Universität
Münster

Matching and merging
Part 1: Matching (continued)

Tomáš CJežo (tomas.jezo@uni-muenster.de)

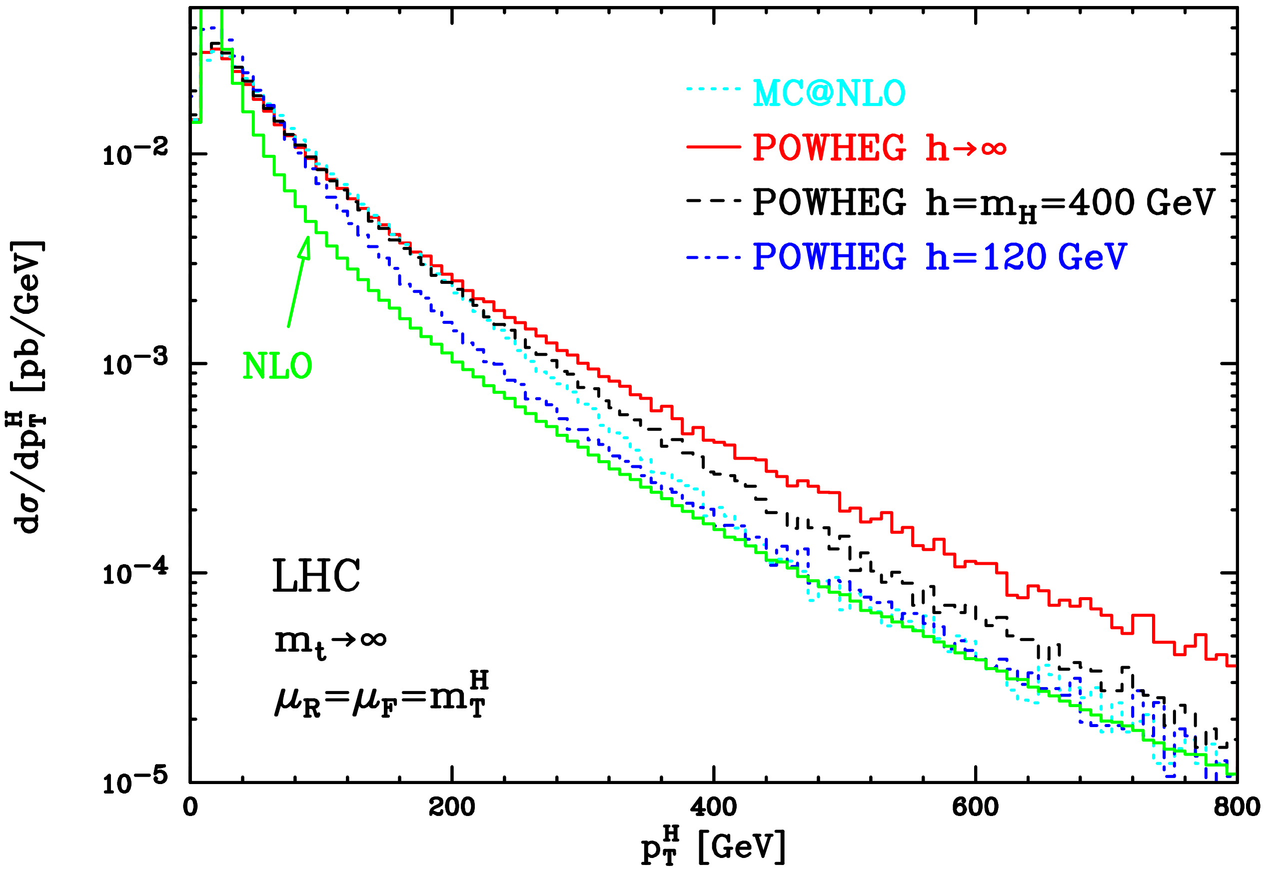
institut für
theoretische physik

28th November 2025

Summary from the last lecture

- Matching = combining fixed-order NLO with parton showers
  - keep inclusive NLO accuracy and LL resummation
  - avoid double counting between real matrix elements and shower emissions
- Matching schemes intro
  - additive (MC@NLO-type): NLO result + [PS − approximate real], negative weights but FO structure is transparent
  - multiplicative (POWHEG-type): generate hardest emission with R/B and a Sudakov, effectively reweighting LO+PS by an NLO K-factor
- POWHEG method
  - from LO+PS to POWHEG formula
  - singular regions and $\Phi_B \leftrightarrow (\Phi_R, \Phi_{\rm rad})$
  - Sudakov veto loop
- Tuning the real cross section
  - split $R = R_s + R_f$ with `hdamp` / `bornzerodamp` so that only the genuinely singular part R_s is exponentiated
  - control the high-p_T tail and scale dependence in the remnant R_f

Tuning the real cross section



Outline, Part 1: Continued

Resonance-aware POWHEG

- Resonance histories
- Multiple-radiation scheme

POWHEG BOX V2/RES

- POWHEG method implementations
- From POWHEG events to parton shower

Resonance-aware POWHEG method (POWHEG RES)

Global recoil

- In standard POWHEG, each real configuration is mapped to an underlying Born: $\Phi_R \leftrightarrow (\Phi_B, \Phi_{\text{rad}}^{\alpha})$ for each singular region α
  - This mapping redistributes real emission recoil globally and so in general **does not preserve internal resonance virtualities**

- If a narrow resonance r is on-shell in the real kinematics:
  - real: $M_r^2(\Phi_R) \simeq m_r^2$ (Breit–Wigner peak)
  - mapped Born: $M_r^2(\Phi_B) \neq m_r^2$ in general
  - the recoil that removes $\Phi_{\text{rad}}^{\alpha}$ shifts M_r^2 off-shell

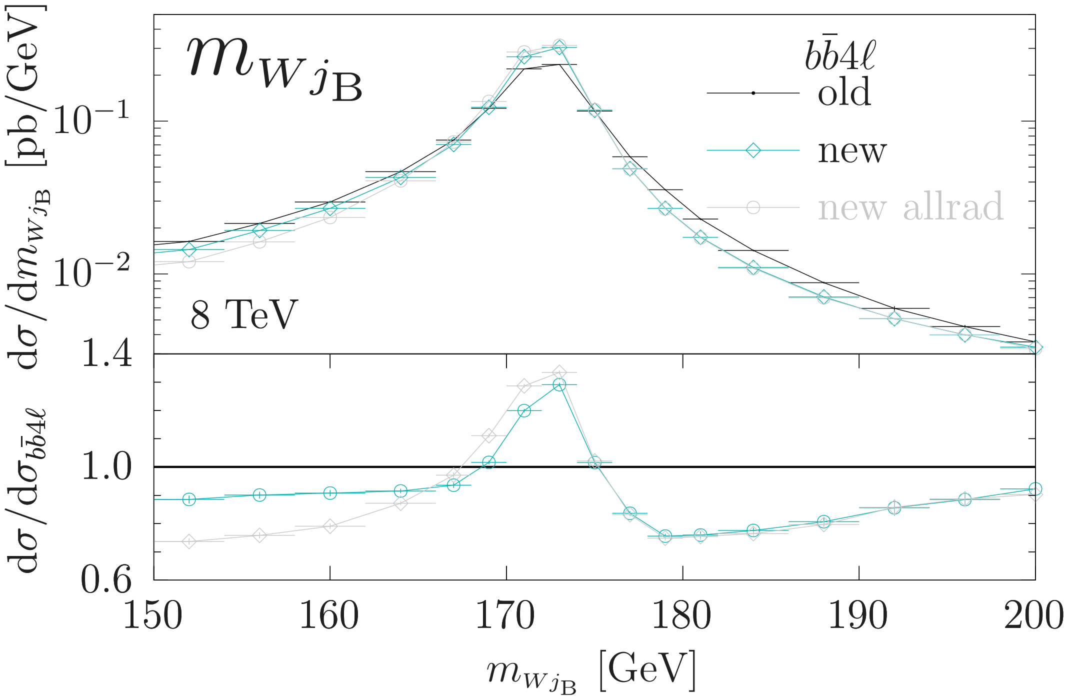
Consequence

- Large R/B means Sudakov is close to zero
  - we get artificial depletion on the peak regardless of hardness of the emission
  - resonance line shapes are distorted

Resonance-aware POWHEG method (POWHEG RES)

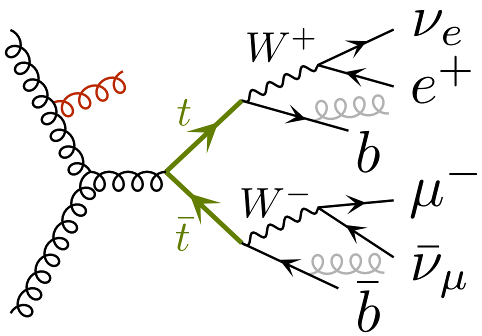
Consequence

- Large R/B means Sudakov is close to zero
  - we get artificial depletion on the peak regardless of hardness of the emission
  - resonance line shapes are distorted

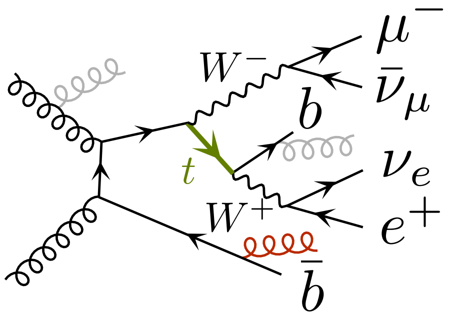


How to overcome this?[†]

1. Introduce resonance histories
   - Decompose each configuration into contributions with definite internal resonances (t, W, H, ...)
   - Each parton is attributed either to production or to the decay of a specific resonance
   - Example:




```
[21,21,6,-6,24,-24,-11,12,5,13,-14,5,21]
[ 0, 0,0, 0, 3,  4,  5, 5,3, 6,  6,4, 0]
```



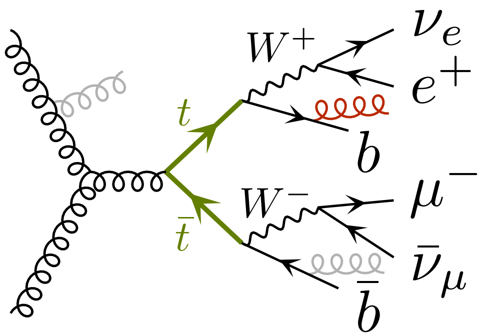

```
[21,21,6,24,-24,-11,12,5,13,-14,5,21]
[ 0, 0,0, 3,  0,  4, 4,3, 5,  5,0, 0]
```

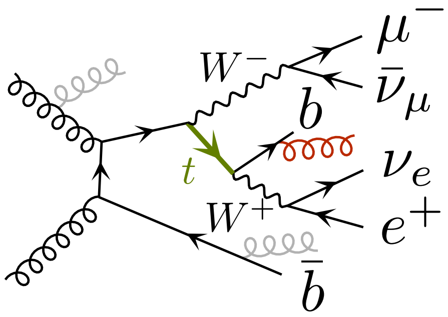
[†] [Ježo et al., JHEP 12 (2015) 065], [Ježo et al., JHEP 10 (2023) 008]

Resonance-aware POWHEG method (POWHEG RES)

How to overcome this?†

1. Introduce resonance histories
   - Decompose each configuration into contributions with definite internal resonances ($t, W, H, \ldots$)
   - Each parton is attributed either to production or to the decay of a specific resonance
   - Example:






```
[21,21,6,-6,24,-24,-11,12,5,13,-14,5,21]
[ 0, 0,0, 0, 3,  4,  5, 5,3, 6,  6,4, 3]
```

```
[21,21,6,24,-24,-11,12,5,13,-14,5,21]
[ 0, 0,0, 3,  0,  4, 4,3, 5,  5,0, 3]
```

† [Ježo et al., JHEP 12 (2015) 065], [Ježo et al., JHEP 10 (2023) 008]

Resonance-aware POWHEG method (POWHEG RES)

How to overcome this?[†]

1. Introduce resonance histories
   - Decompose each configuration into contributions with definite internal resonances ($t, W, H, \ldots$)
   - Each parton is attributed either to production or to the decay of a specific resonance
2. Resonance-preserving mappings (used throughout the POWHEG fomula)
   - For each history and each singular region α, define $\Phi_R \leftrightarrow \left(\Phi_B, \Phi_{\rm rad}^{\{\alpha\}}\right)$ such that $M_r^2(\Phi_R) = M_r^2(\Phi_B)$ for all resonances r in that history
   - Recoil distributed within the production system or a given decay system, not across resonances
   - Example: mapping from $t \to bgW^+$ to $t \to bW^+$
     - go to the top rest frame, recombine $b + g$ to set the b' direction, apply on-shell $1 \to 2$ kinematics $t \to b'W^{+'}$, then boost back to the lab frame

[†][Randall et al., JHEP 12 (2015) 065], [Randall et al., JHEP 10 (2023) 008]

Resonance-aware POWHEG method (POWHEG RES)

How to overcome this?[†]

1. Introduce resonance histories
   - Decompose each configuration into contributions with definite internal resonances ($t, W, H, \ldots$)
   - Each parton is attributed either to production or to the decay of a specific resonance
2. Resonance-preserving mappings (used throughout the POWHEG fomula)
   - For each history and each singular region α, define $\Phi_R \leftrightarrow \left(\Phi_B, \Phi_{\text{rad}}^{\{\alpha\}}\right)$ such that $M_r^2(\Phi_R) = M_r^2(\Phi_B)$ for all resonances r in that history
   - Recoil distributed within the production system or a given decay system, not across resonances
3. Resonance-aware FKS[‡] subtraction
   - new mappings are resonance history dependent: $\overline{B}$ and R/B must be partitioned
   - within one resonance history they are basically the original FKS mappings but now applied in resonance rest frames instead of the lab frame
   - FKS needs re-deriving, since the reference frame is no longer fixed: `softmismatch`

[†] [Ježo et al., JHEP 12 (2015) 065], [Ježo et al., JHEP 10 (2023) 008]

[‡] This is also available in CS [Höche et al., Eur.Phys.J.C 79 (2019) 728]

Resonance histories (nay, singular regions on steroids)

- In standard POWHEG, singular regions α defined only by collinearity of external legs:
  - radiation from a b in the top decay can still compete with ISR gluon
- In POWHEG RES, each singular region is refined to a pair (α, h):
  - h = resonance history
  - an emission in resonance decay only competes with other emissions in the same decay

Resonance histories (nay, singular regions on steroids)

- In standard POWHEG, singular regions α defined only by collinearity of external legs:
  - radiation from a b in the top decay can still compete with ISR gluon
- In POWHEG RES, each singular region is refined to a pair (α, h):
  - h = resonance history
  - an emission in resonance decay only competes with other emissions in the same decay
- Additional resonance-history–phase-space partition:
  - for each real phase-space point Φ_R construct weights $\Pi_{h(\Phi_R)}$ with $\Pi_{h(\Phi_R)} \geq 0$ and $\sum_h \Pi_{h(\Phi_R)} = 1$
  - per default: Π_h are built from Breit–Wigner–like factors for the resonances in history h (top, W, ...), so the history with resonances closest to shell dominates

$$\Pi_{h_{t\bar{t}}}(\Phi) = \frac{w_{h_{t\bar{t}}}(\Phi)}{\sum_{h'} w_{h'}(\Phi)}, \quad w_{h_t\bar{t}}(\Phi) = F_t\Big(M_t^2(\Phi)\Big)F_{\bar{t}}\Big(M_{\bar{t}}^2(\Phi)\Big)F_{W^+}\Big(M_{W^+}^2(\Phi)\Big)F_{W^-}\Big(M_{W^-}^2(\Phi)\Big)$$

$$F_R\Big(M_R^2(\Phi)\Big) = m_R^4 \Big/ \left(\Big(M_R^2(\Phi) - m_R^2\Big)^2 + m_R^2\Gamma_R^2\right)$$

  - for $t\bar{t}$: there is also a custom matrix element based separation
  - the real matrix element is split as $R(\Phi_R) = \sum_h R^{h(\Phi_R)}$ with $R^{h(\Phi_R)} = \Pi_{h(\Phi_R)} R(\Phi_R)$, and then further into singular pieces $R_\alpha^{h(\Phi_R)}$ for POWHEG

Multiple-radiation scheme

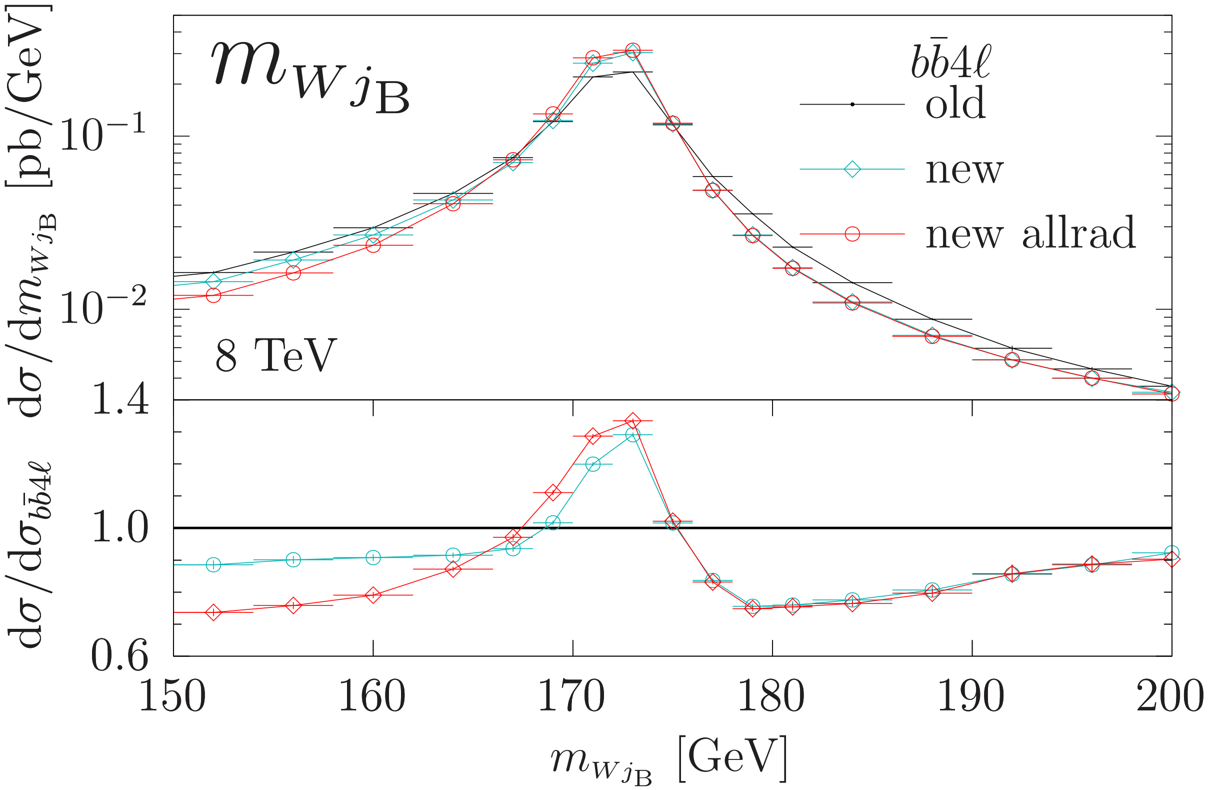
- Motivation
  - In standard NLO+PS matching only one hardest emission is corrected
  - In $pp \to t\bar{t}$ the hardest emission is almost always ISR, so radiation in decays is down to PS
- Idea
  - We can keep up to $n_h + 1$ POWHEG emissions: one from production ($r = 0$) and one from each resonance decay ($h = 1, \ldots, n_h$)

$$d\sigma = \overline{B}(\Phi_B) \prod_{h=0}^{n_h} \left[\Delta_h(t_0) + \int d\Phi_{\rm rad}^h \Delta_h\left(p_T^h\left(\Phi_{\rm rad}^h\right)\right) \frac{R_h\left(\Phi_B, \Phi_{\rm rad}^h\right)}{B(\Phi_B)} \right]$$

  - One emission – the usual NLO correction, multiple emissions – formally higher order
  - Product over h turns into a sum upon expansion $\prod_h [\Delta_h(t_0) + \cdots] = \Delta_h(t_0) + \sum_h R_{h,c} \frac{\Phi_{R,c}}{B_{h(\Phi_B)}}, d\Phi_{\rm rad,c}$
- Why the extra higher orders are useful
  - Observables built from decay-product kinematics (e.g. lepton angles, reconstructed resonance masses), NLO corrections in **production** act largely as a flat K-factor

Multiple-radiation scheme

- Why the extra higher orders are useful
  - Observables built from decay-product kinematics (e.g. lepton angles, reconstructed resonance masses), NLO corrections in **production** act largely as a flat *K*-factor



Checkpoint

Are there any questions?

POWHEG method implementations

- POWHEG method implementations:
  - POWHEG BOX V2: generic NLO+PS framework for many LHC processes
  - POWHEG BOX RES: resonance-aware extension (production $\otimes$ decay)
  - Herwig Matchbox: also implements POWHEG (and MC@NLO)
- [New!] Fortran codes, now hosted on gitlab.com
- A long list of available processes (examples)
  - Drell–Yan: W, Z, γ^* (with QCD and EW corrections); vector boson(s) + jets: Wj, Zj, $V+2j$, WW, WZ, ZZ, VBF $VVjj$; Higgs + jets: ggH, VBF, VH, $H+j$, Hjj, tops: t, $\bar{t}$, single-top (s-, t-, tW-channel)

POWHEG method implementations

- POWHEG method implementations:
  - POWHEG BOX V2: generic NLO+PS framework for many LHC processes
  - POWHEG BOX RES: resonance-aware extension (production $\otimes$ decay)
  - Herwig Matchbox: also implements POWHEG (and MC@NLO)
- [New!] Fortran codes, now hosted on gitlab.com
- A long list of available processes (examples)
  - Drell–Yan: W, Z, γ^* (with QCD and EW corrections); vector boson(s) + jets: $Wj, Zj, V+2j, WW, WZ, ZZ$, VBF $VVjj$; Higgs + jets: ggH, VBF, VH, $H+j$, Hjj, tops: $t, \bar{t}$, single-top (s-, t-, tW-channel)
- Philosophy
  - BOX core: interface to MEs, phase space, subtraction, POWHEG Sudakovs, and LHE interface, ...
  - Each process: in its own directory and is implemented largely independently on top of the core
- No full automation
  - downside: no “one-click” NLO+PS like MG5_aMC
  - upside: if a process is in the BOX, it usually has
    - a dedicated publication and thorough validation
    - with recommended settings and options, reference plots / benchmarks to reproduce

From POWHEG events to parton showers

- POWHEG output:
  - events in LHE: particles with momenta and colour, weight, `scalup`, …
- Pythia
  - reads LHE state and starts ISR/FSR shower from there
  - uses `scalup` as a veto scale: no emission with $p_T >$ `scalup` (also via `PowhegHooks`)
- Herwig
  - ordering variable is emission angle, so the hardest p_T emission is not necessarily the first
  - formally needs a **truncated shower**: soft wide-angle emissions above the POWHEG scale generated before the hardest emission, then the usual shower below it
  - practically small impact (in processes that were studied)
- Resonance-aware showers
  - POWHEG-RES requires multiple `scalups` not possible in LHE
  - Pythia / Herwig need dedicated interfaces
  - shower then typically preserves resonance virtualities

Summary

- Resonance-aware POWHEG:
  - standard $\Phi_R \leftrightarrow (\Phi_B, \Phi_{\text{rad}})$ mapping distorts internal resonances: on-shell in R, off-shell in $B \longrightarrow$ large R/B, Sudakov suppresses the peak
  - RES introduces resonance histories and resonance-preserving mappings: recoil is confined within each resonance system, resonance virtualities are identical in real and underlying Born kinematics
- Resonance histories and multiple radiation:
  - phase space is partitioned with resonance projectors $\Pi_{h(\Phi)}$: $R(\Phi) = \sum_h R^{h(\Phi)}$, each h a definite pattern of intermediate tops, W's, …
  - singular regions become (h, c) and the Sudakov per history uses a sum over splitting kernels; multi-rad scheme allows one POWHEG emission from production and from each resonance decay, adding useful higher-order structure in the decay chains while keeping NLO accuracy
- Showering POWHEG events:
  - Shower Monte Carlo's (Pythia, Herwig) read LHE files, and attach further emissions respecting POWHEG shower starting scale

Backup

Backup slides follow

Resonance-aware mapping for $t \to bgW^+$

Go to the top rest frame, recombine $p_b + k$ to define the direction of the Born b', and perform a standard 2-body decay $t \to b' + W^{+'}$ by rescaling $|(p_b)|$ and $|(p_W)|$ to the on-shell 2-body kinematics while keeping their back-to-back directions (with b' along $p_b + k$). Then boost the resulting $p_{b'}$ and $p_{W'}$ back to the lab frame with the inverse top boost $B_t^{\{-1\}}$.

Setup

- Real decay in lab frame: $t(p_t) \to b(p_b) + g(k) + W^+(p_W)$, with $p_t = p_b + k + p_W$.
- Underlying Born decay: $t(p_t) \to b'(p_{b'}) + W^{+'}(p_{W'})$, with $p_t = p_{b'} + p_{W'}$.

1. Go to the top rest frame

- Boost with B_t such that in this frame $p_t = (m_t, 0)$.
- Real momenta in this frame: $p_b \to B_t p_b, k \to B_t k, p_W \to B_t p_W$.

2. Recombine $b + g$ and fix direction

- Combined emitter momentum: $q = p_b + k$.
- Take unit vector along it: $\hat{q} = \frac{q}{\|q\|}$.

Resonance-aware mapping for $t \to bgW^+$

3. Two–body kinematics for $t \to b'W^{+'}$

- Treat the Born decay as a $1 \to 2$ decay $t \to b' + W^{+'}$.
- Common three–momentum magnitude: $p^* = \frac{\sqrt{\lambda(m_t^2, m_b^2, m_W^2)}}{2m_t}$ with $\lambda(a, b, c) = a^2 + b^2 + c^2 - 2ab - 2ac - 2bc$.
- Energies: $E_b^* = \sqrt{(p^*)^2 + m_b^2}$, $E_W^* = \sqrt{(p^*)^2 + m_W^2}$.
- Define Born momenta in the top rest frame: $p_{b'} = (E_b^*, +p^*\hat{q})$, $p_{W'} = (E_W^*, -p^*\hat{q})$.
- Then $p_{b'}^2 = m_b^2$, $p_{W'}^2 = m_W^2$ and $p_{b'} + p_{W'} = (m_t, 0) = p_t$.

4. Boost back to the lab frame

- Final underlying-Born momenta: $p_{b'} \to B_t^{\{-1\}} p_{b'}$, $p_{W'} \to B_t^{\{-1\}} p_{W'}$.

- In words: go to the top rest frame, recombine $b + g$ to set the b' direction, apply on-shell $1 \to 2$ kinematics $t \to b'W^{+'}$, then boost back to the lab frame

Universität Münster

Matching and merging
Part 2: Merging

Tomáš Ježo (tomas.jezo@uni-muenster.de)

institut für theoretische physik

28th November 2025

Outline, Part 2: Merging

Multi-jet merging

- Jet multiplicity and accuracy
- Concepts behind merging
- Shower branching history
- LO Merging formula

Merging via matching

- Suppressing singularities with Sudakovs
- MiNLO and MiNLO'

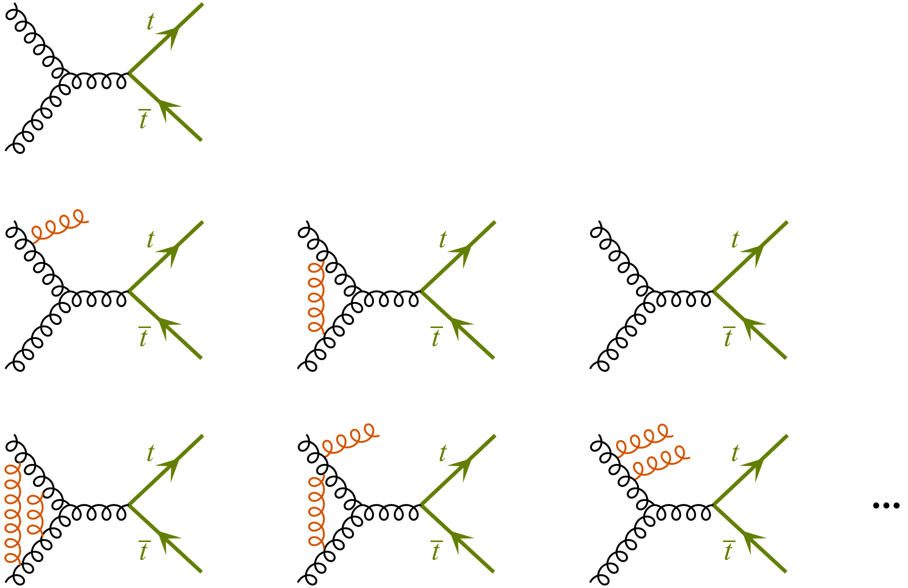
Jet multiplicity and accuracy

- Consider top-pair production at **FO**

LO:

NLO:

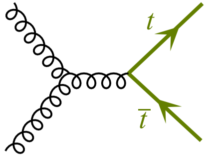
NNLO:



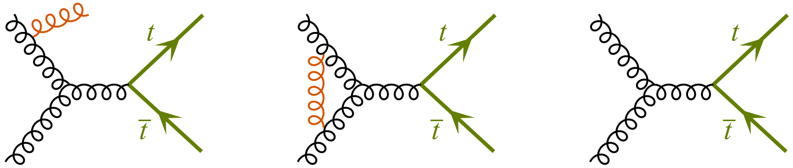
Jet multiplicity and accuracy

- Consider top-pair production at **FO**

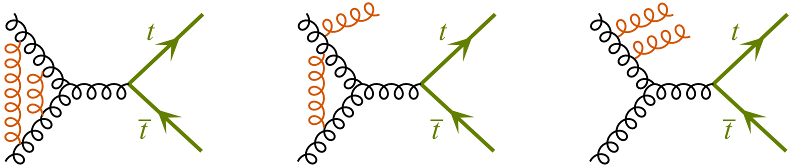
LO:



NLO:



NNLO:



...

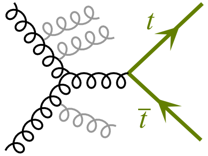
- What is the accuracy of the prediction for $t\bar{t} + n_j$ cross section?

calculation vs jet multiplicity	$n_j \geq 0$	$n_j \geq 1$	$n_j \geq 2$	$n_j \geq 3$
LO	LO	–	–	–
NLO	NLO	LO	–	–
NNLO	NNLO	NLO	LO	–

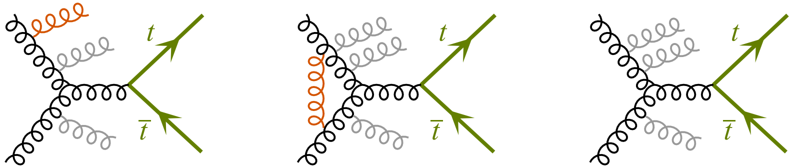
Jet multiplicity and accuracy

- Consider top-pair production at **FO** matched to **PS**

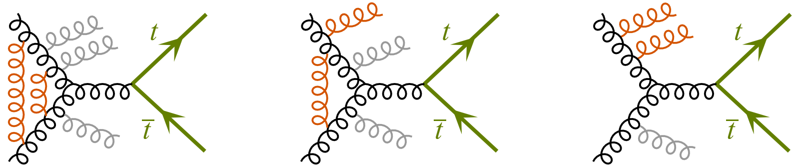
LO+PS:



NLO+PS:



NNLO+PS:



...

- What is the accuracy of the prediction for $t\bar{t} + n_j$ cross section?

calculation vs jet multiplicity	$n_j \geq 0$	$n_j \geq 1$	$n_j \geq 2$	$n_j \geq 3$
LO+PS	LO	PS	PS	PS
NLO+PS	NLO	LO	PS	PS
NNLO+PS	NNLO	NLO	LO	PS

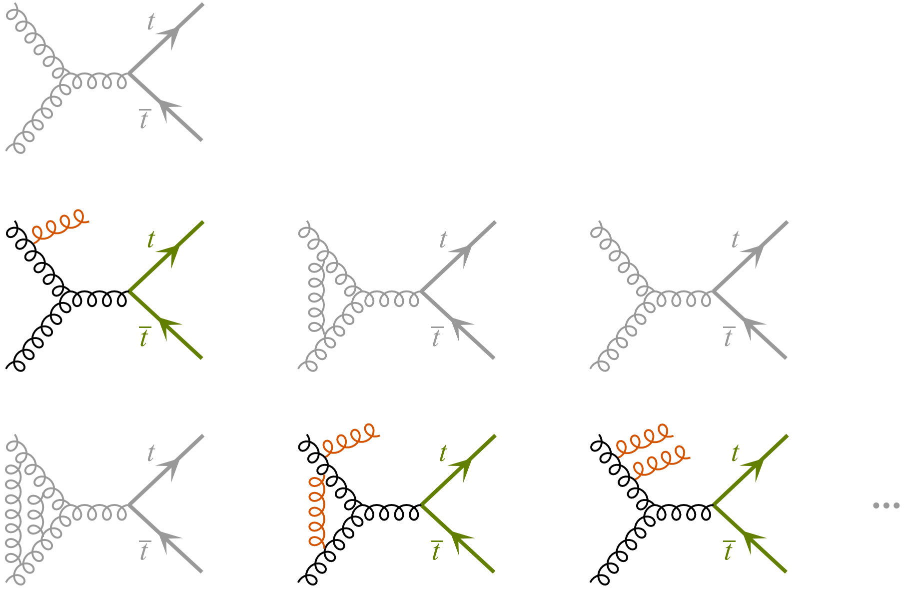
Jet multiplicity and accuracy

- Consider top-pair production in association with a jet at **FO**

LO:

NLO:

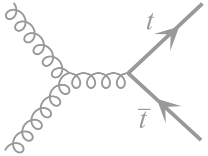
NNLO:



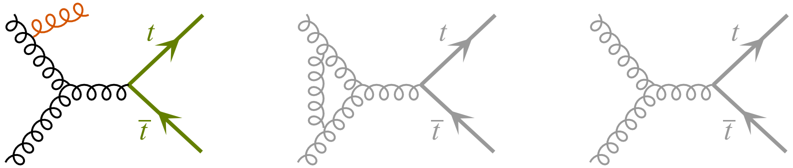
Jet multiplicity and accuracy

- Consider top-pair production in association with a jet at **FO**

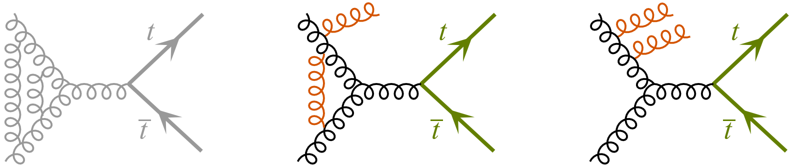
LO:



NLO:



NNLO:



...

- What is the accuracy of the prediction for $t\bar{t} + n_j$ cross section?

calculation vs jet multiplicity	$n_j \geq 0$	$n_j \geq 1$	$n_j \geq 2$	$n_j \geq 3$
LO	∞	LO	–	–
NLO	∞	NLO	LO	–

Checkpoint

Are there any questions?

Merging

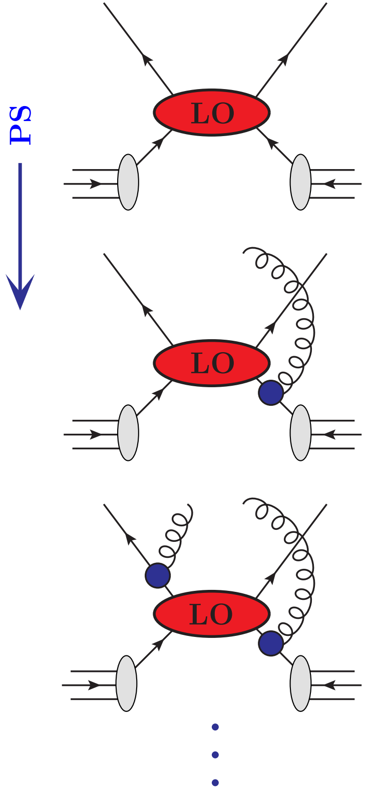
- **Matching FO** calculations with **PS**
  - N^nLO+PS describe observables dominated by topologies of a single multiplicity
  - But many observables receive contributions from many final state multiplicities, e.g. $p_{T,t\bar{t}}$
  - NLO+PS will describe this observable the low end at NLO accuracy, an intermediate region at LO accuracy, and the high end at PS accuracy only
  - If we want to describe such observables as uniformly as possible we need multi-jet merging

calculation vs jet multiplicity	$\geq$ 0 jets	$\geq$ 1 jet	$\geq$ 2 jets	$\geq$ 3 jets
LO+PS	LO	PS	PS	PS
(LO+j)+PS		LO	PS	PS
MEPS@LO (0-2j)	LO	LO	LO	PS

- **Merging** combines several **FO** samples with different jet multiplicities with **PS**
  - avoids double counting between matrix elements of different multiplicities and the **PS**

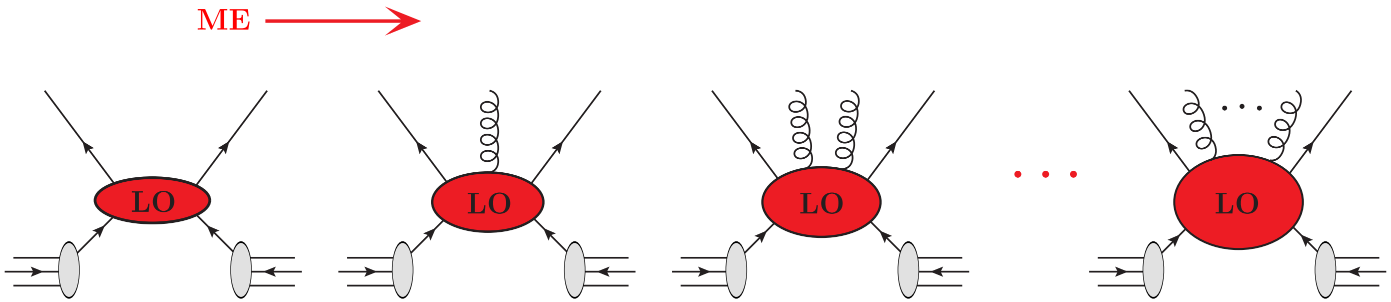
Merging

- **Merging** combines several **FO** samples with different jet multiplicities with **PS**



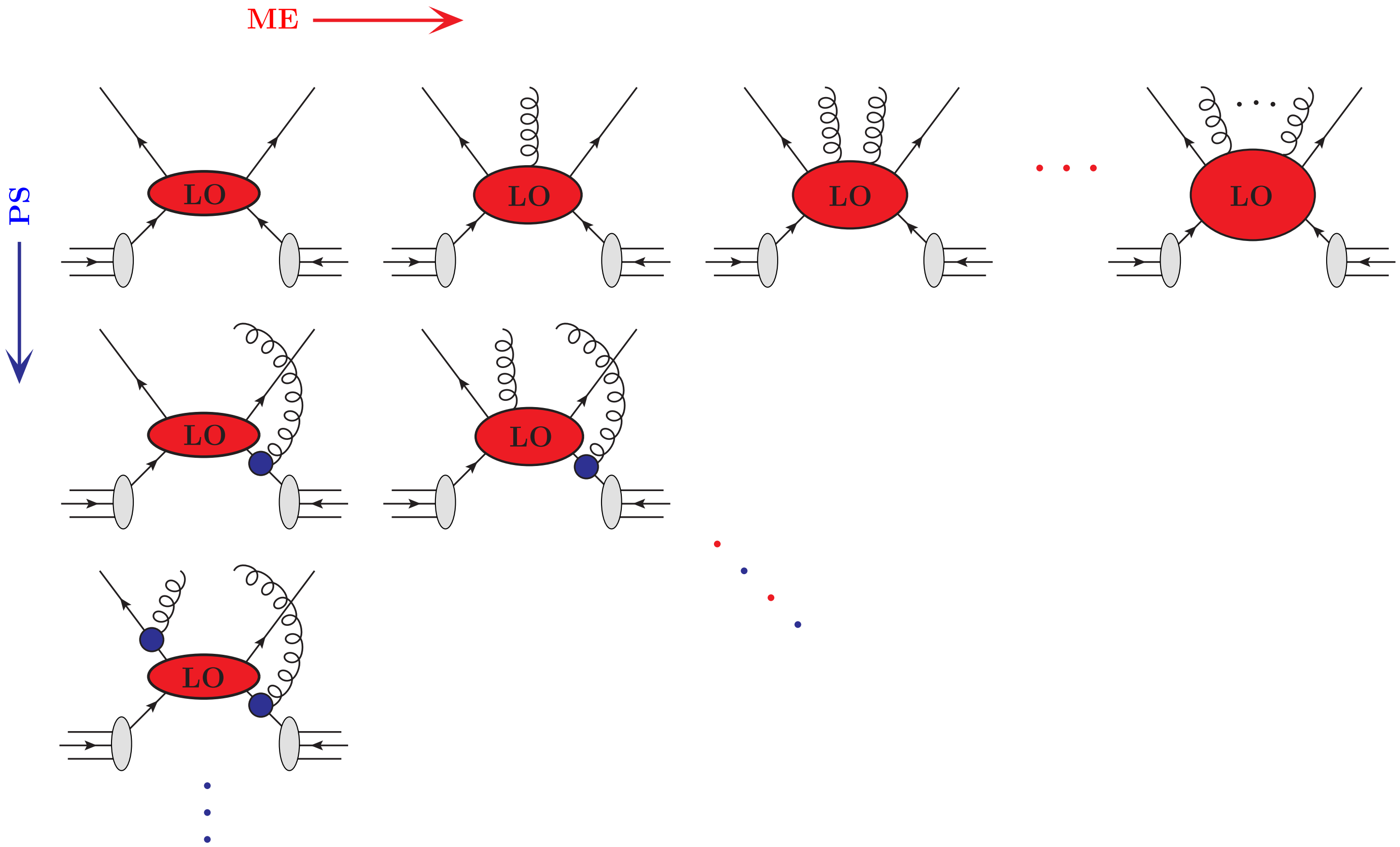
Merging

- **Merging** combines several **FO** samples with different jet multiplicities with **PS**



Merging

- **Merging** combines several **FO** samples with different jet multiplicities with **PS**



Shower branching history

- How a p_T-ordered shower builds an event
  - Start from a hard process at scale Q (e.g. $m_{t\bar{t}}$)
  - The shower emits partons with decreasing transverse momentum: $p_{T,1} > p_{T,2} > \cdots > p_{T,\min}$
  - $p_{T,1}$ = hardest emission, $p_{T,2}$ = 2nd-hardest, etc; $\{p_{T,i}\}$ encode **PS** branching history ordered in p_T

Shower branching history

- How a p_T-ordered shower builds an event
  - Start from a hard process at scale Q (e.g. $m_{t\bar{t}}$)
  - The shower emits partons with decreasing transverse momentum: $p_{T,1} > p_{T,2} > \cdots > p_{T,\text{min}}$
  - $p_{T,1}$ = hardest emission, $p_{T,2}$ = 2nd-hardest, etc; $\{p_{T,i}\}$ encode **PS** branching history ordered in p_T
- In order to replace some **PS** emissions by **FO**
  - Choose a merging scale $p_{T,\text{cut}}$:
    - jets with $p_T > p_{T,\text{cut}} \longrightarrow$ described by **FO**
    - jets with $p_T < p_{T,\text{cut}} \longrightarrow$ left to the shower
  - To know "which **FO** sample owns an event", we need the full sequence $\{p_{T,i}\}$, not just $p_{T,\text{cut}}$:
    - 1-jet **FO** region: $p_{T,1} > p_{T,\text{cut}}$ and $p_{T,2} < p_{T,\text{cut}}$
    - 2-jet **FO** region: $p_{T,2} > p_{T,\text{cut}}$ and $p_{T,3} < p_{T,\text{cut}}$
    - etc.
  - $\{p_{T,i}\}$ are obtained by clustering **FO** samples
  - They classify the event as a 0-jet, 1-jet, 2-jet, … ME configuration by counting how many emissions lie above $p_{T,\text{cut}}$

Checkpoint

Are there any questions?

Merging formula (schematic)

$$d\sigma^{\mathrm{MEPS}} = \int d\Phi_n B_n(\Phi_n)\mathrm{PS}_n \Theta(p_{T,\mathrm{cut}} - p_{T,n+1})$$

$$+ \int d\Phi_{n+1} B_{n+1}(\Phi_{n+1})\Theta(p_{T,n+1} - p_{T,\mathrm{cut}})\Delta_n(p_{T,n+1}, p_{T,n})\Theta(p_{T,\mathrm{cut}} - p_{T,n+2})\mathrm{PS}_{n+1}$$

$$+ \cdots$$

where $\mathrm{PS}_i = [\Delta_i(p_{T,\mathrm{min}}) + K_{\mathrm{PS},i}\Delta_i(p_T)d\Phi_{\mathrm{rad}}]$, $p_{T,n}$... the scale of the hard process

1. Generate $n + 0j$ sample, shower below $p_{T,\mathrm{cut}}$
2. Generate $n + 1j$ sample
   - for each event calculate branching history and keep only those with $p_{T,n+1} > p_{T,\mathrm{cut}} > p_{T,n+2}$
   - reweight with probability of no emissions harder than $p_{T,n+1}$, all the way to $p_{T,n}$
3. Generate $n + 2j$ sample
   - for each event calculate branching history and keep only those with $p_{T,n+2} > p_{T,\mathrm{cut}} > p_{T,n+1}$
   - reweight with probability of no emissions harder than $p_{T,n+2}$, up to $p_{T,n+1}$
4. etc.

Merging formula (schematic)

$$d\sigma^{\text{MEPS}} = \int d\Phi_n B_n(\Phi_n)\text{PS}_n\Theta(p_{T,\text{cut}} - p_{T,n+1})$$

$$+ \int d\Phi_{n+1} B_{n+1}(\Phi_{n+1})\Theta(p_{T,n+1} - p_{T,\text{cut}})\Delta_n(p_{T,n+1}, p_{T,n})\Theta(p_{T,\text{cut}} - p_{T,n+2})\text{PS}_{n+1}$$

$$+ \cdots$$

where $\text{PS}_i = [\Delta_i(p_{T,\text{min}}) + K_{\text{PS},i}\Delta_i(p_T)d\Phi_{\text{rad}}]$, $p_{T,n}$... the scale of the hard process

1. Generate $n + 0j$ sample, shower below $p_{T,\text{cut}}$
2. Generate $n + 1j$ sample
    - for each event calculate branching history and keep only those with $p_{T,n+1} > p_{T,\text{cut}} > p_{T,n+2}$
    - reweight with probability of no emissions harder than $p_{T,n+1}$, all the way to $p_{T,n}$
3. Generate $n + 2j$ sample
    - for each event calculate branching history and keep only those with $p_{T,n+2} > p_{T,\text{cut}} > p_{T,n+3}$
    - reweight with probability of no emissions harder than $p_{T,n+2}$, up to $p_{T,n+1}$
4. etc.

Merging formula (schematic)

$$d\sigma^{\text{MEPS}} = \int d\Phi_n B_n(\Phi_n)\text{PS}_n\Theta(p_{T,\text{cut}} - p_{T,n+1})$$
$$+ \int d\Phi_{n+1} B_{n+1}(\Phi_{n+1})\Theta(p_{T,n+1} - p_{T,\text{cut}})\Delta_n(p_{T,n+1}, p_{T,n})\Theta(p_{T,\text{cut}} - p_{T,n+2})\text{PS}_{n+1}$$
$$+ \cdots$$

where $\text{PS}_i = [\Delta_i(p_{T,\text{min}}) + K_{\text{PS},i}\Delta_i(p_T)d\Phi_{\text{rad}}]$, $p_{T,n}$... the scale of the hard process

1. Generate $n + 0j$ sample, shower below $p_{T,\text{cut}}$
2. Generate $n + 1j$ sample
   - for each event calculate branching history and keep only those with $p_{T,n+1} > p_{T,\text{cut}} > p_{T,n+2}$
   - reweight with probability of no emissions harder than $p_{T,n+1}$, all the way to $p_{T,n}$
3. Generate $n + 2j$ sample
   - for each event calculate branching history and keep only those with $p_{T,n+2} > p_{T,\text{cut}} > p_{T,n+3}$
   - reweight with probability of no emissions harder than $p_{T,n+2}$, up to $p_{T,n+1}$
4. etc.

Merging formula (schematic)

$$d\sigma^{\text{MEPS}} = \int d\Phi_n B_n(\Phi_n)\text{PS}_n \Theta(p_{T,\text{cut}} - p_{T,n+1})$$

$$+ \int d\Phi_{n+1} B_{n+1}(\Phi_{n+1})\Theta(p_{T,n+1} - p_{T,\text{cut}})\Delta_n(p_{T,n+1}, p_{T,n})\Theta(p_{T,\text{cut}} - p_{T,n+2})\text{PS}_{n+1}$$

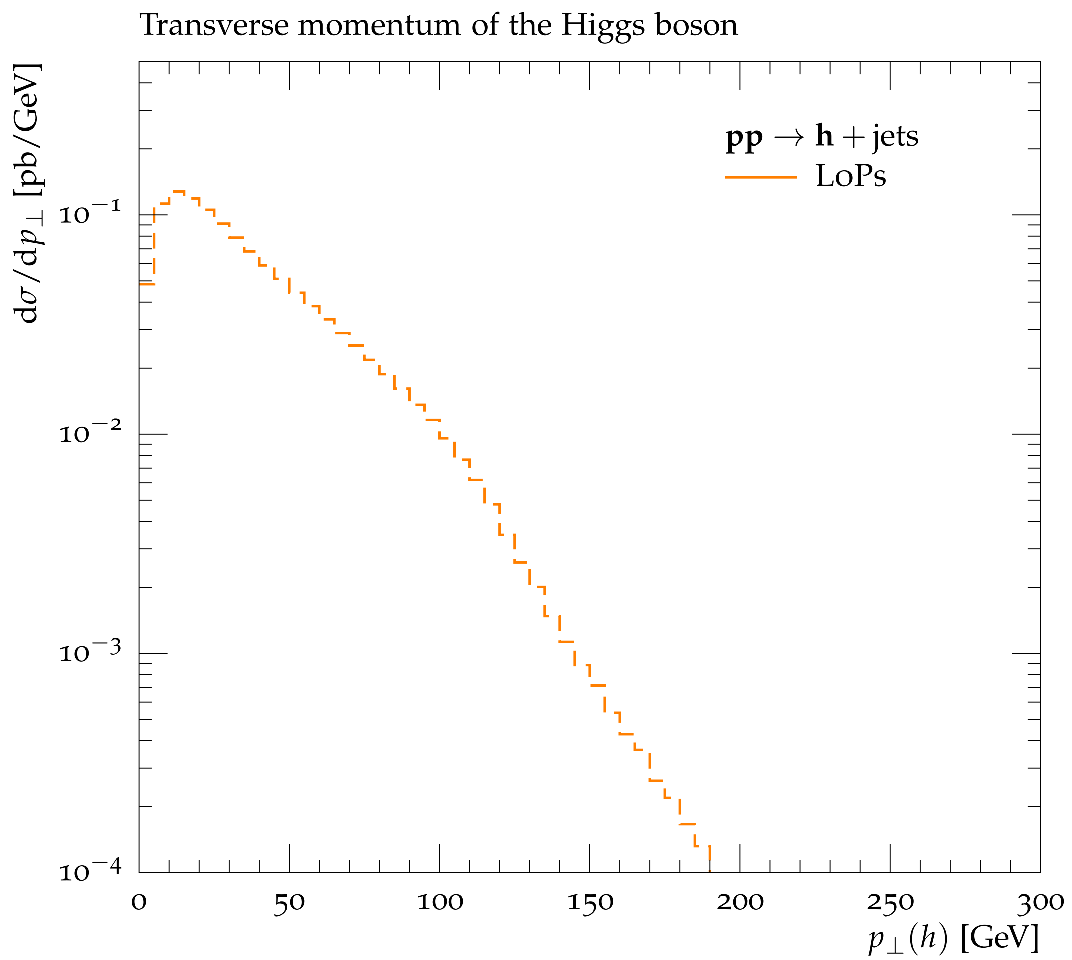
$$+ \cdots$$

where $\text{PS}_i = [\Delta_i(p_{T,\text{min}}) + K_{\text{PS},i}\Delta_i(p_T)d\Phi_{\text{rad}}]$, $p_{T,n}$... the scale of the hard process

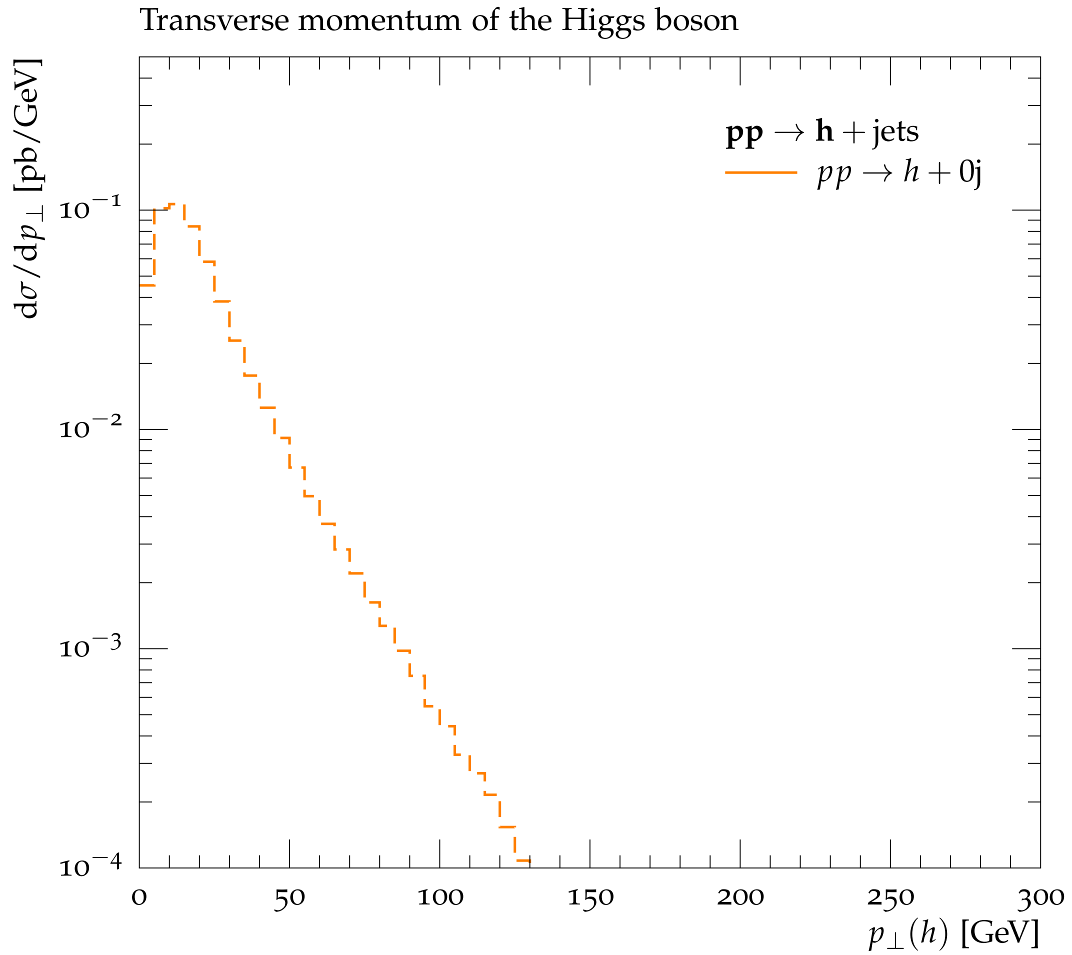
1. Generate $n + 0j$ sample, shower below $p_{T,\text{cut}}$
2. Generate $n + 1j$ sample
   - for each event calculate branching history and keep only those with $p_{T,n+1} > p_{T,\text{cut}} > p_{T,n+2}$
   - reweight with probability of no emissions harder than $p_{T,n+1}$, all the way to $p_{T,n}$
3. Generate $n + 2j$ sample
   - for each event calculate branching history and keep only those with $p_{T,n+2} > p_{T,\text{cut}} > p_{T,n+3}$
   - reweight with probability of no emissions harder than $p_{T,n+2}$, up to $p_{T,n+1}$
4. etc.

- singularities in **FO** are avoided by never integrating below $p_{T,\text{cut}}$

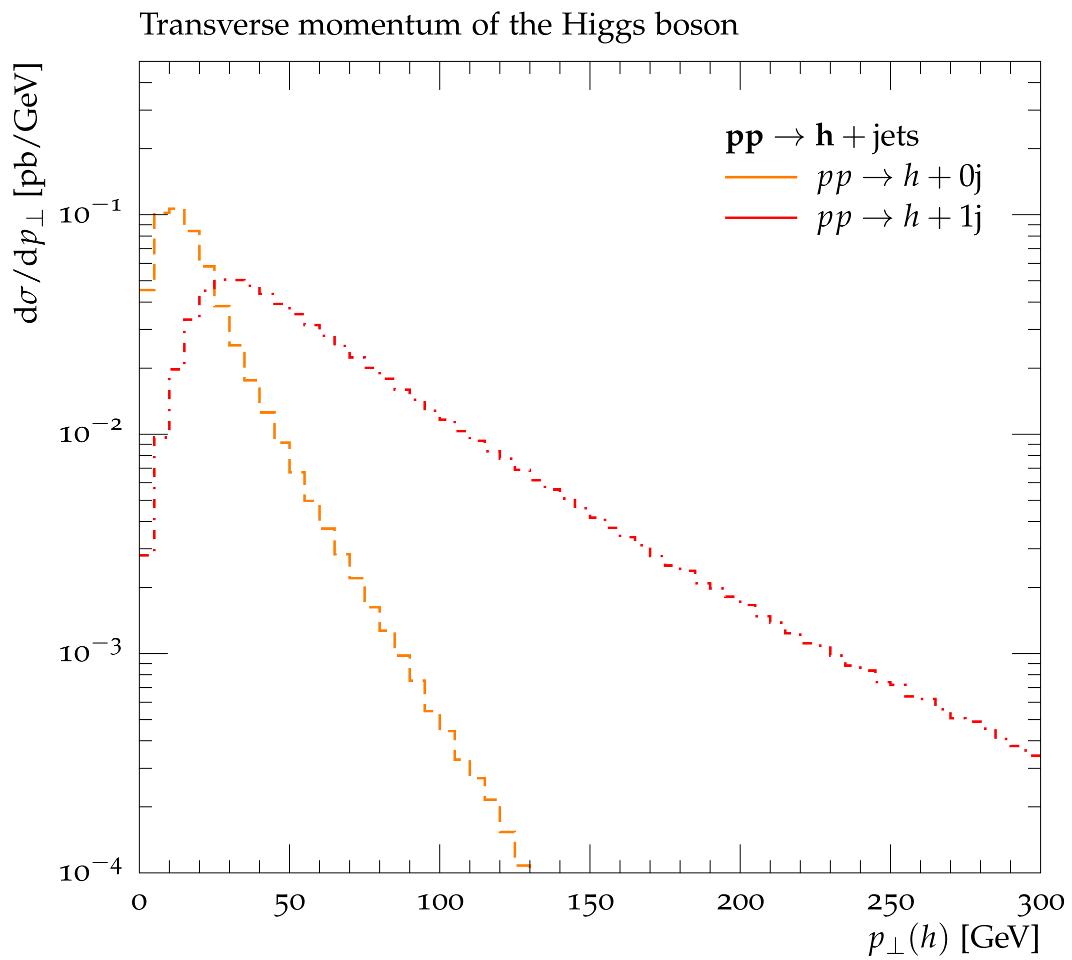
Higgs p_T at MEPS@LO (0-3j)



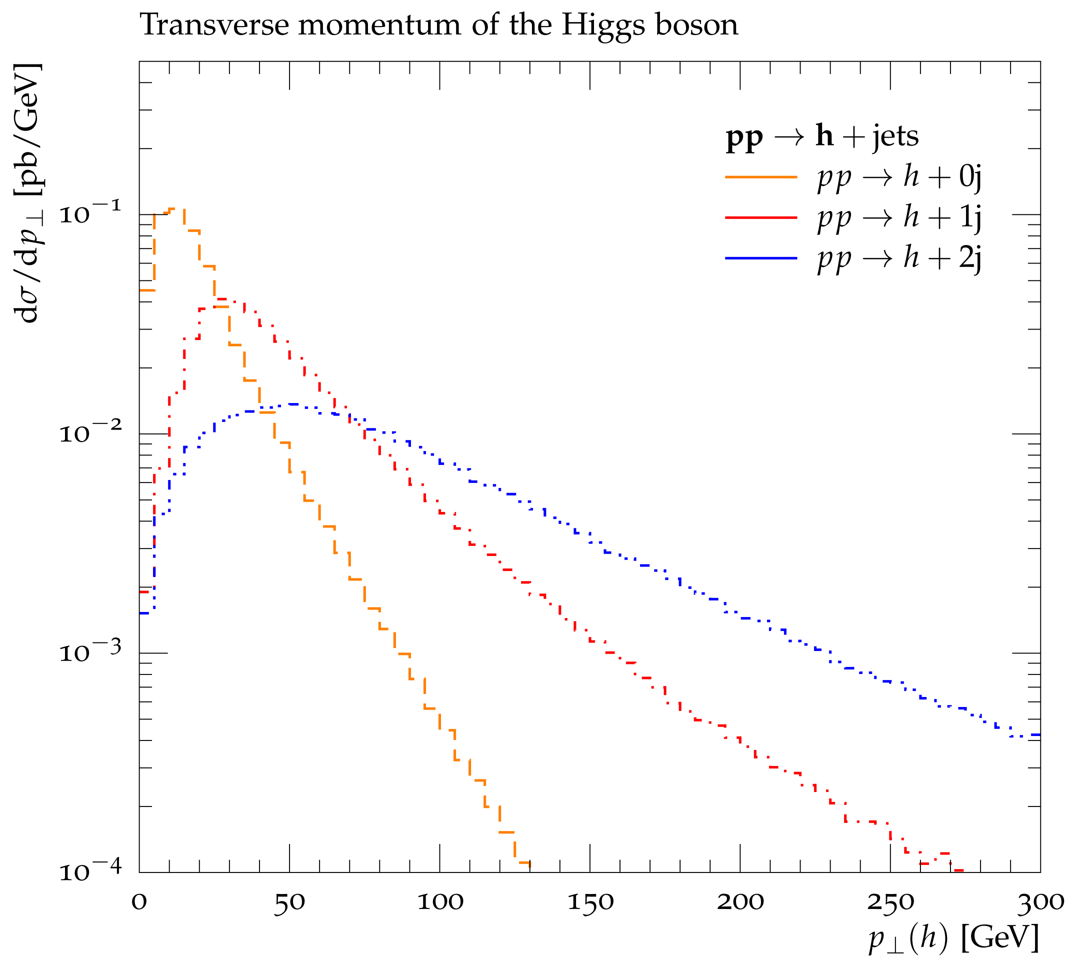
Higgs p_T at MEPS@LO (0-3j)



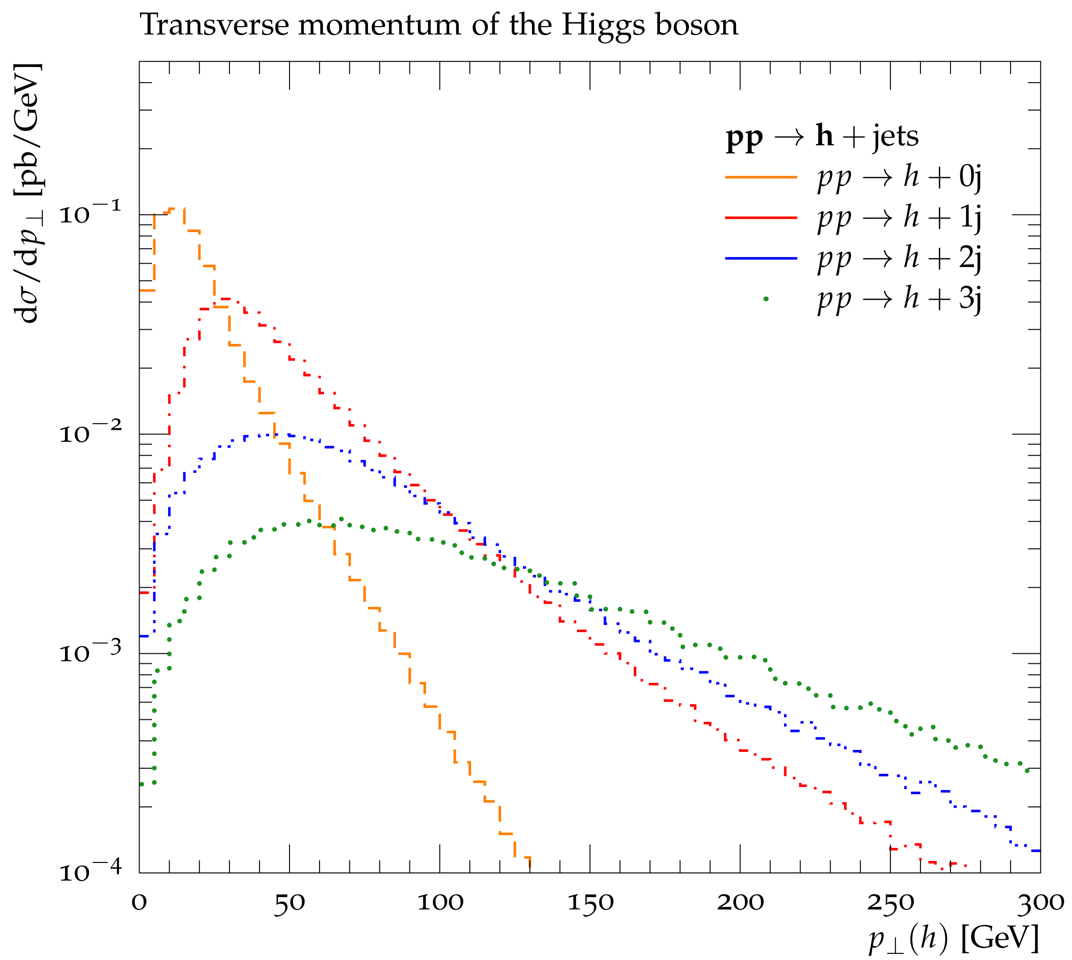
Higgs p_T at MEPS@LO (0-3j)



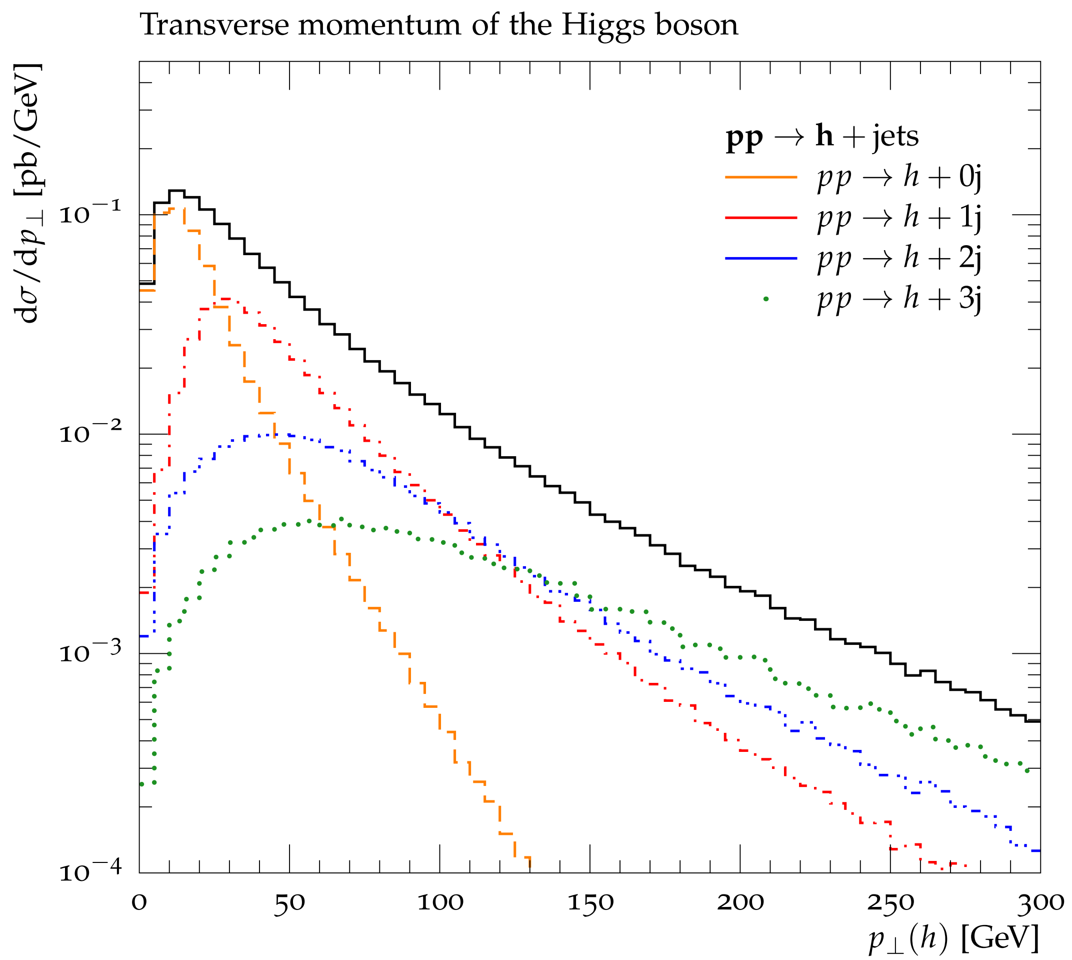
Higgs p_T at MEPS@LO (0-3j)



Higgs p_T at MEPS@LO (0-3j)



Higgs p_T at MEPS@LO (0-3j)



Available multi-jet merging methods (LO & NLO)

LO merging

- CKKW [Catani et al., JHEP 0111 (2001) 063]
- CKKW-L [Lönnblad, JHEP 0205 (2002) 046]
- MLM [Mangano et al., JHEP 0307 (2003) 001]
- UMEPS [Lönnblad & Prestel, JHEP 1203 (2012) 019]

NLO merging

- MEPS@NLO [Höche et al., JHEP 1304 (2013) 027]
- FxFx [Frederix & Frixione, JHEP 1212 (2012) 061]
- UNLOPS [Lönnblad & Prestel, JHEP 1303 (2013) 166]
- MENLOPS [Hamilton, Nason, JHEP 1006 (2010) 039]

Summary

- Jet multiplicity vs fixed order
  - LO: only core process $\longrightarrow$ "≥0 jets" LO, no prediction for higher jet bins
  - NLO: first real emission $\longrightarrow$ "≥1 jet" LO, "≥0 jets" NLO
  - etc.
- Why merging
  - LO+PS gives many jets but only LL accuracy for hard ones
  - Separate 0, 1, 2, …-jet **FO** samples overlap; need a clean way to assign phase space and avoid double counting
- Shower branching history and $p_{T,i}$
  - A p_T-ordered shower generates a ladder $p_{T,1} > p_{T,2} > \ldots$
  - In **FO** sample obtained by clustering and used to (a) count how many jets are above $p_{T,\mathrm{cut}}$ and (b) attach Sudakovs $\Delta(p_{T,i}, p_{T,i-1})$
- Schematic multi-jet merging picture
  - For each multiplicity k: use B_k only when exactly k emissions have $p_T > p_{T,\mathrm{cut}}$, multiply by Sudakovs built from the underlying lower-multiplicity process, and shower below $p_{T,\mathrm{cut}}$

Checkpoint

Are there any questions?

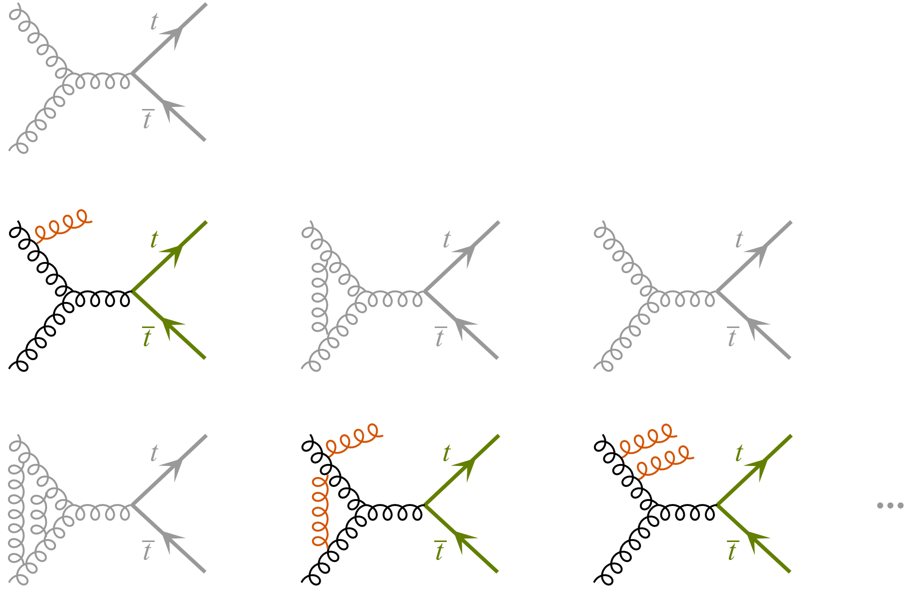
Jet multiplicity and accuracy, repeated

- Consider top-pair production in association with a jet at **FO**

LO:

NLO:

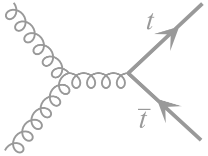
NNLO:



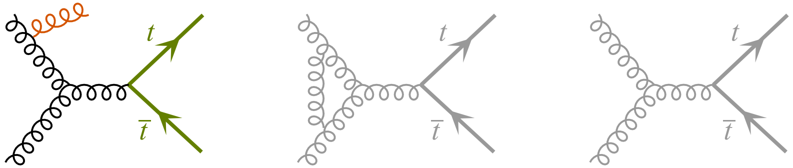
Jet multiplicity and accuracy, repeated

- Consider top-pair production in association with a jet at **FO**

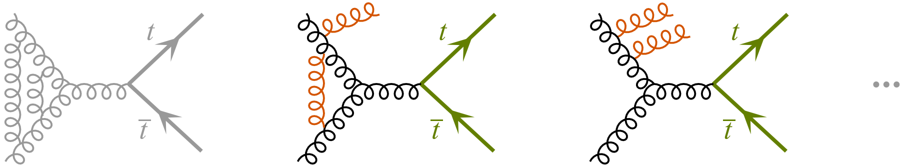
LO:



NLO:



NNLO:



- What is the accuracy of the prediction for $t\bar{t} + n_j$ cross section?

calculation vs jet multiplicity	$n_j \geq 0$	$n_j \geq 1$	$n_j \geq 2$	$n_j \geq 3$
LO	∞	LO	–	–
NLO	∞	NLO	LO	–

Singularity? Sudakov please!

- What is the accuracy of the prediction for $t\bar{t} + n_j$ cross section?

calculation vs jet multiplicity	$n_j \geq 0$	$n_j \geq 1$	$n_j \geq 2$	$n_j \geq 3$
LO	∞	LO	–	–
NLO	∞	NLO	LO	–

- when the extra parton becomes soft/collinear, the matrix element behaves like $B_{t\bar{t}+1j}(p_T) \propto 1/p_T^2$
- the inclusive contribution from this region is $\sigma_{t\bar{t}+1j} \sim \int_0^{Q^2} dp_T^2/p_T^2$, which diverges as $p_T \to 0$
- Cure: multiply by a Sudakov form factor
  - introduce $\Delta(Q, p_T) \approx \exp\left(-a\alpha_s \ln^2(Q/p_T)\right)$: probability of no emissions between the hard scale Q and an emission with scale p_T
  - then $\sigma \sim \int_0^{Q^2} dp_T^2/p_T^2 \Delta(Q, p_T)$ becomes finite, since $\Delta(Q, p_T) \to 0$ rapidly as $p_T \to 0$

Merging via matching

- MiNLO (Multi-scale improved NLO):
  - Start from an implementation of "process +1j" (e.g. $t\bar{t} + 1j$) at NLO in POWHEG
  - Evaluate each power of α_s at the corresponding scale from the branching history
  - Multiply by $\Delta(Q, p_T)$ built from the known small-p_T resummation of the underlying process
  - Result: 1-jet region still NLO accurate, 0-jet region finite with the correct "log structure"

Merging via matching

- MiNLO (Multi-scale improved NLO):
  - Start from an implementation of "process +1j" (e.g. $t\bar{t} + 1j$) at NLO in POWHEG
  - Evaluate each power of α_s at the corresponding scale from the branching history
  - Multiply by $\Delta(Q, p_T)$ built from the known small-p_T resummation of the underlying process
  - Result: 1-jet region still NLO accurate, 0-jet region finite with the correct "log structure"
- MiNLO′ refinement:
  - Standard analytic resummation of the underlying process yields a process-dependent constant (often called B_2) in the exponential
  - MiNLO construction supplemented with this B_2 term leads to, after integrating over the jet, to NLO accuracy for the lower multiplicity
  - Net effect: a single MiNLO′ generator is NLO in the 1-jet region and NLO for inclusive H / $t\bar{t}$, without an explicit 0-jet sample or a separate merging step

calculation vs jet multiplicity	$n_j \geq 0$	$n_j \geq 1$	$n_j \geq 2$	$n_j \geq 3$
MiNLO'	NLO	NLO	LO	–

Merging via matching

- MiNLO (Multi-scale improved NLO):
  - Start from an implementation of "process +1j" (e.g. $t\bar{t} + 1j$) at NLO in POWHEG
  - Evaluate each power of α_s at the corresponding scale from the branching history
  - Multiply by $\Delta(Q, p_T)$ built from the known small-p_T resummation of the underlying process
  - Result: 1-jet region still NLO accurate, 0-jet region finite with the correct "log structure"
- MiNLO′ refinement:
  - Standard analytic resummation of the underlying process yields a process-dependent constant (often called B_2) in the exponential
  - MiNLO construction supplemented with this B_2 term leads to, after integrating over the jet, to NLO accuracy for the lower multiplicity
  - Net effect: a single MiNLO′ generator is NLO in the 1-jet region and NLO for inclusive H / $t\bar{t}$, without an explicit 0-jet sample or a separate merging step

calculation vs jet multiplicity	$n_j \geq 0$	$n_j \geq 1$	$n_j \geq 2$	$n_j \geq 3$
MiNLO'	NLO	NLO	LO	–

- MiNNLOPS: NNLO+PS matching building on to of the MiNLO' idea

Summary

- For "process $+1j$", as $p_T \to 0$ the ME behaves like $B \propto \frac{1}{p_T^2} \longrightarrow$ non–integrable divergence
- Multiplying by a Sudakov $\Delta(Q, p_T)$ (no-emission probability) suppresses the small-p_T region: the product $B\Delta$ is integrable and has the correct small-p_T logarithmic structure
  - here the Sudakov argument p_T is the hardest scale in the branching history (from clustering the event)
- This idea supplemented with a process-dependent constant (the B_2 term from p_T resummation) yields a sample in which the inclusive lower-multiplicity is at NLO
  - so there are two multiplicities at the same accuracies with explicit merging
- Foundation for NNLO+PS matching

Backup: MiNLO and the correct log structure

- Fixed order at small p_T
  - For colour–singlet Q,
$$d\frac{\sigma}{dp_T^2} \sim \sum_n \alpha_s^n \sum_{k=0}^{2n-1} c_{n,k} L^k, \quad L = \ln\left(Q^2/p_T^2\right)$$
  - As $p_T \to 0$ ($L \to \infty$), each order has large terms like $\alpha_s L^2, \alpha_s^2 L^4, \ldots \longrightarrow$ FO breaks down / diverges
- Resummation: reorganise the series into a Sudakov exponent
$$d\frac{\sigma}{dp_T^2} \sim H(Q)\exp\left[Lg_1(\alpha_s L) + g_2(\alpha_s L) + \alpha_s g_3(\alpha_s L) + \cdots\right]$$
  - g_1 = LL terms, g_2 = NLL, etc.
  - The Sudakov $\Delta(Q, p_T)$ suppresses the cross section as $p_T \to 0$ and makes the integral finite
- "Correct log structure"
  - If we expand $\Delta(Q, p_T)$ used in MiNLO/MiNLO′, the coefficients of $\alpha_s^n L^{\{2n\}}, \alpha_s^n L^{\{2n-1\}}, \ldots$ match the known LL/NLL (...) coefficients for the underlying process
  - So "Correct log structure" in the 0-jet region means:
    - finite as $p_T \to 0$, and
    - the same Sudakov suppression pattern as in the standard p_T resummation