Universität Münster

Matching and merging
Part 1: Matching

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

institut für theoretische physik

27th November 2025

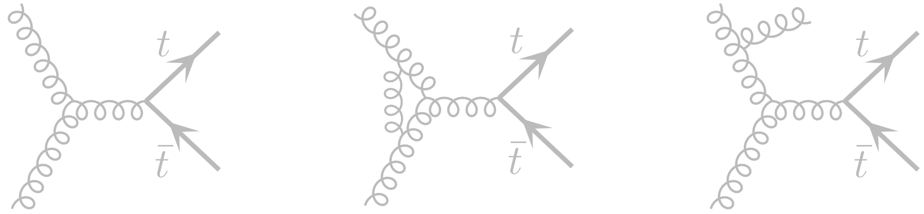
Preliminaries

What I expect you already know

- Fixed-order (FO) calculations:
  - Precise predictions for a scattering involving up to a fixed number of final-state particles

$$O \approx \alpha_s^m \alpha^n O^{m,n} + \alpha_s^{m+1} \alpha^n O^{m+1,n} + \alpha_s^m \alpha^{n+1} O^{m,n+1} + \cdots$$

  - Example: $t\bar{t}$ production at NLO QCD



  - Rene's lecture: Born, virtual and real corrections, treatment of IR singularities, ...

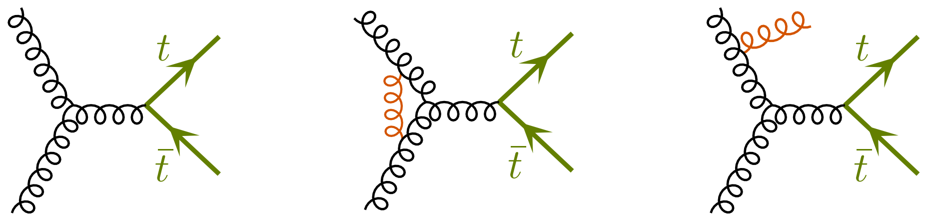
Preliminaries

What I expect you already know

- Fixed-order (FO) calculations:
  - Precise predictions for a scattering involving up to a fixed number of final-state particles

$$O \approx \alpha_s^2 \alpha^0 O^{2,0} + \alpha_s^3 \alpha^n O^{3,0} + \alpha_s^2 \alpha^1 O^{m,n+1} + \cdots$$

  - Example: $t\bar{t}$ production at NLO QCD



  - Rene's lecture: Born, virtual and real corrections, treatment of IR singularities, …

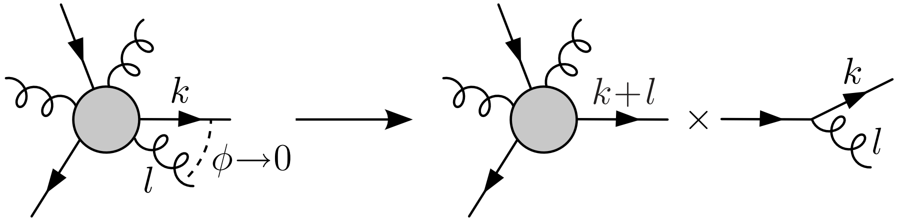
Preliminaries

What I expect you already know

- Fixed-order (FO) calculations:
  - Precise predictions for a scattering involving up to a fixed number of final-state particles

$$O \approx \alpha_s^m \alpha^n O^{m,n} + \alpha_s^{m+1} \alpha^n O^{m+1,n} + \alpha_s^m \alpha^{n+1} O^{m,n+1} + \cdots$$

  - Rene's lecture: Born, virtual and real corrections, treatment of IR singularities, ...
- Parton showers (PS):
  - Iterated parton branchings based on $n + 1 \approx n \times 1$ factorisation in soft/collinear limits



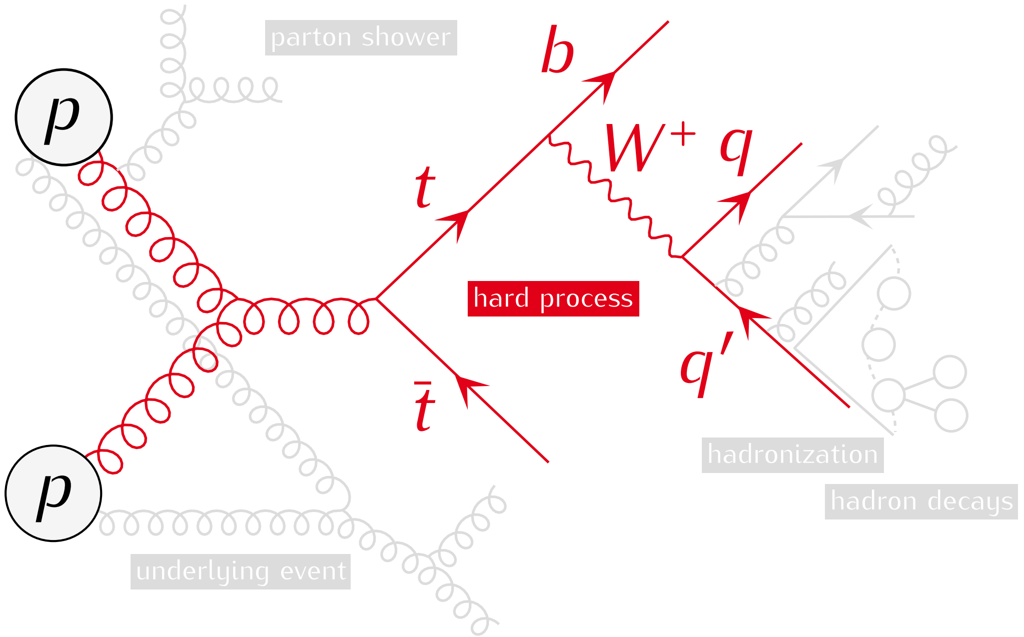
  - Torbjörn's lecture: splitting kernels, Sudakov form factor, evolution variable, ...
  - Andera's lecture: logarithmic expansion in α_s and L

What is matching?

FO versus PS

	FO	PS
Inclusive rate (normalisation)	✓	✗
Well-defined $(N)^i$LO accuracy	✓	✗
Missing HO uncertainties	✓	✗
Exact wide angle emissions	✓	✗
Small-p_T/jet-veto region (soft/ collinear log resummation)	✗	✓
Realistic event structure (many jets)	✗	✓
Hadronisation	✗	✓
Underlying event / MPI	✗	✓

Matching: a recipe to combine FO and PS

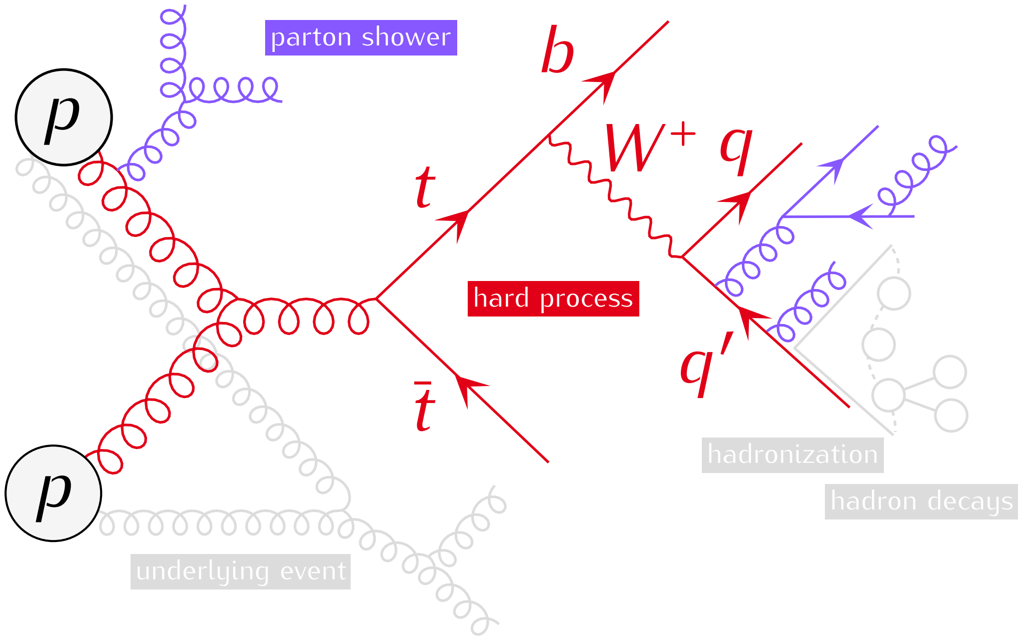


What is matching?

FO versus PS

	FO	PS
Inclusive rate (normalisation)	✓	✗
Well-defined $(N)^i$LO accuracy	✓	✗
Missing HO uncertainties	✓	✗
Exact wide angle emissions	✓	✗
Small-p_T/jet-veto region (soft/ collinear log resummation)	✗	✓
Realistic event structure (many jets)	✗	✓
Hadronisation	✗	✓
Underlying event / MPI	✗	✓

Matching: a recipe to combine FO and PS

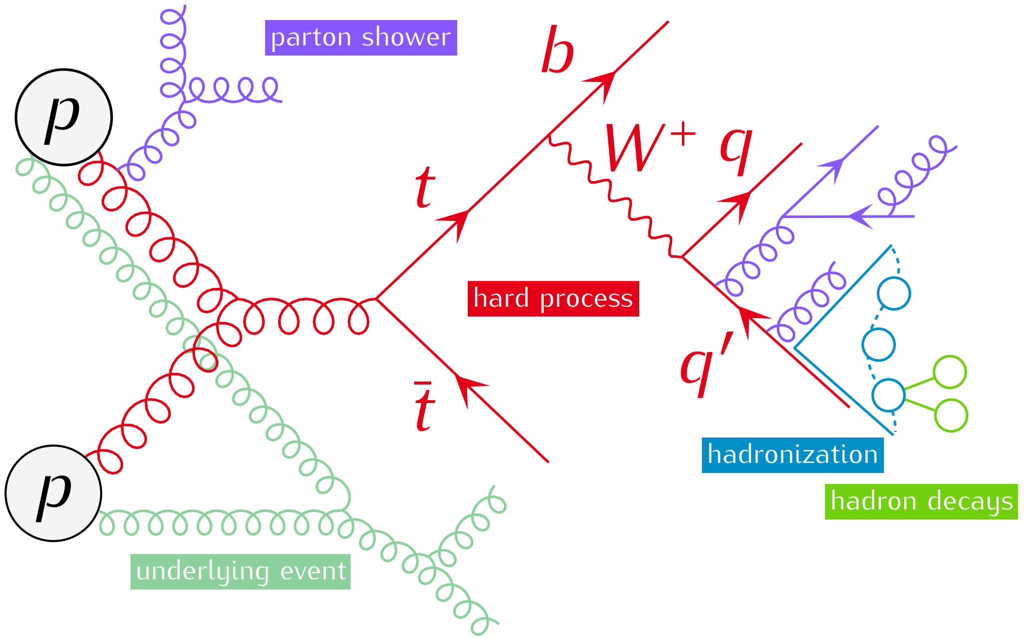


What is matching?

FO versus PS

	FO	PS
Inclusive rate (normalisation)	✓	✗
Well-defined $(N)^i$LO accuracy	✓	✗
Missing HO uncertainties	✓	✗
Exact wide angle emissions	✓	✗
Small-p_T/jet-veto region (soft/ collinear log resummation)	✗	✓
Realistic event structure (many jets)	✗	✓
Hadronisation	✗	✓
Underlying event / MPI	✗	✓

Matching: a recipe to combine FO and PS

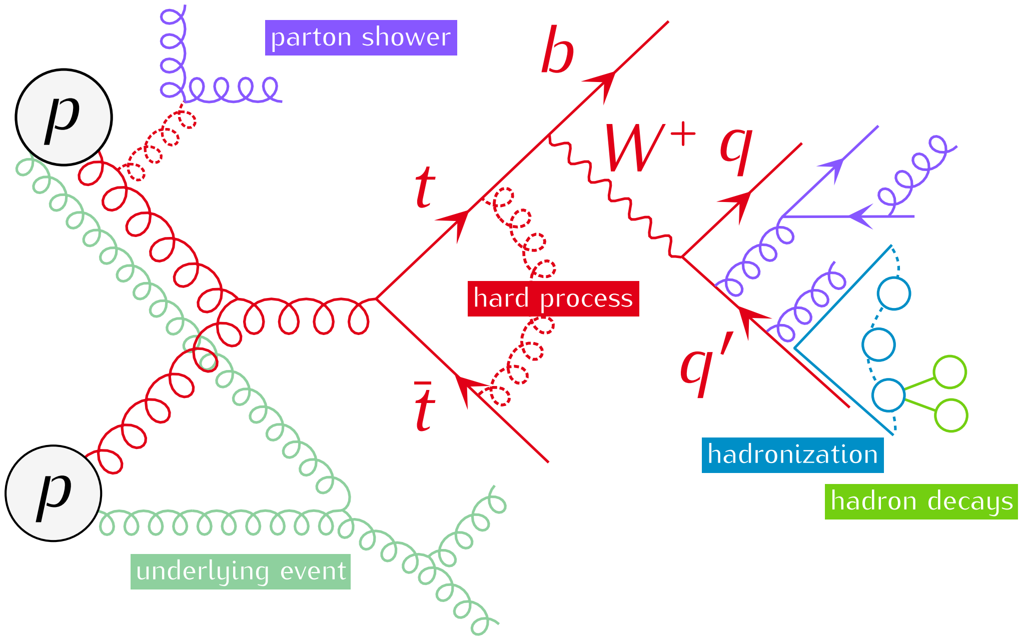


What is matching?

FO versus PS

	FO	PS
Inclusive rate (normalisation)	✓	✗
Well-defined $(N)^i$LO accuracy	✓	✗
Missing HO uncertainties	✓	✗
Exact wide angle emissions	✓	✗
Small-p_T/jet-veto region (soft/ collinear log resummation)	✗	✓
Realistic event structure (many jets)	✗	✓
Hadronisation	✗	✓
Underlying event / MPI	✗	✓

Matching: a recipe to combine FO and PS



What is matching?

What is matching

- **Matching** combines hard process, described by **FO** matrix element, with **PS**
  - LO+PS straightforward: shower off the Born legs; the unitary shower keeps the total rate at LO
  - NLO+PS tricky but mostly worked out†: the **FO** real correction contains one extra parton, already considered in the **PS** $\longrightarrow$ overlap/potential double counting; need explicit **matching schemes**
  - going beyond NLO is work in progress

What matching is NOT

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

Matching vs. Merging

- **Matching**: one FO process + PS.
- **Merging**: many FO processes (different multiplicities) + PS.

† Up leading-logarithmic accuracy.

Outline, Part 1: Matching

NLO+PS matching

- Matching types
- Available schemes

NLO+PS matching with POWHEG

- POWHEG formula
- Singular regions
- Tuning the real cross section

Resonance-aware POWHEG

- Resonance histories
- Multiple-radiation scheme

POWHEG BOX V2/RES

- Note on negative weights
- Les Houches Events and shower interface

Types of matching schemes: additive vs multiplicative

Additive matching (MC@NLO-type)

$$\sigma_{\text{add}} = \sigma_{\text{LO+PS}} + \left(\sigma_{\text{NLO}} - \sigma^{\text{NLO}}_{\text{LO+PS}}\right)$$

- Starts from LO+PS prediction and **add a correction**
- Reproduces the fixed-order high-p_T tail by construction
- Shower starting scale is a matching choice
- The correction can be negative and larger than the first term: negative weighted events

Multiplicative matching (POWHEG-type)

$$\sigma_{\text{mult}} = \sigma_{\text{LO+PS}} \cdot \left(\frac{\sigma_{\text{NLO}}}{\sigma^{\text{NLO}}_{\text{LO+PS}}}\right)$$

- Starts from LO+PS prediction and **rescale it** with a local NLO K-factor
- The hardest emission is generated with the exact real matrix element
- Hardest-emission p_T defines a natural event-by-event shower starting scale
- Local K-factor $>0 \Longrightarrow$ positive weighted events

Where: $\sigma_{\text{LO+PS}}$: LO **FO** + **PS**; σ_{NLO}: NLO **FO** (no shower); $\sigma^{\text{NLO}}_{\text{LO+PS}}$: NLO expansion of $\sigma_{\text{LO+PS}}$ (LO+PS contribution that is already NLO)

Available NLO+PS matching schemes

- **MC@NLO** (additive; shower-dependent): additive matching around the PS; very general and widely used, but intrinsically produces negative-weight events.
  [Frixione & Webber, JHEP 0206 (2002) 029]
- **POWHEG** (multiplicative; largely shower-agnostic): hardest emission from an NLO Sudakov, PS adds only softer emissions under a veto; positive weights as long as $\overline{B}(\Phi_B) > 0$.
  [Nason, JHEP 0411 (2004) 040]
- **KrkNLO** (multiplicative; shower-specific): NLO reweighting in a dedicated MC factorisation scheme; positive weights but tightly tied to the chosen shower.
  [Jadach et al., JHEP 1510 (2015) 052]
- **MAcNLOPS** (hybrid multiplicative–accumulative): combines MC@NLO- and POWHEG-like ideas for better control of logs and reduced negative weights. [Nason & Salam, JHEP 2201 (2022) 067]
- **ESME** (hybrid; NLL+NLO, positive-weight): PanScales matching scheme with NLL-accurate showers and exponentiated subtractions. [van Beekveld et al., to appear in JHEP (2025)]

Checkpoint

Are there any questions?

POWHEG Method Introduction

Why POWHEG?

- POWHEG = POsitive Weight Hardest Emission Generator
- Goal: combine **FO** NLO QCD with **PS**, preserving fully differential NLO accuracy for inclusive observables (inclusive w.r.t. radiation)
- Generate (mostly) positive-weight events output in shower-independent Les Houches events (same samples can be showered in Pythia and Herwig)

Core Idea

- Compute the **FO** NLO differential cross section for the underlying Born configuration ($\overline{B}(\Phi_B)$)
- Generate the hardest radiation (POWHEG emission) according to a unitary probability distribution built from the exact real matrix element (the square bracket)

POWHEG formula: $$d\sigma = \overline{B}(\Phi_B)\left[\Delta(p_{T,\,\mathrm{min}}) + d\Phi_{\mathrm{rad}}\Delta\big(p_{T(\Phi_{\mathrm{rad}})}\big)\frac{R(\Phi_B,\Phi_{\mathrm{rad}})}{B(\Phi_B)}\right]$$

- Then let the **PS** add softer, ordered emissions
  - with a veto to avoid harder emissions than the POWHEG one

From LO+PS to POWHEG

Starting point: LO+PS master formula

$$d\sigma_{\text{LO+PS}} = B(\Phi_B)\left[\Delta_{\text{PS}}(t_0) + d\Phi_{\text{rad}}\Delta_{\text{PS}}(t)K_{\text{PS}}(\Phi_B, \Phi_{\text{rad}})\right]$$

- t is the **PS** evolution variable, $K_{\text{PS}}(\Phi_B, \Phi_{\text{rad}})$ the **PS** emission kernel, $B(\Phi_B)$ the FO LO weight
- The square bracket is **unitary**: $\left[\Delta_{\text{PS}}(t_0) + \int d\Phi_{\text{rad}}\Delta_{\text{PS}}(t)K_{\text{PS}}(\Phi_B, \Phi_{\text{rad}})\right] = 1$

Upgrades that give POWHEG

- $t = p_{T(\Phi_{\text{rad}})}$
- $B(\Phi_B) \to \overline{B}(\Phi_B)$, where $\overline{B}(\Phi_B)$ is the **FO** NLO weight
- $K_{\text{PS}}(\Phi_B, \Phi_{\text{rad}}) \to R(\Phi_B, \Phi_{\text{rad}})/B(\Phi_B)$, same in soft/collinear limits
- Build the Sudakov with R / B (which guarantees that the square bracket is still **unitary**)

From LO+PS to POWHEG

Starting point: LO+PS master formula

$$d\sigma_{\text{LO+PS}} = B(\Phi_B)[\Delta_{\text{PS}}(t_0) + d\Phi_{\text{rad}}\Delta_{\text{PS}}(t)K_{\text{PS}}(\Phi_B, \Phi_{\text{rad}})]$$

- t is the **PS** evolution variable, $K_{\text{PS}}(\Phi_B, \Phi_{\text{rad}})$ the **PS** emission kernel, $B(\Phi_B)$ the FO LO weight
- The square bracket is **unitary**: $[\Delta_{\text{PS}}(t_0) + \int d\Phi_{\text{rad}}\Delta_{\text{PS}}(t)K_{\text{PS}}(\Phi_B, \Phi_{\text{rad}})] = 1$

Upgrades that give POWHEG

- $t = p_{T(\Phi_{\text{rad}})}$
- $B(\Phi_B) \to \overline{B}(\Phi_B)$, where $\overline{B}(\Phi_B)$ is the **FO** NLO weight
- $K_{\text{PS}}(\Phi_B, \Phi_{\text{rad}}) \to R(\Phi_B, \Phi_{\text{rad}})/B(\Phi_B)$, same in soft/collinear limits
- Build the Sudakov with R / B (which guarantees that the square bracket is still **unitary**)

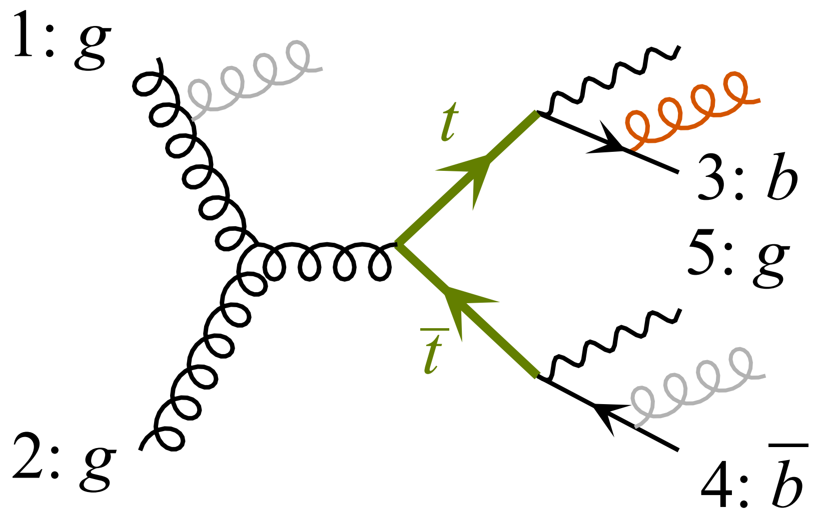
$$d\sigma_{\text{POWHEG}} = \overline{B}(\Phi_B)\left[\Delta(p_{T,\min}) + d\Phi_{\text{rad}}\Delta(p_T(\Phi_{\text{rad}}))\frac{R(\Phi_B, \Phi_{\text{rad}})}{B(\Phi_B)}\right]$$

where $\overline{B}(\Phi_B) = B(\Phi_B) + V(\Phi_B) + \int d\Phi_r(R(\Phi_B, \Phi_r) - C(\Phi_B, \Phi_r))$

Singular regions

Where do we attach emissions?

- $R(\Phi_R)$ can have several collinear singularities (different emitter-emitted pairs)
- The real phase space can be split into pieces with only **one** collinear singularity, labelled by α



Singular regions:
- (1,5) & (2,5)
- (3,5)
- (4,5)

Singular regions

Where do we attach emissions?

- $R(\Phi_R)$ can have several collinear singularities (different emitter-emitted pairs)
- The real phase space can be split into pieces with only **one** collinear singularity, labelled by α

$$d\sigma_{\text{POWHEG}} = \overline{B}(\Phi_B)\left[\Delta(p_{T,\min}) + \sum_{\alpha} d\Phi^{\alpha}_{\text{rad}} \Delta(p_T(\Phi^{\alpha}_{\text{rad}})) \frac{R^{\alpha}(\Phi_B, \Phi^{\alpha}_{\text{rad}})}{B(\Phi_B)}\right]$$

where now $\overline{B}(\Phi_B) = B(\Phi_B) + V(\Phi_B) + \sum_{\alpha} \int d\Phi^{\alpha}_r (R^{\alpha}(\Phi_B, \Phi^{\alpha}_r) - C^{\alpha}(\Phi_B, \Phi^{\alpha}_r))$

- First and foremost we need mappings $f^{\alpha} : \Phi_R \leftrightarrow (\Phi_B, \Phi^{\alpha}_{\text{rad}})$, typically with global recoil

Singular regions

Where do we attach emissions?

- $R(\Phi_R)$ can have several collinear singularities (different emitter-emitted pairs)
- The real phase space can be split into pieces with only **one** collinear singularity, labelled by α

$$d\sigma_{\text{POWHEG}} = \overline{B}(\Phi_B)\left[\Delta(p_{T,\text{min}}) + \sum_\alpha d\Phi^\alpha_{\text{rad}}\Delta(p_T(\Phi^\alpha_{\text{rad}}))\frac{R^\alpha(\Phi_B,\Phi^\alpha_{\text{rad}})}{B(\Phi_B)}\right]$$

where now $\overline{B}(\Phi_B) = B(\Phi_B) + V(\Phi_B) + \sum_\alpha \int d\Phi^\alpha_r (R^\alpha(\Phi_B,\Phi^\alpha_r) - C^\alpha(\Phi_B,\Phi^\alpha_r))$

- First and foremost we need mappings $f^\alpha : \Phi_R \leftrightarrow (\Phi_B, \Phi^\alpha_{\text{rad}})$, typically with global recoil
- For each α we also define a corresponding transverse momentum p_T^α that enters the Sudakov
- And we need a decomposition of R into contributions labelled by singular regions α

$$R(\Phi_B,\Phi^\alpha_{\text{rad}}) = \sum_\alpha R^\alpha(\Phi_B,\Phi^\alpha_{\text{rad}})$$

  - In practice this is done with smooth functions $D_\alpha(\Phi_B,\Phi^\alpha_{\text{rad}})$ such that $\sum_\alpha D_\alpha = 1$ and $R^\alpha = D_\alpha R$

Singular regions

Where do we attach emissions?

- $R(\Phi_R)$ can have several collinear singularities (different emitter-emitted pairs)
- The real phase space can be split into pieces with only **one** collinear singularity, labelled by α

$$d\sigma_{\text{POWHEG}} = \overline{B}(\Phi_B)\left[\Delta(p_{T,\min}) + \sum_\alpha d\Phi^\alpha_{\text{rad}}\Delta(p_T(\Phi^\alpha_{\text{rad}}))\frac{R^\alpha(\Phi_B,\Phi^\alpha_{\text{rad}})}{B(\Phi_B)}\right]$$

where now $\overline{B}(\Phi_B) = B(\Phi_B) + V(\Phi_B) + \sum_\alpha \int d\Phi^\alpha_r (R^\alpha(\Phi_B,\Phi^\alpha_r) - C^\alpha(\Phi_B,\Phi^\alpha_r))$

- First and foremost we need mappings $f^\alpha : \Phi_R \leftrightarrow (\Phi_B, \Phi^\alpha_{\text{rad}})$, typically with global recoil
- For each α we also define a corresponding transverse momentum p_T^α that enters the Sudakov
- And we need a decomposition of R into contributions labelled by singular regions α

$$R(\Phi_B,\Phi^\alpha_{\text{rad}}) = \sum_\alpha R^\alpha(\Phi_B,\Phi^\alpha_{\text{rad}})$$

  - In practice this is done with smooth functions $D_\alpha(\Phi_B,\Phi^\alpha_{\text{rad}})$ such that $\sum_\alpha D_\alpha = 1$ and $R^\alpha = D_\alpha R$
- Some of the objects definitions depend on the subtraction scheme (FKS, CS, …)

Sudakov veto loop

How do we attach emissions?

- Goal is to generate a POWHEG emission according to the unitary square bracket: $[\Delta(p_{T,\min}) + \sum_\alpha d\Phi^\alpha_{\rm rad}\Delta(p_T(\Phi^\alpha_{\rm rad}))R(\Phi_B, \Phi^\alpha_{\rm rad})/B(\Phi_B)]$
- For a given Φ_B (chosen with weight $\overline{B}(\Phi_B)$), start at a maximal scale $t_{\max}(\Phi_B)$
- Repeatedly:
  1. propose a trial emission at some lower scale t with radiation variables $\Phi^\alpha_{\rm rad}$
  2. perform a Sudakov veto step to decide whether to accept this emission

† `scalup`: shower starting scale

Sudakov veto loop

How do we attach emissions?

- Goal is to generate a POWHEG emission according to the unitary square bracket:
  $[\Delta(p_{T,\min}) + \sum_\alpha d\Phi_{\text{rad}}^\alpha \Delta(p_T(\Phi_{\text{rad}}^\alpha)) R(\Phi_B, \Phi_{\text{rad}}^\alpha)/B(\Phi_B)]$
- For a given Φ_B (chosen with weight $\overline{B}(\Phi_B)$), start at a maximal scale $t_{\max}(\Phi_B)$
- Repeatedly:
  1. propose a trial emission at some lower scale t with radiation variables Φ_{rad}^α
  2. perform a Sudakov veto step to decide whether to accept this emission
- If a proposal is accepted:
  - we return this emission as the hardest emission in region α with $t = p_T(\Phi_{\text{rad}}^\alpha)$

[†] `scalup`: shower starting scale

Sudakov veto loop

How do we attach emissions?

- Goal is to generate a POWHEG emission according to the unitary square bracket:
  $[\Delta(p_{T,\min}) + \sum_\alpha d\Phi^\alpha_{\rm rad}\Delta(p_T(\Phi^\alpha_{\rm rad}))R(\Phi_B,\Phi^\alpha_{\rm rad})/B(\Phi_B)]$
- For a given Φ_B (chosen with weight $\overline{B}(\Phi_B)$), start at a maximal scale $t_{\max}(\Phi_B)$
- Repeatedly:
  1. propose a trial emission at some lower scale t with radiation variables $\Phi^\alpha_{\rm rad}$
  2. perform a Sudakov veto step to decide whether to accept this emission
- If a proposal is accepted:
  - we return this emission as the hardest emission in region α with $t = p_T(\Phi^\alpha_{\rm rad})$
- If successive proposals are rejected until t drops below $p_{T,\min}$:
  - no emission harder than $p_{T,\min}$ is generated

† `scalup`: shower starting scale

Sudakov veto loop

How do we attach emissions?

- Goal is to generate a POWHEG emission according to the unitary square bracket: $[\Delta(p_{T,\min}) + \sum_\alpha d\Phi^\alpha_{\text{rad}} \Delta(p_T(\Phi^\alpha_{\text{rad}}))R(\Phi_B, \Phi^\alpha_{\text{rad}})/B(\Phi_B)]$
- For a given Φ_B (chosen with weight $\overline{B}(\Phi_B)$), start at a maximal scale $t_{\max}(\Phi_B)$
- Repeatedly:
  1. propose a trial emission at some lower scale t with radiation variables Φ^α_{rad}
  2. perform a Sudakov veto step to decide whether to accept this emission
- If a proposal is accepted:
  - we return this emission as the hardest emission in region α with $t = p_T(\Phi^\alpha_{\text{rad}})$
- If successive proposals are rejected until t drops below $p_{T,\min}$:
  - no emission harder than $p_{T,\min}$ is generated
- Finally: this is done in each singular region and all emissions are sorted by hardness, descending
  - we keep only the first emission, construct Φ_R and return the event with `scalup`†=$p_T(\Phi^\alpha_{\text{rad}})$
  - if no emissions were generated, the event is kept as a pure Born event with `scalup`=$p_{T,\min}$

† `scalup`: shower starting scale

Checkpoint

Are there any questions?

Tuning the real cross section

The idea

- So far real matrix element decomposed into singular regions α, $R(\Phi_R) = \sum_\alpha R^\alpha(\Phi_B, \Phi_{\rm rad}^\alpha)$
- We can now further split each R^α into
  - R_s^α: a singular part exponentiated in the Sudakov
  - R_f^α: a finite part treated purely at fixed order

Tuning the real cross section

The idea

- So far real matrix element decomposed into singular regions α, $R(\Phi_R) = \sum_\alpha R^\alpha(\Phi_B, \Phi_{\text{rad}}^\alpha)$
- We can now further split each R^α into
  - R_s^α: a singular part exponentiated in the Sudakov
  - R_f^α: a finite part treated purely at fixed order
- With a damping factor $F(p_T)$: $F(p_T) \to 1$ in the soft/collinear region and $F(p_T) \to 0$ far away
- POWHEG's standard choice: $F(p_T) = h_{\text{damp}}^2 / \left(h_{\text{damp}}^2 + p_T^2\right)$.
- For each region α: $R_s^\alpha = R^\alpha F(p_T(\Phi_{\text{rad}}^\alpha))$ and $R_f^\alpha = R^\alpha - R_s^\alpha$

Tuning the real cross section

The idea

- So far real matrix element decomposed into singular regions α, $R(\Phi_R) = \sum_\alpha R^\alpha(\Phi_B, \Phi_{\text{rad}}^\alpha)$
- We can now further split each R^α into
  - R_s^α: a singular part exponentiated in the Sudakov
  - R_f^α: a finite part treated purely at fixed order
- With a damping factor $F(p_T)$: $F(p_T) \to 1$ in the soft/collinear region and $F(p_T) \to 0$ far away
- POWHEG's standard choice: $F(p_T) = h_{\text{damp}}^2 / \left(h_{\text{damp}}^2 + p_T^2\right)$.
- For each region α: $R_s^\alpha = R^\alpha F(p_T(\Phi_{\text{rad}}^\alpha))$ and $R_f^\alpha = R^\alpha - R_s^\alpha$
- POWHEG formula splits into two parts:
  - Exponentiated, `btilde` (**soft**), part: $d\sigma_{\text{POWHEG}} \to d\sigma_{\text{POWHEG}}^s$ with $R^\alpha \to R_s^\alpha$, including `overline` B
  - `remnant` (**hard**) part: $d\sigma_{\text{POWHEG}}^f = \sum_\alpha d\Phi_R^\alpha R_f^\alpha(\Phi_R^\alpha)$

Tuning the real cross section

The idea

- So far real matrix element decomposed into singular regions α, $R(\Phi_R) = \sum_\alpha R^\alpha(\Phi_B, \Phi_{\text{rad}}^\alpha)$
- We can now further split each R^α into
  - R_s^α: a singular part exponentiated in the Sudakov
  - R_f^α: a finite part treated purely at fixed order
- With a damping factor $F(p_T)$: $F(p_T) \to 1$ in the soft/collinear region and $F(p_T) \to 0$ far away
- POWHEG's standard choice: $F(p_T) = h_{\text{damp}}^2 / \left(h_{\text{damp}}^2 + p_T^2\right)$.
- For each region α: $R_s^\alpha = R^\alpha F(p_T(\Phi_{\text{rad}}^\alpha))$ and $R_f^\alpha = R^\alpha - R_s^\alpha$
- POWHEG formula splits into two parts:
  - Exponentiated, `btilde` (**soft**), part: $d\sigma_{\text{POWHEG}} \to d\sigma_{\text{POWHEG}}^s$ with $R^\alpha \to R_s^\alpha$, including `overline` B
  - `remnant` (**hard**) part: $d\sigma_{\text{POWHEG}}^f = \sum_\alpha d\Phi_R^\alpha R_f^\alpha(\Phi_R^\alpha)$
- h_{damp} provides a smooth switch:
  - for $p_T \ll h_{\text{damp}}$: $F(p_T) \approx 1 \Rightarrow$ fully exponentiated
  - for $p_T \gg h_{\text{damp}}$: $F(p_T) \approx 0 \Rightarrow$ treated as remnant

Tuning the real cross section

But why

- Complete R/B exponentiation pulls the inclusive NLO K-factor into the hard tail
  - for large p_T: $d\sigma_{\text{POWHEG}} \sim \left(\overline{B}/B\right)R$
  - can noticeably over-enhance the high-p_T region vs fixed-order NLO
- h_{damp} splits $R = R_s + R_f$ to localise exponentiation and reduces the local K-factor ($\overline{B}/B$), while preserving NLO accuracy
  - R_s: matches R in the soft/collinear region and is exponentiated with $\overline{B}$
  - R_f: finite hard part, treated additively, follows fixed-order behaviour

Tuning the real cross section

But why

- Complete R/B exponentiation pulls the inclusive NLO K-factor into the hard tail
  - for large p_T: $d\sigma_{\text{POWHEG}} \sim \left(\overline{B}/B\right)R$
  - can noticeably over-enhance the high-p_T region vs fixed-order NLO
- h_{damp} splits $R = R_s + R_f$ to localise exponentiation and reduces the local K-factor ($\overline{B}/B$), while preserving NLO accuracy
  - R_s: matches R in the soft/collinear region and is exponentiated with $\overline{B}$
  - R_f: finite hard part, treated additively, follows fixed-order behaviour

- How to choose h_{damp}? No universal answer! (What does “hard” mean for your process?)
  - Default is a fixed value, but it can be made dynamic
  - Experiments tune it
  - Varying it should be part of the modelling uncertainty

Tuning the real cross section

A cool sideffect

- There is also `bornzerodamp`, which shifts all contributions that are "numerically" far away from soft/collinear approx into **`remnant`**
  - If $R_s^\alpha > N(R_{\text{soft}}^\alpha + R_{\text{coll}}^\alpha - R_{\text{coll-soft}}^\alpha)$ it concludes something goes wrong and shifts the event into the remnant. Examples:
    - When B is zero and R is not
    - When there is an enhancement spoiled by a recoil (e.g. intermediate resonance, $g \to b\bar{b}$ splittings)

Tuning the real cross section

A cool sideffect

- There is also `bornzerodamp`, which shifts all contributions that are "numerically" far away from soft/collinear approx into **`remnant`**
  - If $R_s^\alpha > N(R_{\text{soft}}^\alpha + R_{\text{coll}}^\alpha - R_{\text{coll-soft}}^\alpha)$ it concludes something goes wrong and shifts the event into the remnant. Examples:
    - When B is zero and R is not
    - When there is an enhancement spoiled by a recoil (e.g. intermediate resonance, $g \to b\bar{b}$ splittings)

Keep in mind

- **`btilde`**: one factor of α_s in R is evaluated at $p_T(\Phi_{\text{rad}})$ (and is never subject to scale variations), remaining powers at μ_R
- **`remnant`** R_f: all powers of α_s are usually taken at μ_R
- Caveat: as soon as $R_s \neq R$ the exact NLO cancellation of μ_R dependence between virtual + real is spoiled beyond the singular part
  - always compare POWHEG μ_R, μ_F variations to the fixed order