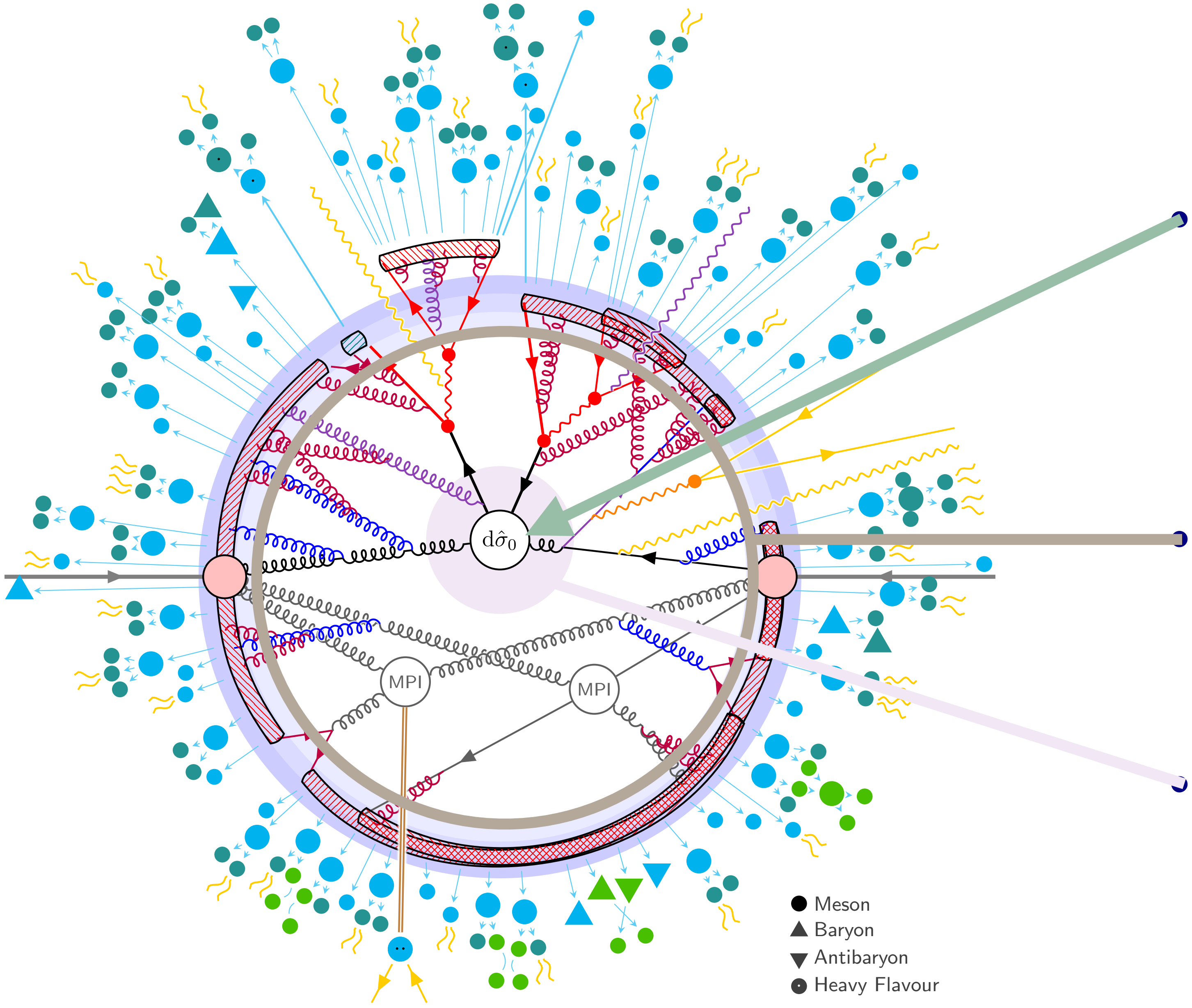
  - …negatively weighted events…

Matching hard ME with PS



- Hard matrix elements: Monte-Carlo integration over phase-space
(MadGraph5_aMC@NLO)
- Parton shower: Markov chain evolution
(Pythia8)
- MC@NLO (or POWHEG)

Matching hard ME with PS



Beyond lowest order in perturbation theory, the hard matrix elements are no longer positive definite in all phase-space points. MC@NLO predictions have "negatively weighted" events

Figure taken from Bierlich et al., 2022 *(Pythia8.3 manual)*

The cost of negative weights

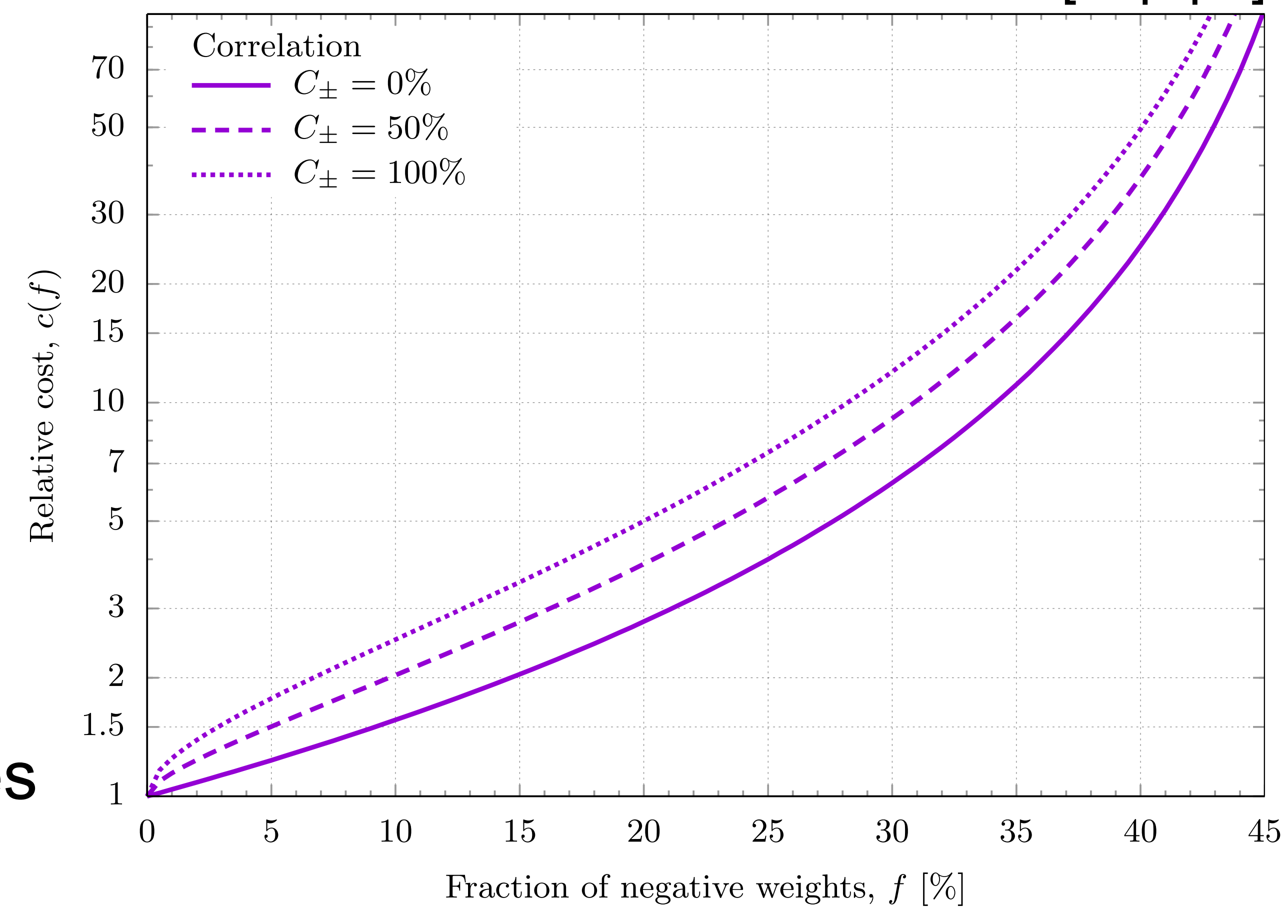
- Main disadvantage of MC@NLO is the (large) fraction of negatively weighted events
  - IR-safe observables will be positive in all bins
    (up to statistical fluctuations)
- Efficiency and relative cost:

$$\varepsilon(f) = 1 - 2f$$

$$c(f) = \frac{1 + C_\pm \sqrt{1 - \varepsilon(f)^2}}{\varepsilon(f)^2}$$

- Not only is there a cancelation between negative and positive events, the remaining distributions still have the statistical uncertainties of the original (larger) event files

RF et al. 2002.12716 [hep-ph]



MC@NLO anatomy

- Generating functional for MC@NLO

$$\mathcal{F}_{\mathrm{MC@NLO}} = \mathcal{F}_{\mathrm{MC}}\left(\mathcal{K}^{(\mathbb{H})}\right) d\sigma^{(\mathbb{H})} + \mathcal{F}_{\mathrm{MC}}\left(\mathcal{K}^{(\mathbb{S})}\right) d\sigma^{(\mathbb{S})}$$

MC (Shower) functional, starting from (n+1)-body kinematics

MC (Shower) functional, starting from n-body kinematics

with

real emission

shower counter terms

H-events: $d\sigma^{(\mathbb{H})} = d\sigma^{(\mathrm{NLO},E)} - d\sigma^{(\mathrm{MC})}\,,$

S-events: $d\sigma^{(\mathbb{S})} = d\sigma^{(\mathrm{MC})} + \sum_{\alpha=S,C,SC} d\sigma^{(\mathrm{NLO},\alpha)}$

Born, virtual, soft/collinear

MC@NLO
origins of negative weights

$$\mathcal{F}_{\mathrm{MC@NLO}} = \mathcal{F}_{\mathrm{MC}}\left(\mathcal{K}^{(\mathbb{H})}\right) d\sigma^{(\mathbb{H})} + \mathcal{F}_{\mathrm{MC}}\left(\mathcal{K}^{(\mathbb{S})}\right) d\sigma^{(\mathbb{S})}$$

$$d\sigma^{(\mathbb{H})} = d\sigma^{(\mathrm{NLO},E)} - d\sigma^{(\mathrm{MC})},$$

$$d\sigma^{(\mathbb{S})} = d\sigma^{(\mathrm{MC})} + \sum_{\alpha=S,C,SC} d\sigma^{(\mathrm{NLO},\alpha)}$$

- Three sources of negative weights (with some overlap in the first two)

N.1 $\mathbb{H}$ events with $P(\mathcal{K}^{(\mathbb{H})}) \ll M_H$;

N.2 $\mathbb{H}$ events with $P(\mathcal{K}^{(\mathbb{H})}) \sim M_H$;

Shower counter terms overestimate the real-emission corrections

N.3 $\mathbb{S}$ events.

n-body kinematics, but support in n+1-body phase space

Folding

S-events's support in n+1-body phase space

Born, virtual, soft/collinear

$$d\sigma^{(\mathbb{S})} = d\sigma^{(\mathrm{MC})} + \sum_{\alpha=S,C,SC} d\sigma^{(\mathrm{NLO},\alpha)}$$

$$d\sigma^{(\mathrm{MC})}(\Phi_B) = \int d\Phi_r K^{(\mathrm{MC})}(\Phi_B, \Phi_r)$$

$$d\sigma^{(\mathrm{NLO},\alpha)}(\Phi_B) = \alpha^{(\mathrm{NLO})}(\Phi_B)$$

- Monte-Carlo integration:
  - generate a random phase-space point in Φ_B
  - for a given Φ_B, generate a random point in Φ_r
- Since $K^{(\mathrm{MC})}$ and $\alpha^{(\mathrm{NLO})}$ are non-positive definite, negative events arise

Folding

$$d\sigma^{(\mathbb{S})} = d\sigma^{(\mathrm{MC})} + \sum_{\alpha=S,C,SC} d\sigma^{(\mathrm{NLO},\alpha)}$$

$$d\sigma^{(\mathrm{MC})}(\Phi_B) = \int d\Phi_r K^{(\mathrm{MC})}(\Phi_B, \Phi_r)$$

$$d\sigma^{(\mathrm{NLO},\alpha)}(\Phi_B) = \alpha^{(\mathrm{NLO})}(\Phi_B)$$

- Folding: for every Φ_B phase-space point, throw multiple Φ_r points
  - This smoothens the $K^{(\mathrm{MC})}$ contribution, reducing the number of negative weights
  - Φ_r contains 3 integration variables
  - Developed in the context of the POWHEG BOX generator
    P. Nason, arXiv:0709.2085
- Reduction significant, but at a considerable computational cost

	step-0 (s) (grid setup)	step-1 (s) (integration)	step-2 (s) (generation)	negative $\mathbb{S}$ weights
$\boxed{pp \to e^+e^-}$				
default	1	14	147	7.1%
$2 \times 2 \times 1$ folding	1	33	258	2.1%
$4 \times 4 \times 1$ folding	1	114	781	1.8%
$\boxed{pp \to H}$				
default	1	121	187	10.6%
$2 \times 2 \times 1$ folding	1	115	399	2.7%
$4 \times 4 \times 1$ folding	1	228	1190	0.6%
$\boxed{pp \to t\bar{t}}$				
default	2	132	455	8.6%
$2 \times 2 \times 1$ folding	2	262	1005	2.2%
$4 \times 4 \times 1$ folding	2	1092	3189	1.2%
$\boxed{pp \to W^+t\bar{t}}$				
default	5	346	1511	4.2%
$2 \times 2 \times 1$ folding	2	661	2938	2.2%
$4 \times 4 \times 1$ folding	2	2605	10020	1.7%
$\boxed{pp \to W^+j}$				
default	10	604	2013	24.2%
$2 \times 2 \times 1$ folding	10	1265	5160	13.2%
$4 \times 4 \times 1$ folding	7	2803	16020	9.0%
$\boxed{pp \to Hb\bar{b}}$				
default	77	1311	19440	27.3%
$2 \times 2 \times 1$ folding	39	4320	16380	22.4%
$4 \times 4 \times 1$ folding	48	17220	34260	20.9%

Nason, arXiv:0709.2085

Born spreading
—alternative to folding

Born spreading

$$d\sigma^{(\mathbb{S})} = d\sigma^{(\mathrm{MC})} + \sum_{\alpha=B,S,C,SC} d\sigma^{(\mathrm{NLO},\alpha)}$$

Born, virtual, soft/collinear

$$d\sigma^{(\mathrm{MC})}(\Phi_B) = \int d\Phi_r K^{(\mathrm{MC})}(\Phi_B, \Phi_r)$$

$$d\sigma^{(\mathrm{NLO},\alpha)}(\Phi_B) = \alpha^{(\mathrm{NLO})}(\Phi_B) \longrightarrow d\sigma^{(\mathrm{NLO},\alpha)}(\Phi_B) = \frac{\int \Phi_r \alpha^{(\mathrm{NLO})}(\Phi_B)}{\int \Phi_r}$$

$$d\sigma^{(\mathrm{NLO},B)}(\Phi_B) = \frac{\int \Phi_r \mathcal{F}(\Phi_r) B(\Phi_B)}{\int \Phi_r \mathcal{F}(\Phi_r)}$$

include $F(\Phi_r)$ for Born contribution only

- All contributions in $d\sigma^{(\mathbb{S})}$ now contain the integral of Φ_r
- Spreading function $F(\Phi_r)$ can be any arbitrary function
    - For simplicity, take it independent from (i.e., integrated over) Φ_B,
      i.e., we assume that the negatively weighted events are correlated strongly with the Φ_r dependence
  - Simple choice: since Born contribution is always positive we can
    - take $F(\Phi_r)$ to be zero where the rest of the contribution is already positive
    - and positive where the rest of the event is negative

RF and P. Torrielli, arXiv:2310.04160

Born spreading vs. Folding

- Define $F(\Phi_r)$ by filling a 3D (or 2D) grid, since it is integrated over Φ_B
- Significant reduction of negative weights
  - (albeit not as strong as folding)
- at a very modest computational cost
- Current setup not optimised: BSc student working on a more optimal $F(\Phi_r)$

	step-0 (s) (grid setup)	step-1 (s) (integration)	step-2 (s) (generation)	negative $\mathbb{S}$ weights
$\boxed{pp \to e^+e^-}$				
default	1	14	147	7.1%
$2 \times 2 \times 1$ folding	1	33	258	2.1%
$4 \times 4 \times 1$ folding	1	114	781	1.8%
Born spreading	113	30	189	2.0%
$\boxed{pp \to H}$				
default	1	121	187	10.6%
$2 \times 2 \times 1$ folding	1	115	399	2.7%
$4 \times 4 \times 1$ folding	1	228	1190	0.6%
Born spreading	82	122	203	1.1%
$\boxed{pp \to t\bar{t}}$				
default	2	132	455	8.6%
$2 \times 2 \times 1$ folding	2	262	1005	2.2%
$4 \times 4 \times 1$ folding	2	1092	3189	1.2%
Born spreading	199	137	448	2.1%
$\boxed{pp \to W^+t\bar{t}}$				
default	5	346	1511	4.2%
$2 \times 2 \times 1$ folding	2	661	2938	2.2%
$4 \times 4 \times 1$ folding	2	2605	10020	1.7%
Born spreading	202	741	2138	2.6%
$\boxed{pp \to W^+j}$				
default	10	604	2013	24.2%
$2 \times 2 \times 1$ folding	10	1265	5160	13.2%
$4 \times 4 \times 1$ folding	7	2803	16020	9.0%
Born spreading	355	645	2226	18.8%
$\boxed{pp \to Hb\bar{b}}$				
default	77	1311	19440	27.3%
$2 \times 2 \times 1$ folding	39	4320	16380	22.4%
$4 \times 4 \times 1$ folding	48	17220	34260	20.9%
Born spreading	578	1263	20760	24.7%

RF and P. Torrielli, arXiv:2310.04160

Recap

- A source of negative weights in an MC@NLO computation is from the S-events
- It is an contribution differential in the n-body (Born) phase-space, but with support in the (n+1)-body phase-space

- **Folding** smoothens the integral over the additional radiative phase-space by trowing more points for the latter for a given n-body phase-space point

- **Born Spreading** moves the Born contribution into the (n+1)-body phase-space, and most strongly where the latter is negative. Since the Born contribution is always positive, it reduces the negative contributions

- Both the original and these new methods yield strictly identical results (within statistical fluctuations), although with a reduction of negative weights

MC@NLO-Δ

MC@NLO origins of negative weights

$$\mathcal{F}_{\rm MC@NLO} = \mathcal{F}_{\rm MC}\left(\mathcal{K}^{(\mathbb{H})}\right) d\sigma^{(\mathbb{H})} + \mathcal{F}_{\rm MC}\left(\mathcal{K}^{(\mathbb{S})}\right) d\sigma^{(\mathbb{S})}$$

$$d\sigma^{(\mathbb{H})} = d\sigma^{(\rm NLO,\mathit{E})} - d\sigma^{(\rm MC)},$$

$$d\sigma^{(\mathbb{S})} = d\sigma^{(\rm MC)} + \sum_{\alpha=S,C,SC} d\sigma^{(\rm NLO,\alpha)}$$

- Three sources of negative weights (with some overlap in the first two)

N.1 $\mathbb{H}$ events with $P(\mathcal{K}^{(\mathbb{H})}) \ll M_H$;

N.2 $\mathbb{H}$ events with $P(\mathcal{K}^{(\mathbb{H})}) \sim M_H$;

N.3 $\mathbb{S}$ events.

Shower counter terms overestimate the real-emission corrections

n-body kinematics, but support in n+1-body phase space

Type N.2

N.1 $\mathbb{H}$ events with $P(\mathcal{K}^{(\mathbb{H})}) \ll M_H$;

N.2 $\mathbb{H}$ events with $P(\mathcal{K}^{(\mathbb{H})}) \sim M_H$;

N.3 $\mathbb{S}$ events.

$$\mathcal{F}_{\mathrm{MC@NLO}} = \mathcal{F}_{\mathrm{MC}}\Big(\mathcal{K}^{(\mathbb{H})}\Big)\, d\sigma^{(\mathbb{H})} + \mathcal{F}_{\mathrm{MC}}\Big(\mathcal{K}^{(\mathbb{S})}\Big)\, d\sigma^{(\mathbb{S})}$$

$$d\sigma^{(\mathbb{H})} = d\sigma^{(\mathrm{NLO},E)} - d\sigma^{(\mathrm{MC})}\,,$$

$$d\sigma^{(\mathbb{S})} = d\sigma^{(\mathrm{MC})} + \sum_{\alpha=S,C,SC} d\sigma^{(\mathrm{NLO},\alpha)}$$

- Reduction of negative events of type N.2
  - The shower is radiating into the hard region; fine for LO, but at NLO one emission is explicitly included through real-emission matrix elements
    $\Longrightarrow$ prefer smaller shower starting scales

MC@NLO-Δ

N.1 $\mathbb{H}$ events with $P(\mathcal{K}^{(\mathbb{H})}) \ll M_H$;

N.2 $\mathbb{H}$ events with $P(\mathcal{K}^{(\mathbb{H})}) \sim M_H$;

N.3 $\mathbb{S}$ events.

$$\mathcal{F}_{\rm MC@NLO} = \mathcal{F}_{\rm MC}\left(\mathcal{K}^{(\mathbb{H})}\right) d\sigma^{(\mathbb{H})} + \mathcal{F}_{\rm MC}\left(\mathcal{K}^{(\mathbb{S})}\right) d\sigma^{(\mathbb{S})}$$

$$d\sigma^{(\mathbb{H})} = d\sigma^{({\rm NLO},E)} - d\sigma^{({\rm MC})}\,,$$

$$d\sigma^{(\mathbb{S})} = d\sigma^{({\rm MC})} + \sum_{\alpha=S,C,SC} d\sigma^{({\rm NLO},\alpha)}$$

- Reduction of negative events of type N.1
  - Modify the MC@NLO procedure:

$$\mathcal{F}_{\rm MC@NLO\text{-}\Delta} = \mathcal{F}_{\rm MC}\left(\mathcal{K}^{(\mathbb{H})}\right) d\sigma^{(\Delta,\mathbb{H})} + \mathcal{F}_{\rm MC}\left(\mathcal{K}^{(\mathbb{S})}\right) d\sigma^{(\Delta,\mathbb{S})}$$

$$d\sigma^{(\Delta,\mathbb{H})} = \left(d\sigma^{({\rm NLO},E)} - d\sigma^{({\rm MC})}\right)\Delta\,,$$

$$d\sigma^{(\Delta,\mathbb{S})} = d\sigma^{({\rm MC})}\Delta + \sum_{\alpha=S,C,SC} d\sigma^{({\rm NLO},\alpha)} + d\sigma^{({\rm NLO},E)}\left(1-\Delta\right)$$

  - with

    $\Delta \longrightarrow 0$ in the soft/collinear limits

    $\Delta \longrightarrow 1$ in the hard regions

    $\Longrightarrow$ use shower no-emission probability (between hard scale and scale of the emission)

Δ dampens the contribution from the H-events in the soft/collinear region, and adds it the S-event contribution.
The idea: the shower will do a good job to re-fill the phase-space

NLO accuracy

- The formal expansion of the no-emission probability is $\Delta = 1 + \mathcal{O}(\alpha_S)$
- Furthermore, in the soft/collinear limits the logarithms are similar to the ones that are generated by the shower

$\Longrightarrow$ From this one can conclude that accuracy is the same as with the default MC@NLO method

- However, beyond NLO contributions can be very much different: MC@NLO-Δ is effectively a new matching procedure
  i.e., results will NOT be identical between the original and new predictions

Implementation

- Run-time interface between MG5_aMC and Pythia8
  - MG5_aMC generates phase-space points, all the relevant matrix elements and MC counter terms
  - It calls Pythia8 to determine the relevant emission scales for each dipole in the S-events to obtain the H-event
  - For fast evaluation, the Pythia8 Sudakov factors have been tabulated (2D grids (dipole mass and scale); one for each parton flavour; one for each dipole type (II, IF, FI, FF))
  - No emission probabilities included by MG5_aMC
- Major complication:
  - emission scale is different for each dipole
  - when showering events requires a different starting scale for each dipole

Reduction of negative weights

	MC@NLO			MC@NLO-Δ		
	111	212	414	111-Δ	212-Δ	414-Δ
$pp \to e^+e^-$	7.1% (1.4)	3.6% (1.2)	3.3% (1.1)	5.7% (1.3)	2.9% (1.1)	2.8% (1.1)
$pp \to e^+\nu_e$	7.3% (1.4)	4.0% (1.2)	3.7% (1.2)	6.1% (1.3)	3.5% (1.2)	3.3% (1.1)
$pp \to h$	10.4% (1.6)	4.8% (1.2)	3.4% (1.2)	4.2% (1.2)	1.3% (1.1)	0.4% (1.0)
$pp \to t\bar{t}$	22.9% (3.4)	20.0% (2.8)	19.5% (2.7)	10.1% (1.6)	5.1% (1.2)	4.4% (1.2)
$pp \to W^+t\bar{t}$	16.0% (2.2)	15.0% (2.0)	15.2% (2.1)	11.6% (1.7)	10.4% (1.6)	10.2% (1.6)
$pp \to hb\bar{b}$	40.1% (25.7)	37.9% (17.1)	37.4% (15.7)	36.7% (14.2)	31.2% (7.0)	29.9% (6.2)
$pp \to W^+j[50]$	21.7% (3.1)	16.3% (2.2)	15.7% (2.1)	15.4% (2.1)	7.7% (1.4)	6.0% (1.3)

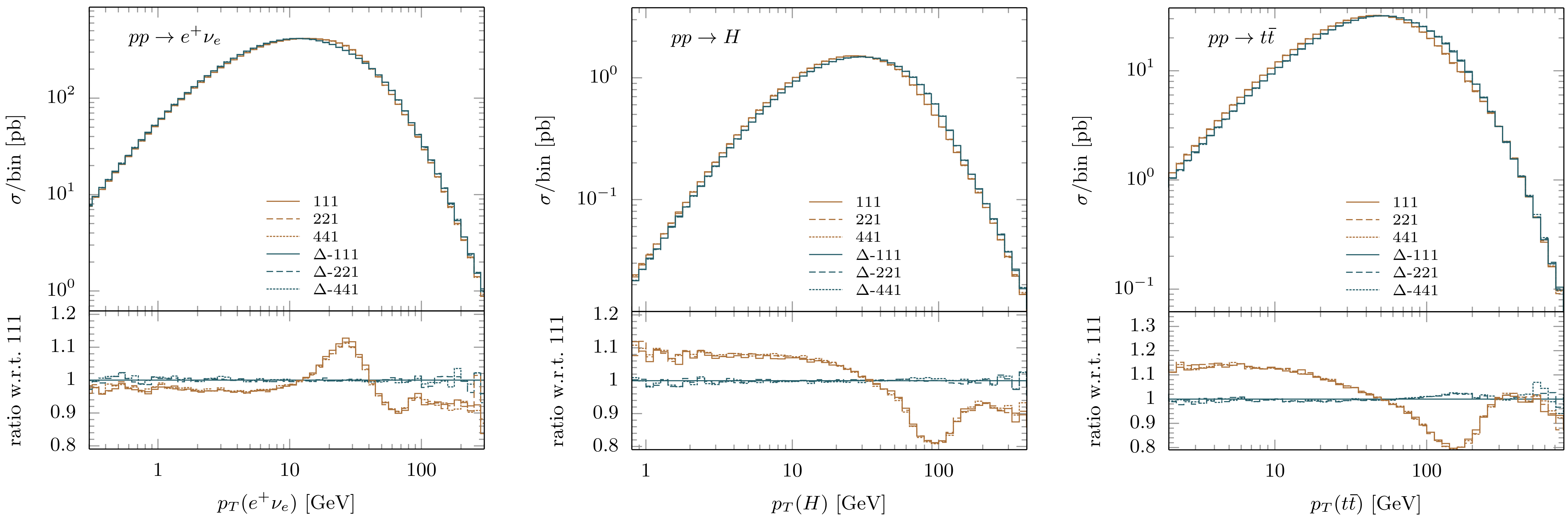
no folding

4x folding

16x folding

- Fraction of negative weights and relative cost (assuming no correlations)

Selected results



- Transverse momentum of the Born system
- Differences between default and Δ are sizeable, but reasonable
- Folding has no effect (apart from increased statistics), as it should be

8 years of development…

- First ideas discussed in 2015
  together with Stefano Frixione, Stefan Prestel, Paolo Torrielli
- Serious work started in 2017
- Published MC@NLO-Δ in 2002.12716 [hep-ph], but code did not go public
  - After publication, we found
    - some bugs…
    - a better treatment of events that are in the dead zone
    - some improvements in the shower scale assignments
    - compatibility with Pythia8.3
      (thanks to Leif Gellersen and Christian Preuss!)
- Code public: 2023

Summary

- Comparisons between LHC data and predictions show excellent agreement
  - The tools are optimised, but there are always improvements possible
- One major drawback of combining higher-order Matrix Element computations with Parton Showers are the event-by-event negative weight contributions that only cancel in (IR-safe) observables

- MC@NLO-Δ reduces the number of negative weights by a significant amount
  - New matching procedure; results differ from default MC@NLO—within the matching systematics
  - Run-time interface between MG5_aMC and Pythia8
    - With Δ enabled, CPU time to generate events increases by a factor ~3
    - 4x folding increases the run time also by a factor ~3
    - 16x folding about a factor ~3^2

    ⟹ reduction of negative weights due to Δ and folding typically not worth it from a CPU point of view, except when there is more overhead than simply showering the events (detector simulation, storage space, etc.)

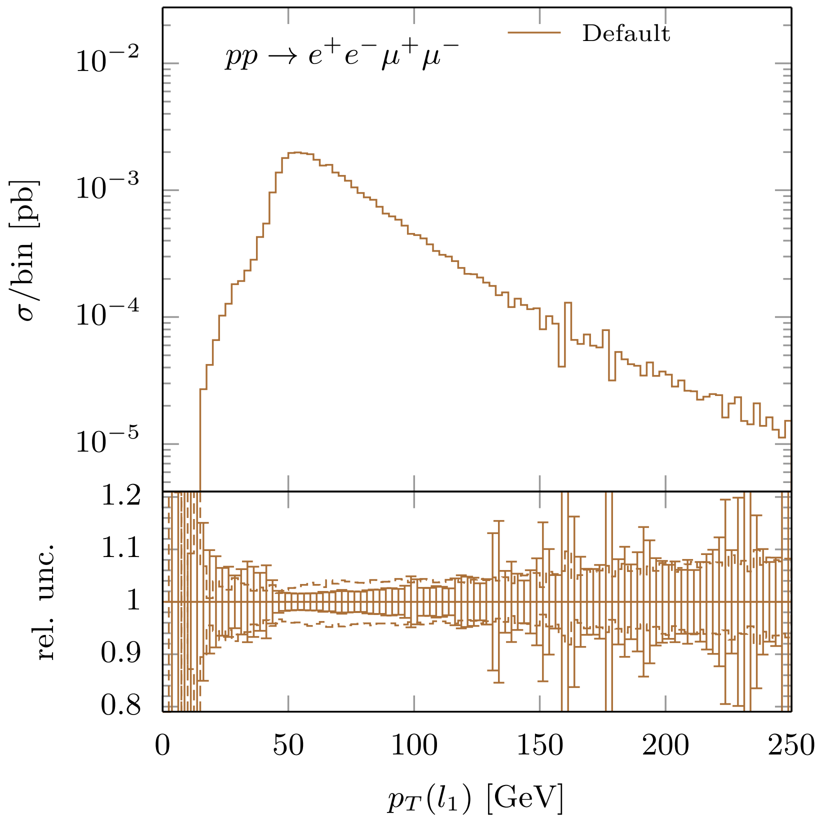
    - but with Born Spreading it probably is!
      (spreading needs to be further optimised, though)

BONUS:

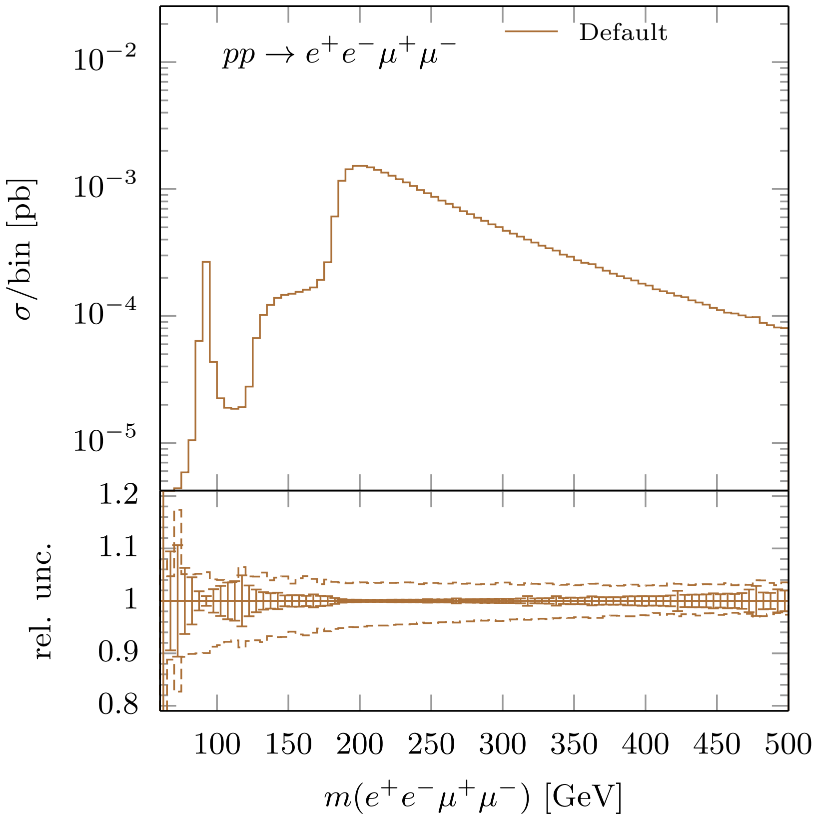
Reducing statistical fluctuations at Fixed Order

Statistical fluctuations at fixed order

- A major source of statistical fluctuations in fixed order differential distributions are the 'misbinning' effects
  - At NLO: the real-emission and IR-subtraction terms can end up in different bins
  - This depends on the mapping between the n and (n+1)-body phase-space



large statistical fluctuations
for lepton p_T's



no misbinning for
e.g. 4-lepton invariant mass

Δ for Fixed order

- NLO diff. cross section (schematically)

$$\frac{d\sigma^{\rm NLO}}{d\mathcal{O}} = \int (B+V+I)\mathcal{O}(\Phi_n)\,d\Phi_n + \int (R\mathcal{O}(\Phi_{n+1}) - S\mathcal{O}(\Phi_n))\,d\Phi_{n+1}$$

- Introduce Δ factor:

$$\frac{d\sigma^{\rm NLO}}{d\mathcal{O}} = \int (B+V+I)\mathcal{O}(\Phi_n)\,d\Phi_n + \int \Big[(R\mathcal{O}(\Phi_{n+1}) - R\mathcal{O}(\Phi_n))\Delta + R\mathcal{O}(\Phi_n) - S\mathcal{O}(\Phi_n) \Big]\, d\Phi_{n+1}$$

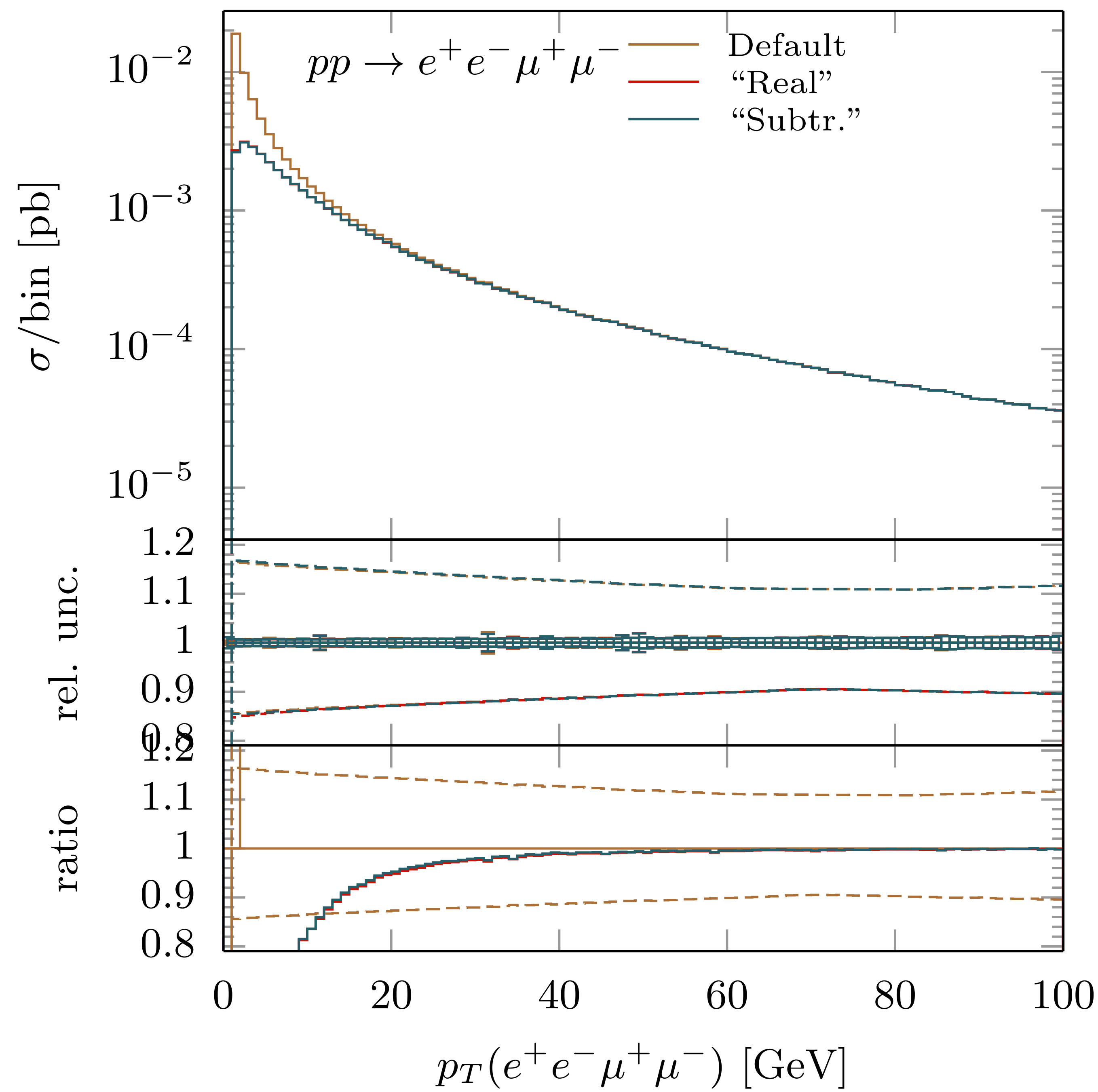
Alternatively:

$$\frac{d\sigma^{\rm NLO}}{d\mathcal{O}} = \int (B+V+I)\mathcal{O}(\Phi_n)\,d\Phi_n + \int \Big[R\mathcal{O}(\Phi_{n+1}) - S\mathcal{O}(\Phi_{n+1}) + (S\mathcal{O}(\Phi_{n+1}) - S\mathcal{O}(\Phi_n))\Delta \Big]\, d\Phi_{n+1}$$

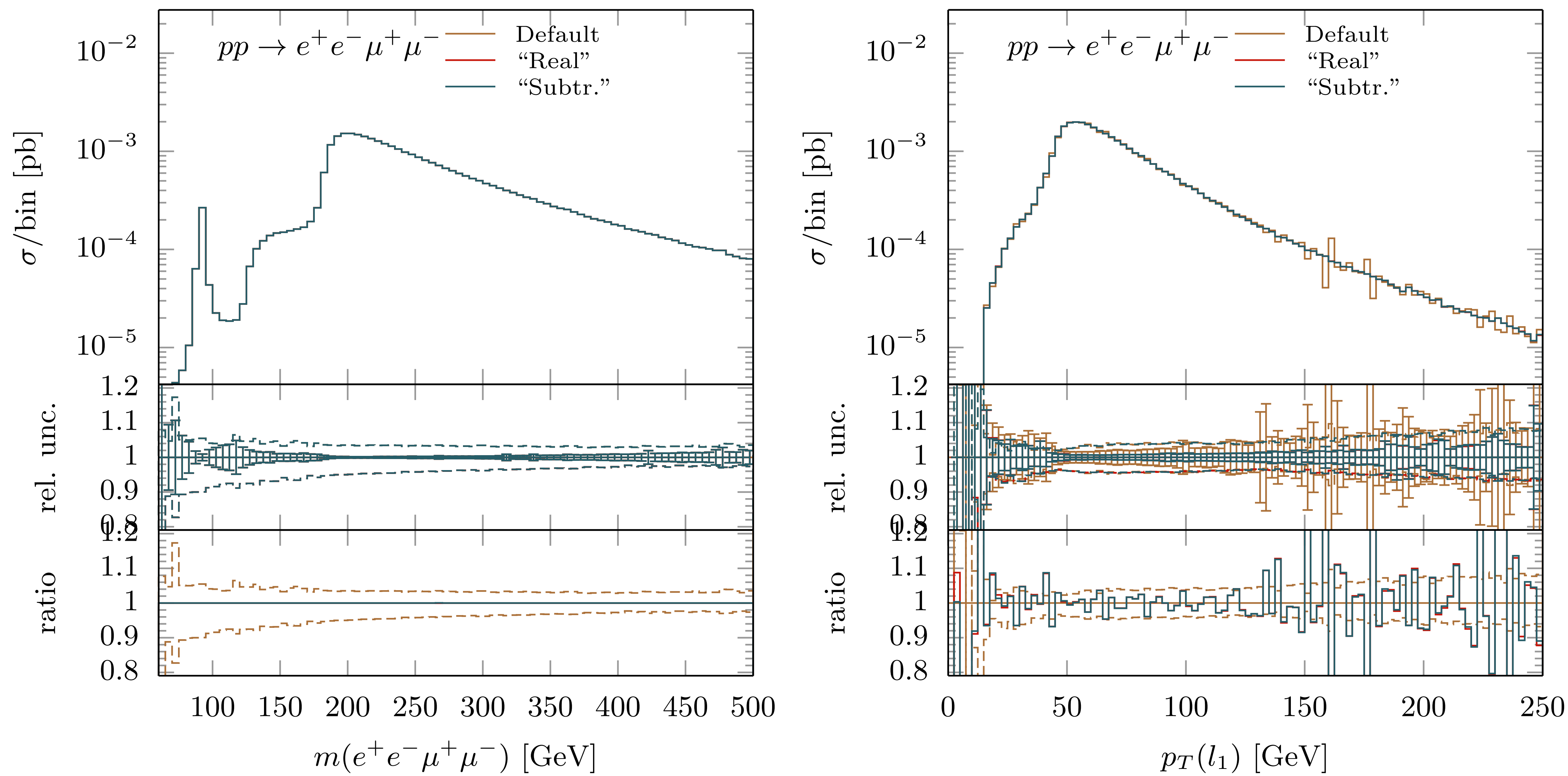
- Δ does not need to be the Pythia8 no-emission probability: it can be a simple LL Sudakov factor between the hard scale and the scale of the emission
$\Longrightarrow$ NLO accuracy is conserved

p_T(4lepton)

- Same random seed: exactly the same PS points
- Inclusion of Δ changes the 4-lepton spectrum at small transverse momenta
- However, this is the region where you cannot trust FO perturbation theory
- Two versions (including Δ for real and subtraction, respectively) give identical results

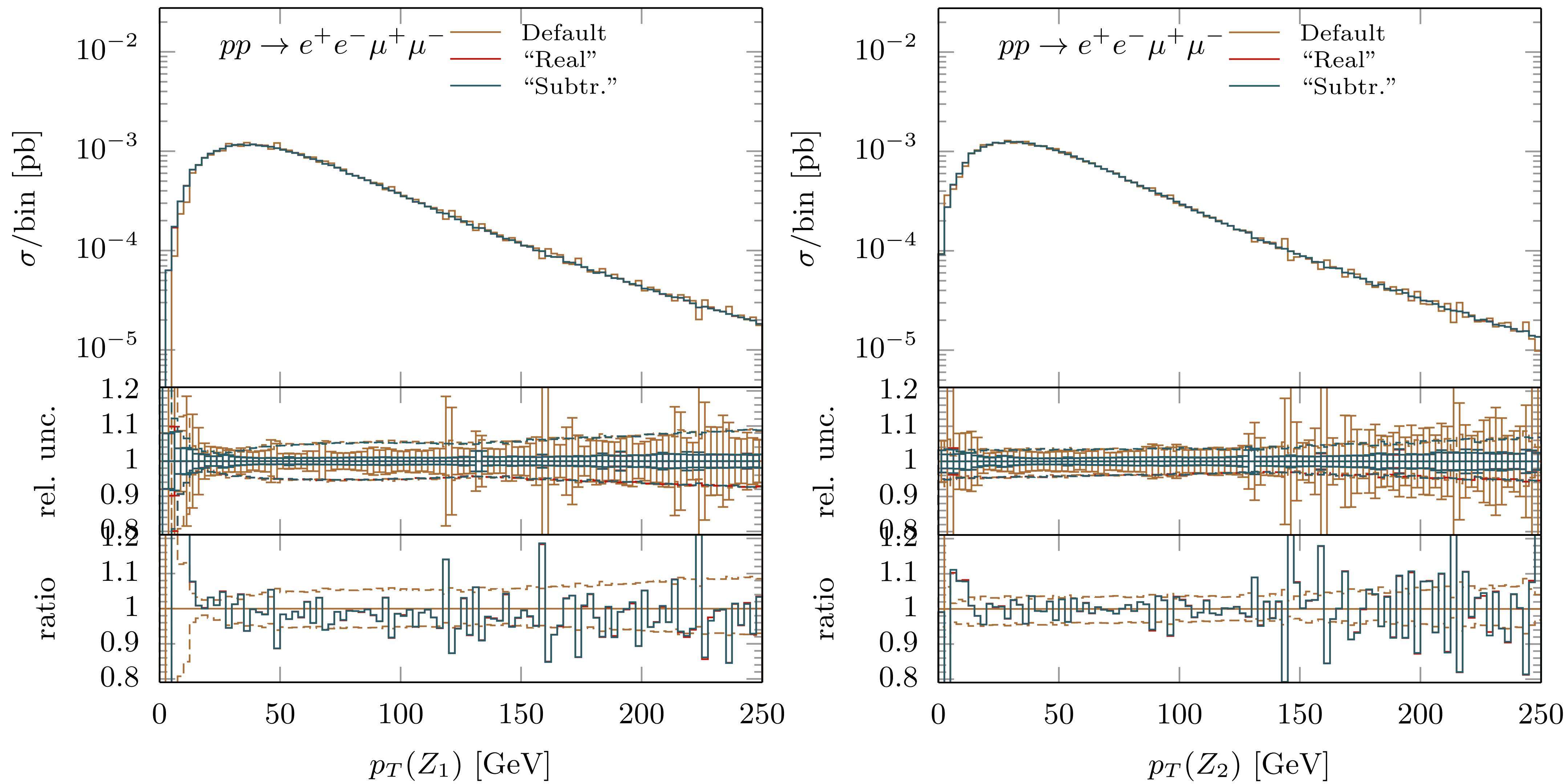


Hardest lepton and invariant mass



- Of course, no effect in 4-lepton invariant mass
- Significant reduction of statistical fluctuations in $p_T(l_1)$; compatible with the default

Z-bosons (ordered in p_T)



- Again, large reduction in statistical fluctuations
- Compatible within scale uncertainties;
  expect at low $p_T(Z)$, where FO perturbation theory cannot be trusted

Summary: BONUS

- Adding Δ is a simple improvement to fixed order computations that can significantly reduce statistical fluctuations in diff. distributions
- Inclusive rates are not affected
  ⟹ Observables conserved in the mapping are not affected
- Other observables see some changes, but only when sensitive to IR region, where fixed order perturbation theory does not work
  ⟹ Statistical fluctuations reduced by factor ~2-3 at no additional cost.
  Severe misbinning is gone
- Worth investigating for other processes and at NNLO