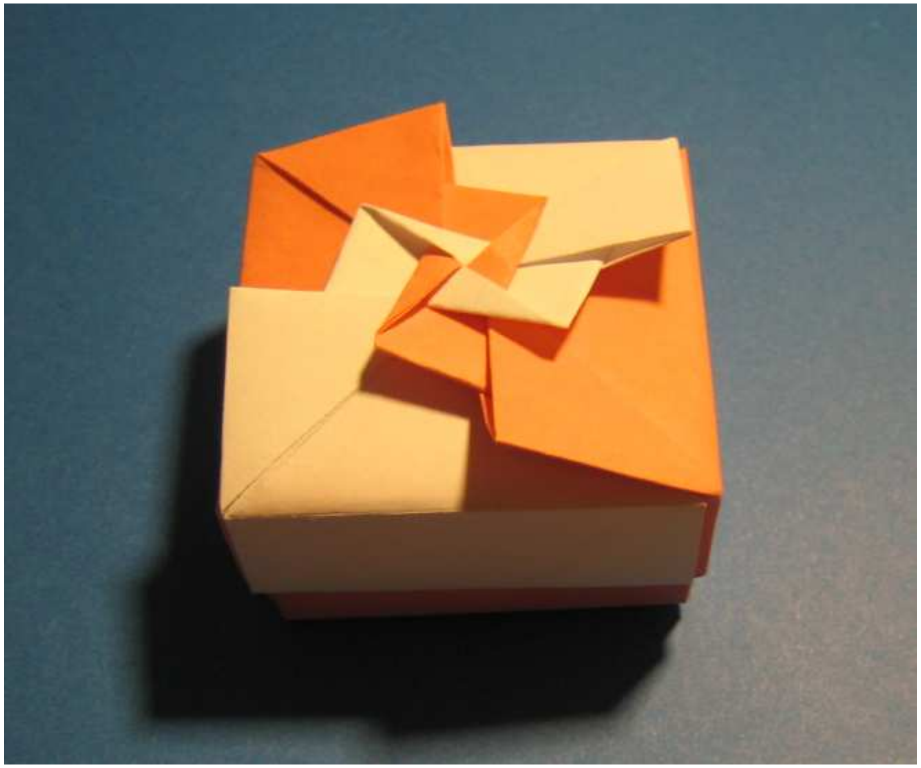
MATCHING NLO CALCULATIONS WITH PARTON SHOWER: THE POSITIVE-WEIGHT HARDEST EMISSION GENERATOR

Carlo Oleari
Università di Milano-Bicocca, Milan

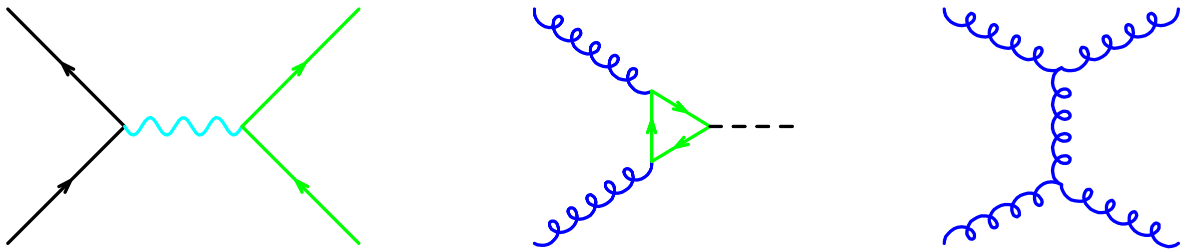
LAPTh, Annecy
29 November – 1 December 2011

- Theory introduction
  - Basics of shower Monte Carlo programs
  - The POWHEG formalism
- The POWHEG BOX
  - What is needed in the POWHEG BOX
  - How to run the POWHEG BOX



High energy collisions

High-energy particle physics deals with the scattering and the production of elementary constituents

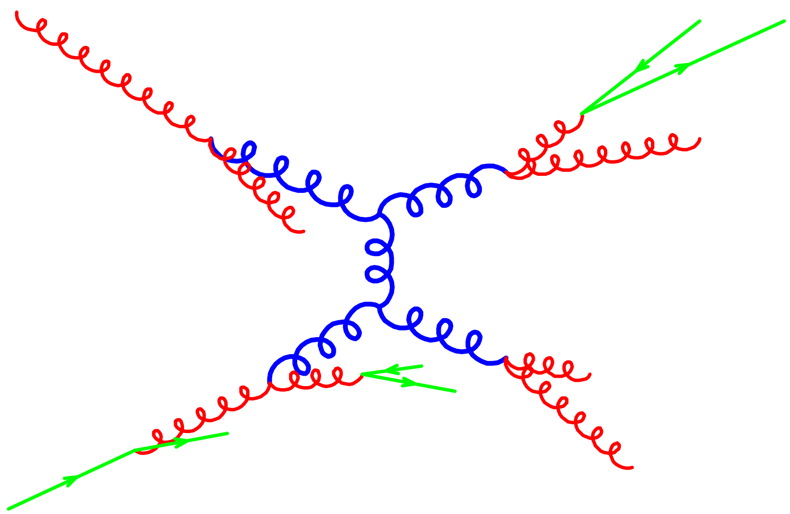


$e^+e^- \longrightarrow q\bar{q}$ $\qquad gg \longrightarrow H$ $\qquad gg \longrightarrow gg$

Ideally, one needs elementary constituents as projectiles and targets, (i.e. a collider for leptons, gluons and quarks) and a final-state detector of leptons, gluons and quarks. Not obvious for quarks and gluons:

- at short distance, due to asymptotic freedom, quarks and gluons behave as free particles
- at long distance, infrared slavery: very strong interactions hide the simplicity of the description of the constituents.

Dominant corrections

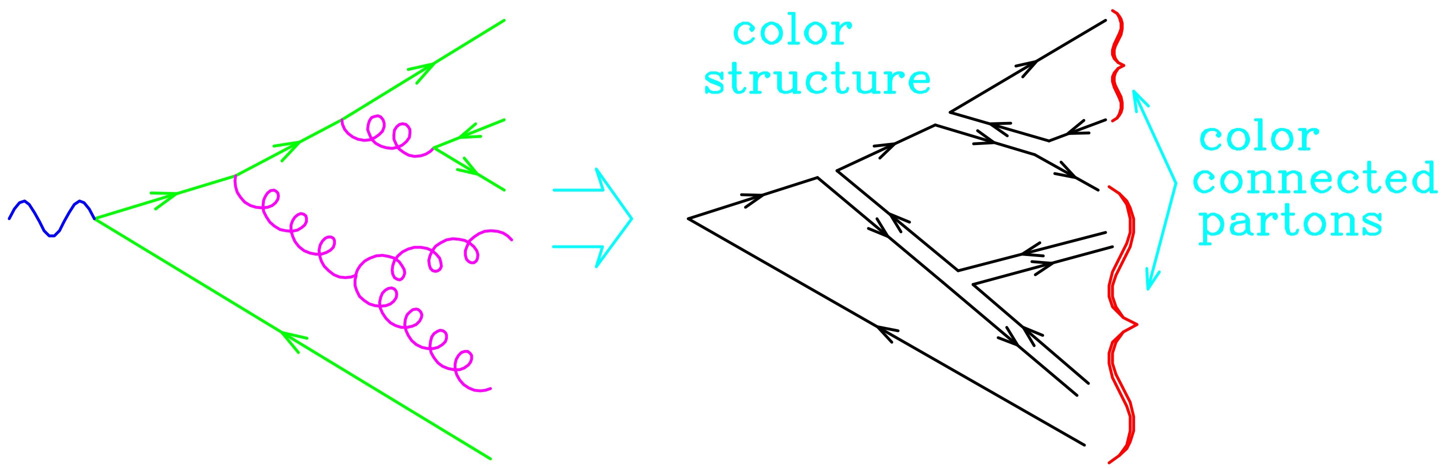


Collinear-splitting processes in the initial and final state (always with transverse momenta $> \Lambda_{QCD}$) are strongly enhanced. This is due to the fact that, in perturbation theory, the denominators in the propagators are small.

- The algorithms that evaluate all these enhanced contributions are called shower algorithms.
- Shower algorithms give a description of a hard collision up to distances of order $1/\Lambda_{QCD}$.
- At larger distances, perturbation theory breaks down and we need to resort to non-perturbative methods (i.e. lattice calculations). However, these methods can be applied only to simple systems. The only viable alternative is to use models of hadron formation.

Color and hadronization

Shower Monte Carlo programs assign color labels to partons. Only color connections are recorded (in large N_c limit). The initial color is assigned according to hard cross section.



Color assignments are used in the hadronization model.

Most popular models: Lund string model, cluster model.

In all models, color singlet structures are formed out of color connected partons, and are decayed into hadrons, preserving energy and momentum.

Hadronic final states

IHEP	ID	IDPDG	IST	MO1	MO2	DA1	DA2	P-X	P-Y	P-Z	ENERGY	MASS	V-X	V-Y	V-Z	V-C*T
30	NU_E	12	1	28	23	0	0	64.30	25.12	-1194.4	1196.4	0.00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
31	E+	-11	1	29	23	0	0	-22.36	6.19	-234.2	235.4	0.00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
230	PI0	111	1	155	24	0	0	0.31	0.38	0.9	1.0	0.13	4.209E-11	6.148E-11	-3.341E-11	5.192E-10
231	RHO+	213	197	155	24	317	318	-0.06	0.07	0.1	0.8	0.77	4.183E-11	6.130E-11	-3.365E-11	5.189E-10
232	P	2212	1	156	24	0	0	0.40	0.78	1.0	1.6	0.94	4.156E-11	6.029E-11	-4.205E-11	5.250E-10
233	NBAR	-2112	1	156	24	0	0	-0.13	-0.35	-0.9	1.3	0.94	4.168E-11	6.021E-11	-4.217E-11	5.249E-10
234	PI-	-211	1	157	9	0	0	0.14	0.34	286.9	286.9	0.14	4.660E-13	8.237E-12	1.748E-09	1.749E-09
235	PI+	211	1	157	9	0	0	-0.14	-0.34	624.5	624.5	0.14	4.056E-13	8.532E-12	2.462E-09	2.462E-09
236	P	2212	1	158	9	0	0	-1.23	-0.26	0.9	1.8	0.94	-4.815E-11	1.893E-11	7.520E-12	3.252E-10
237	DLTABR--	-2224	197	158	9	319	320	0.94	0.35	1.6	2.2	1.23	-4.817E-11	1.900E-11	7.482E-12	3.252E-10
238	PI0	111	1	159	9	0	0	0.74	-0.31	-27.9	27.9	0.13	-1.889E-10	9.893E-11	-2.123E-09	2.157E-09
239	RHO0	113	197	159	9	321	322	0.73	-0.88	-19.5	19.5	0.77	-1.888E-10	9.859E-11	-2.129E-09	2.163E-09
240	K+	321	1	160	9	0	0	0.58	0.02	-11.0	11.0	0.49	-1.890E-10	9.873E-11	-2.135E-09	2.169E-09
241	KL_1-	-10323	197	160	9	323	324	1.23	-1.50	-50.2	50.2	1.57	-1.890E-10	9.879E-11	-2.132E-09	2.166E-09
242	K-	-321	1	161	24	0	0	0.01	0.22	1.3	1.4	0.49	4.250E-11	6.333E-11	-2.746E-11	5.211E-10
243	PI0	111	1	161	24	0	0	0.31	0.38	0.2	0.6	0.13	4.301E-11	6.282E-11	-2.751E-11	5.210E-10

High-energy experimental physicists feed this kind of output through their detector-simulation software, and use it to determine efficiencies for signal detection, and perform background estimates.

Analysis strategies are set up using these simulated data.

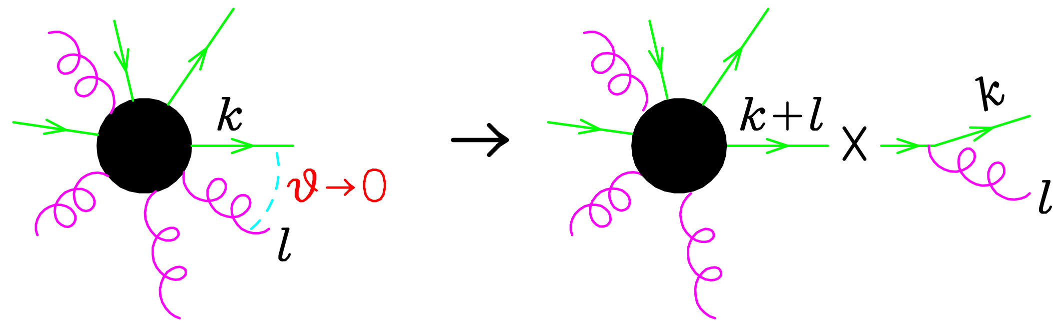
Summarizing

- In high-energy collider physics not many questions can be answered without a Shower Monte Carlo (SMC).
- The name shower comes from the fact that we dress a hard event with QCD radiation.
- After a latency period, many physicists are now looking at shower Monte Carlo models again, under different perspective: Catani, Krauss, Kühn & Webber; Mangano, Moretti, Piccinini, Pittau, Polosa & Treccani; Frixione & Webber; Kramer, Mrenna, Nagy & Soper; Giele, Kosower & Skands; Bauer & Schwartz; Schumann & Krauss; Dinsdale, Ternick & Weinzierl; ...
- Shower algorithms summarize most of our knowledge in perturbative QCD: infrared cancellations, Altarelli-Parisi equations, soft coherence, Sudakov form factors. Most of them have a simple interpretation in terms of shower algorithms.

Shower basics: collinear factorization

QCD emissions are enhanced near the collinear limit

Cross sections factorize near collinear limit



$$d\mathbf{\Phi}_{n+1} = d\mathbf{\Phi}_n \, d\Phi_r \qquad d\Phi_r \div dt\, dz\, d\varphi$$

$$|M_{n+1}|^2 d\mathbf{\Phi}_{n+1} \implies |M_n|^2 \, d\mathbf{\Phi}_n \, \frac{\alpha_s}{2\pi} \, \frac{dt}{t} P_{q,qg}(z)\, dz \, \frac{d\varphi}{2\pi} \quad \begin{cases} \dfrac{dt}{t} \approx \dfrac{d\theta}{\theta} & \text{collinear singularity} \\[2ex] \dfrac{dz}{1-z} \approx \dfrac{dE_g}{E_g} & \text{soft singularity} \end{cases}$$

t : $(k+l)^2,\ p_T^2,\ E^2\theta^2 \ldots$

$z = k^0/(k^0+l^0)$: energy (or $p_\parallel$ or p^+) fraction of quark

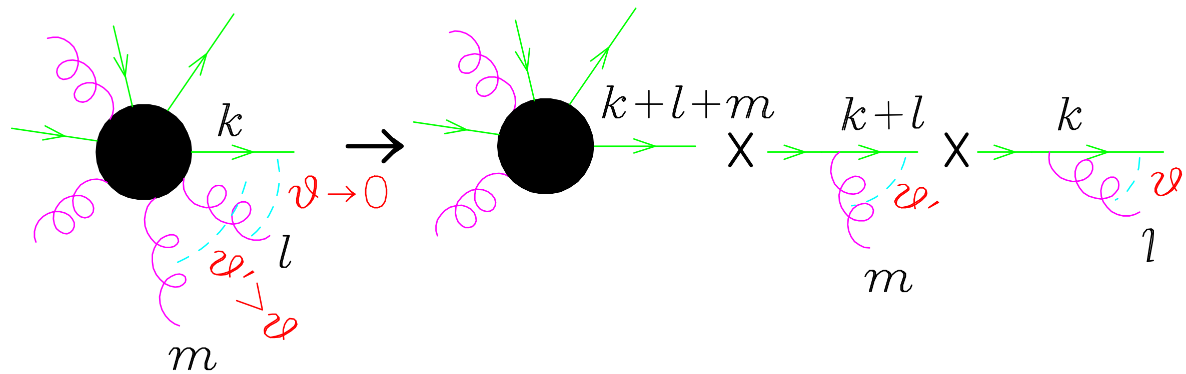
$P_{q,qg}(z) = C_F \dfrac{1+z^2}{1-z}$: Altarelli-Parisi splitting function

(ignore $z \to 1$ IR divergence for now)

Shower basics: collinear factorization

If another gluon becomes collinear, iterate the previous formula

$\theta', \theta \to 0$ with $\theta' > \theta$



$$|M_{n+1}|^2 d\mathbf{\Phi}_{n+1} \quad \Longrightarrow \quad |M_{n-1}|^2 d\mathbf{\Phi}_{n-1} \times \frac{\alpha_s}{2\pi} \frac{dt'}{t'} P_{q,qg}(z')\, dz' \frac{d\varphi'}{2\pi} \times \frac{\alpha_s}{2\pi} \frac{dt}{t} P_{q,qg}(z)\, dz \frac{d\varphi}{2\pi} \theta(t'-t)$$

Collinear partons can be described by a factorized integral ordered in t.

Collinear factorization: multiple emissions

For n collinear emissions, the cross section goes as

$$\sigma \approx \sigma_0 \, \alpha_s^n \int_{t_0}^{Q^2} \frac{dt_1}{t_1} \frac{dt_2}{t_2} \ldots \frac{dt_n}{t_n} \, \theta\left(Q^2 > t_1 > t_2 > \ldots > t_n > t_0\right)$$

$$= \sigma_0 \, \alpha_s^n \int_{t_0}^{Q^2} \frac{dt_1}{t_1} \int_{t_0}^{t_1} \frac{dt_2}{t_2} \ldots \int_{t_0}^{t_{n-1}} \frac{dt_n}{t_n} \approx \sigma_0 \, \alpha_s^n \frac{1}{n!} \left(\log \frac{Q^2}{t_0}\right)^n$$

- Q^2 is an upper cutoff for the ordering variable t
- $t_0 \approx \Lambda^2 \approx \Lambda^2_{\rm QCD}$ is an infrared cutoff (quark mass, confinement scale)
- Due to the log dependence, we call it leading-log approximation.
- According to the Kinoshita-Lee-Nauenberg theorem, the virtual corrections, order by order, contribute with a comparable term, with opposite sign.
- The virtual leading-log contribution should be included in order to get sensible results!

Simple probabilistic interpretation of "not-resolved" corrections

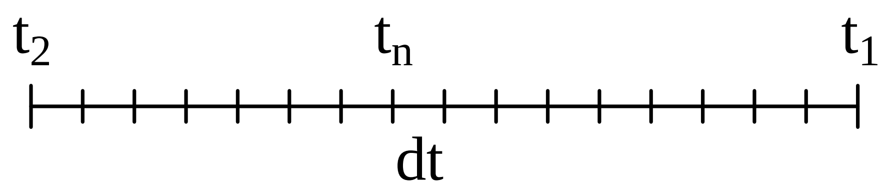
- probability of emission in the interval dt, at order α_s (multiple emissions are of higher orders in α_s)

$$dP_{\rm emis}(t+dt,t) = \frac{dt}{t}\,\frac{\alpha_s(t)}{2\pi}\int dz\, P_{i,jk}(z)$$

- probability of no emission in the interval dt

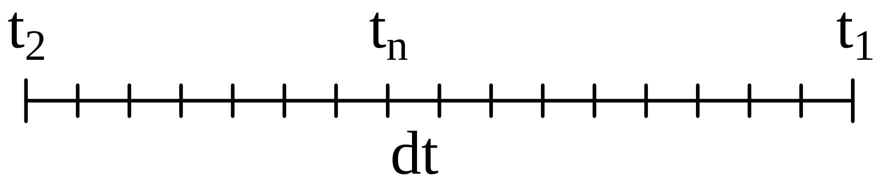
$$dP_{\text{no emis}}(t+dt,t) = 1 - dP_{\rm emis}(t+dt,t) = 1 - \frac{dt}{t}\,\frac{\alpha_s(t)}{2\pi}\int dz\, P_{i,jk}(z)$$

The "no emission" probability contains, through the 1, all the virtual corrections (in the collinear approximation, that is at the leading-log level).



Simple probabilistic interpretation of "not-resolved" corrections

- divide a finite interval $[t_2, t_1]$ in N small intervals $dt = (t_1 - t_2)/N$.



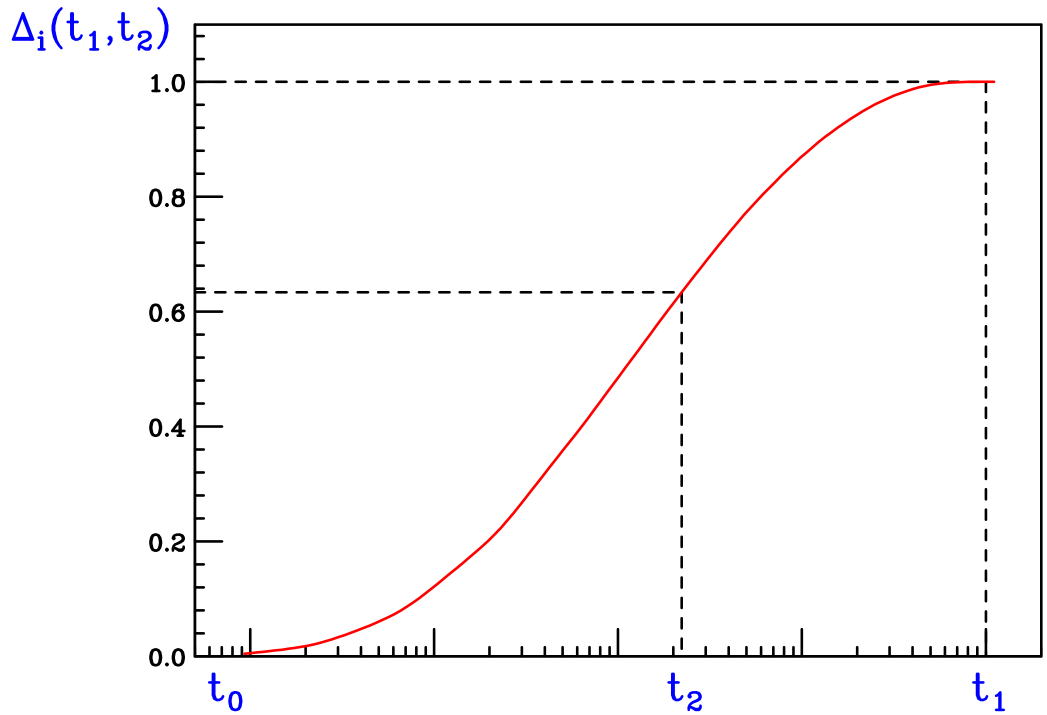
The probability of not emitting radiation between the two ordering scales t_1 and t_2 is given by the product

$$
\begin{aligned}
P_{\text{no emis}}(t_1, t_2) &= \lim_{N \to \infty} \prod_{n=1}^{N} \left[1 - \frac{dt}{t_n} \frac{\alpha_s(t_n)}{2\pi} \int dz\, P_{i,jk}(z) \right] \\
&= \exp\left\{ - \int_{t_2}^{t_1} \frac{dt}{t} \frac{\alpha_s(t)}{2\pi} \int dz\, P_{i,jk}(z) \right\} \\
&\equiv \Delta(t_1, t_2)
\end{aligned}
$$

- The weight $\Delta(t_1, t_2)$ is called Sudakov form factor. It resums all the dominant virtual corrections to the tree graph (in the collinear approximation).

Sudakov form factors

$$\Delta_i(t_1, t_2) = \exp\left\{ -\sum_{jk} \int_{t_2}^{t_1} \frac{dt}{t} \frac{\alpha_S(t)}{2\pi} \int dz\, P_{i,jk}(z) \right\}$$



Notice that, when $t_2 \ll t_1$, $\Delta \to 0$, i.e. the probability that a hard parton turns into a narrow jet, or that it does not radiate at all, is small (it is Sudakov suppressed)

First branching

The probability of the first branching is independent of subsequent branchings because of Kinoshita-Lee-Nauenberg cancellation. It is given by

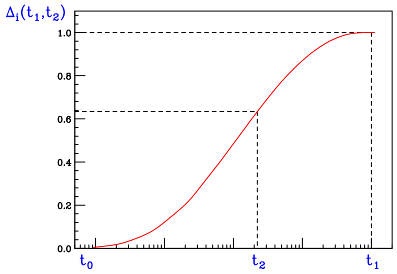
$$dP_{\text{first}} = \Delta_i(t,t')\,\frac{\alpha_S(t')}{2\pi}\,\frac{dt'}{t'}\,P_{i,jk}(z)\,dz\,\frac{d\varphi}{2\pi}$$

Upon integrating in z and φ, and summing over jk, we have

$$dP_{\text{first}} = \Delta_i(t,t')\,\frac{\alpha_S(t')}{2\pi}\,\frac{dt'}{t'}\int\sum_{(jk)}P_{i,jk}(z)\,dz\,\frac{d\varphi}{2\pi} = d\Delta_i(t,t')$$

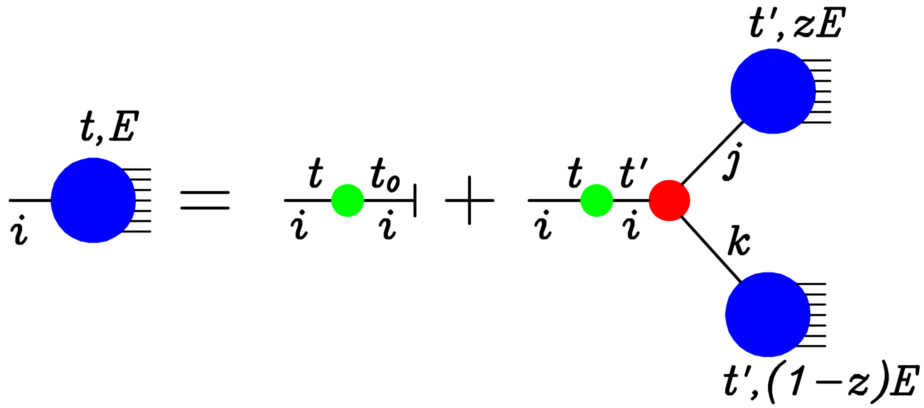
i.e. the distribution is uniform in the Sudakov form factor.
The integral over the whole t' range, from the minimum value t_0 (IR cutoff) up to t, is given by

$$\int_{t_0}^{t} dP_{\text{first}} = \int_{t_0}^{t} d\Delta_i(t,t') = \Delta_i(t,t) - \Delta_i(t,t_0) = 1 - 0 = 1$$



as it should be for a correct probabilistic interpretation.

Final recipe



$$\mathcal{S}_i(t,E) = \Delta_i(t,t_0)\,\mathbb{1} + \sum_{(jk)} \int_{t_0}^{t} \frac{\alpha_S(t')}{2\pi} \frac{dt'}{t'} \int dz \int \frac{d\varphi}{2\pi} \Delta_i(t,t')\, P_{i,jk}(z)\, \mathcal{S}_j(t',zE)\ \mathcal{S}_k(t',(1-z)E)$$

- consider all tree graphs.
- assign values to the radiation variables Φ_r (t, z and φ) to each vertex.
- at each vertex, $i \to jk$, include a factor

$$\frac{dt}{t} dz \frac{\alpha_S(t)}{2\pi} P_{i,jk}(z) \frac{d\varphi}{2\pi}$$

- include a factor $\Delta_i(t_1,t_2)$ to each internal parton i, from hardness t_1 to hardness t_2.
- include a factor $\Delta_i(t,t_0)$ on final lines (t_0 = IR cutoff)

Actual implementation of the shower algorithm

We start from a given value of the ordering variable t. We want to generate the value t' for the next emission, according to the probability

$$dP_{\rm first} = \Delta_i(t,t')\,\frac{\alpha_S(t')}{2\pi}\,\frac{dt'}{t'}\int\sum_{(jk)} P_{i,jk}(z)\,dz\,\frac{d\varphi}{2\pi} = d\Delta_i(t,t')$$

Since this is an exact differential form, we proceed as in the case we want to generate a random variable x according to a distribution function $f(x)$, whose indefinite integral is known, starting from a uniform random variable r

$$dP = f(X)\,dX = 1\,dR \qquad \text{where} \qquad f(X)\,dX = dF(X)$$

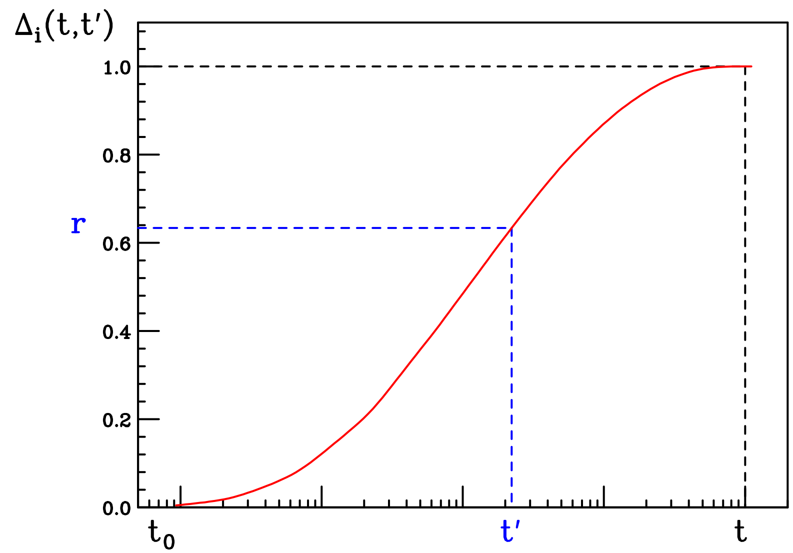
$$\int_{x_{\rm min}}^{x} f(X)\,dX = F(x) = \int_0^r 1\,dR = r \qquad \Longrightarrow \qquad x = F^{-1}(r)$$

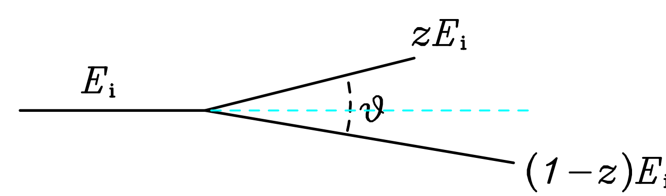
Actual implementation of the shower algorithm

✓ generate a hard process configuration with a probability proportional to its parton-level cross section. Parton densities are evaluated at the typical "high" scale Q of the process

✓ for each final-state colored parton, generate a shower

- set $t = Q^2$
- generate a uniform random number $0 < r < 1$
- solve the equation $\Delta_i(t, t') = r$ for t'
- if $t' < t_0$ stop here (final state line). Begin hadronization
- if $t' > t_0$, generate z, jk with probability $P_{i,jk}(z)$, and $0 < \varphi < 2\pi$ uniformly. Assign energies $E_j = zE_i$ and $E_k = (1-z)E_i$ to partons j and k. The angle θ between their momenta is fixed by t' and with φ their direction is completely specified
- restart shower from each of the two branched parton j and k, setting the ordering parameter $t = t'$.





Shower algorithm

✓ for each initial-state colored parton, generate a shower in a similar way, but using a "trick": the backward evolution (Sjöstrand)

$$\frac{f_i^h(t',x)\,\Delta(t,t')}{f_i^h(t,x)} = r$$

where f_i^h is the parton density for the colliding hadron h, where parton i carries a momentum fraction $x = E_i/E_h$

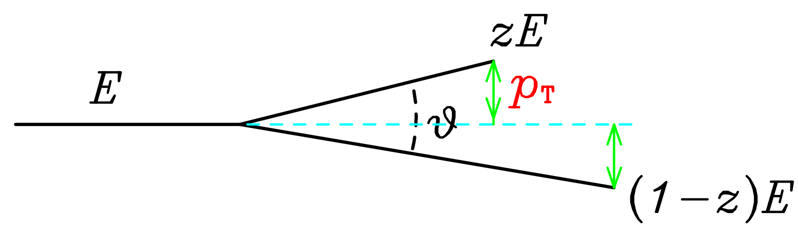
Some momentum reshuffling is needed in order to preserve local (at each vertex) and global momentum conservation

Accuracy: soft divergences and double-log regions

$z \to 1$ ($z \to 0$) region problematic. In fact, for $z \to 1$, $P_{qq}, P_{gg} \div 1/(1-z)$

The choice of the ordering variable t makes a difference

$$
\begin{array}{lccl}
\text{virtuality:} & t & \equiv & E^2 z(1-z) \overbrace{\theta^2}^{2(1-\cos\theta)} \\
p_T^2: & t & \equiv & E^2 z^2 (1-z)^2 \theta^2 \\
\text{angle:} & t & \equiv & E^2 \theta^2
\end{array}
$$



$$\text{virtuality}: \; z(1-z) > t/E^2 \quad \Longrightarrow \quad \int \frac{dt}{t} \int_{\sqrt{t}/E}^{1-\sqrt{t}/E} \frac{dz}{1-z} \approx \frac{1}{4} \log^2 \frac{t}{E^2}$$

$$p_T^2: \; z^2(1-z)^2 > t/E^2 \quad \Longrightarrow \quad \int \frac{dt}{t} \int_{t/E^2}^{1-t/E^2} \frac{dz}{1-z} \approx \frac{1}{2} \log^2 \frac{t}{E^2}$$

$$\text{angle}: \quad \Longrightarrow \quad \int \frac{dt}{t} \int_0^1 \frac{dz}{1-z} \approx \log t \, \log \Lambda$$

Sizable difference in double-log structure!

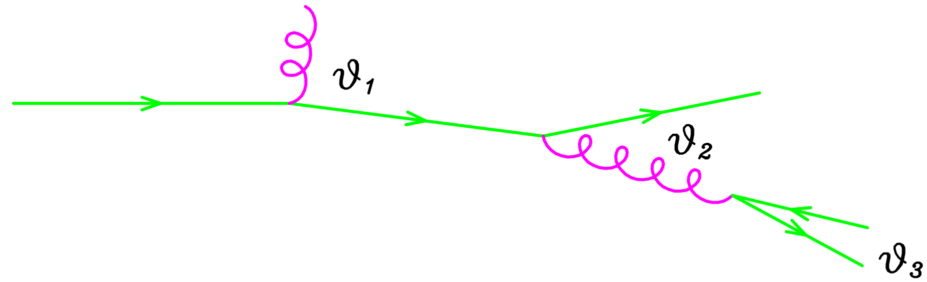
Angular ordering

Mueller (1981) showed that angular ordering is the correct choice

$$\frac{d\theta}{\theta}\,\frac{\alpha_s\left(p_T^2\right)}{2\pi}\,P(z)\,dz$$

$$\theta_1 > \theta_2 > \theta_3 \ldots$$

$$p_T^2 = E^2 z^2 (1-z)^2 \theta^2$$



$\alpha_s(p_T^2)$ for a correct treatment of charge renormalization in soft region (p_T^2 equals to the maximum virtuality of the gluon line).

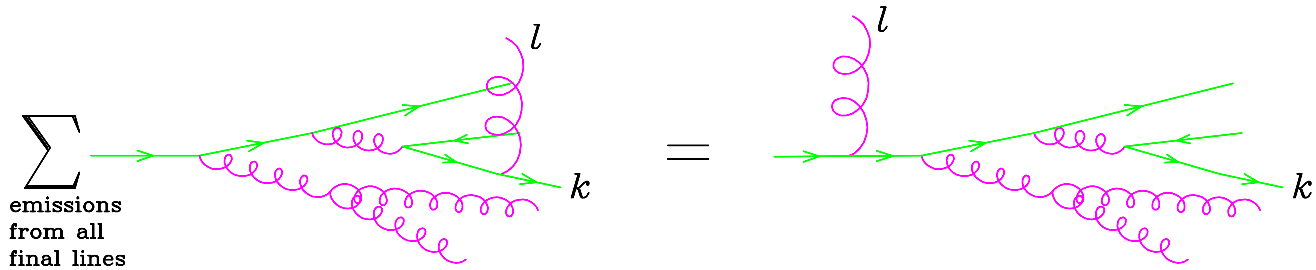
$$\Delta_i(t,t') = \exp\left[-\int_{t'}^{t}\frac{dt}{t}\int_{\sqrt{\frac{t_0}{t}}}^{1-\sqrt{\frac{t_0}{t}}} dz\,\frac{\alpha_s(p_T^2)}{2\pi}\sum_{(jk)} P_{i,jk}(z)\right]$$

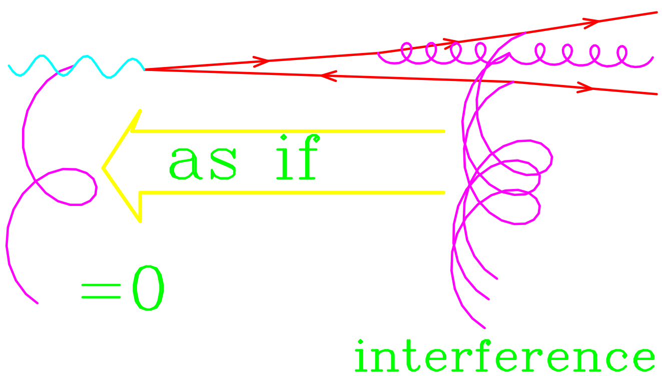
$$\approx \exp\left\{-\frac{c_i}{4\pi b_0}\left[\log\frac{t}{\Lambda^2}\log\frac{\log\frac{t}{\Lambda^2}}{\log\frac{t_0}{\Lambda^2}} - \log\frac{t}{t_0}\right]_{t'}^{t}\right\} \qquad (c_q = C_F, c_g = 2C_A)$$

Sudakov dumping stronger than any power of t.

Color coherence

Soft gluons emitted at large angles from final-state partons add coherently





- angular ordering accounts for soft gluon interference.
- intensity for photon jets $= 0$
- intensity for gluon jets $= C_A$ instead of $2\,C_F + C_A$

In angular-ordered shower Monte Carlo, large-angle soft emission is generated first.

Hardest emission, i.e. highest $p_T = E\,z(1-z)\,\theta$, in general, happens later.

Some available codes

- COJETS Odorico (1984)
- ISAJET Paige+Protopopescu (1986)
- FIELDAJET Field (1986)
- JETSET Sjöstrand (1986)
- PYTHIA Bengtsson+Sjöstrand (1987), Sjöstrand+Skands (2004)
- HERWIG Marchesini+Webber (1988), Marchesini+Webber+Abbiendi+Knowles+Seymour+Stanco (1992)
- ARIADNE Lönnblad (1992)
- SHERPA Gleisberg+Höche+Krauss+Schälicke+Schumann+Winter (2004)

Available accuracy$^{(*)}$

	collinear	soft-collinear	soft large-N_c	soft
PYTHIA	leading	partial	no	no
HERWIG	leading	leading	no	no
ARIADNE	partial	partial	leading	no
PYTHIA6.4	partial	partial	leading	no
SHERPA	leading	partial	no	no

One can realistically aim at

leading collinear, leading double log, leading soft in large-N_c limit

Soft effects for finite N_c require matrix exponentiation in the Sudakov form factor.

$^{(*)}$ At least, to my understanding

NLO + Parton Shower

LO-ME good for shapes. Uncertain absolute normalization

$$\alpha_s^n(2\mu) \approx \alpha_s^n(\mu)\left(1 - b_0\alpha_s(\mu)\log(4)\right)^n \approx \alpha_s^n(\mu)\left(1 - n\alpha_s(\mu)\right)$$

For $\mu = 100$ GeV, $\alpha_s = 0.12$, normalization uncertainty:

$W+1J$	$W+2J$	$W+3J$
$\pm 12\%$	$\pm 24\%$	$\pm 36\%$

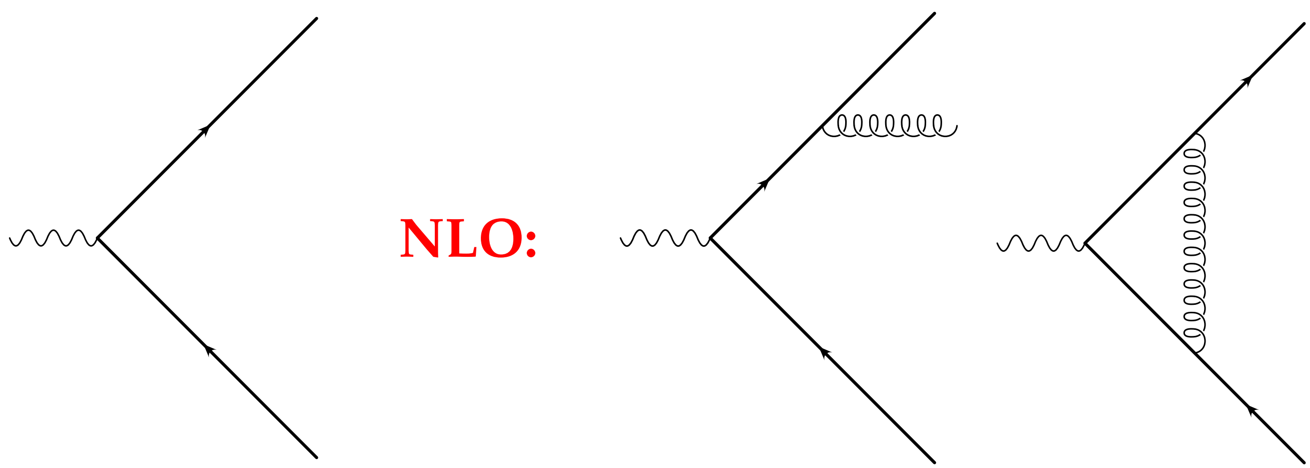
To improve on this, we need to go to NLO

- Positive experience with NLO calculations at LEP, HERA and Tevatron
- NLO results are cumbersome to compute: typically made up of an n-body (Born + virtual + soft and collinear remnants) and $(n+1)$-body (real emission) terms, both divergent (finite only when summed up).
- Merging NLO with shower is a natural extension of present approaches.

NLO + Parton Shower

The main problem in merging a NLO result and a Parton Shower is not to double-count radiation: the shower might produce some radiation already present at the NLO level (both at the virtual and at the real level).

LO: **NLO:**



NLO vs Shower Monte Carlo

NLO

- ✓ accurate shapes at high p_T
- ✓ normalization accurate at NLO order
- ✓ reduced dependence on renormalization and factorization scales
- ✗ wrong shapes at small p_T
- ✗ description only at the parton level

SMC (LO + shower)

- ✗ bad description at high p_T
- ✗ normalization accurate only at LO
- ✓ correct Sudakov suppression at small p_T
- ✓ simulate events at the hadron level

It is natural to try to merge the two approaches, keeping the good features of both

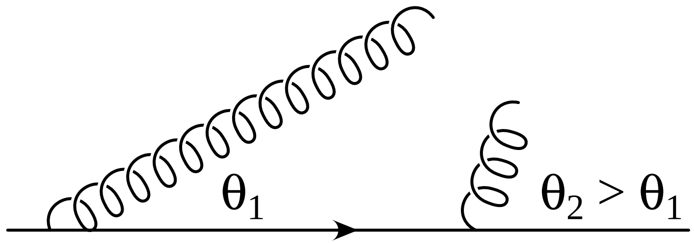
MC@NLO [Frixione and Webber, 2001] and POWHEG [Nason, 2004] do this in a consistent way

POWHEG: how it works

1. POWHEG, POsitive Weight Hardest Emission Generator, [Nason, hep-ph/0409146], generates first a partonic event with just one single emission, at NLO level, and with the correct probability in order not to have double-counting coming from (subsequent) radiation.
   The p_T of the produced radiation works as an upper cutoff for the p_T's of the entire subsequent shower: all the subsequent radition must be softer than the first one.

2. The event is written on a file using the standard Les Houches Interface and is processed by the Parton Shower program (HERWIG, PYTHIA...), that showers the event, but with a p_T less than the p_T generated by POWHEG (p_T veto).

POWHEG: truncated shower



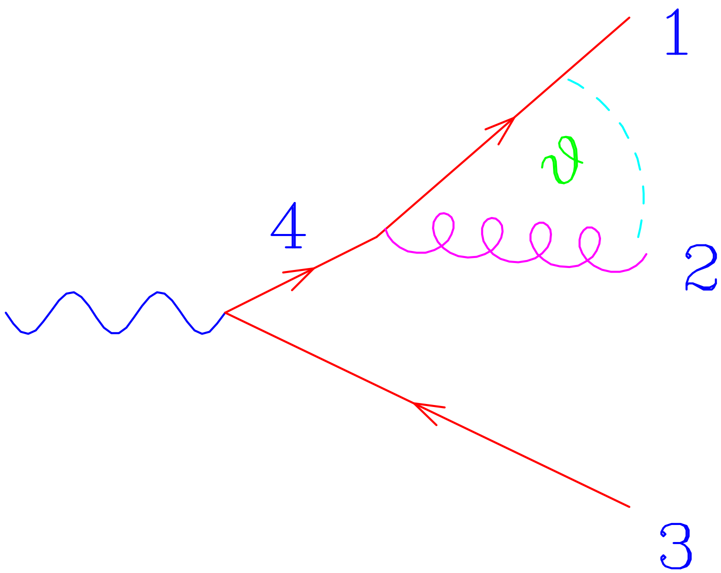
- if the shower is ordered in p_T (for example PYTHIA), nothing else needs to be done
- if the shower is ordered in angle (for example HERWIG), we need to generate correctly soft radiation at large angle.
  - pair up the partons that are nearest in p_T
  - generate an angular-ordered shower associated with the paired parton, stopping at the angle of the paired partons (truncated shower)
  - generate all subsequent vetoed showers

This is a problem that affects all the angular-ordered shower Monte Carlo programs when the shower is initiated by a relatively complex matrix element.

Truncated shower implemented only in HERWIG++

In the cases studied up to now, the effect of truncated shower is very small

Example of truncated shower: e^+e^-



- nearby partons: 1 and 2
- truncated shower: 1 and 2 pair, from θ up to a maximum angle. The truncated shower reintroduces coherent soft radiation from 1 and 2 at angles larger than θ (angular-ordered shower Monte Carlo programs generate those earlier).
- 1 and 2 shower from θ to cutoff
- 3 showers from maximum to cutoff

Truncated showers not yet implemented.

No evidence of effects from their absence in ZZ and e^+e^- production. Might be some effects in heavy-quark production.

Deeper into POWHEG

- In the next slides I will give more details of the POWHEG method

- It is impossible to demonstrate the whole method in a couple of hours. In fact, one has to show that:
  - it is possible to rearrange the shower in such a way that the hardest emission can be performed first. This has some consequences on an angular-ordered shower (truncated shower).
  - take charge of the generation of this first emission, and generate it according to the NLO amplitude, providing the appropriate Sudakov form factor for small transverse momentum
  - show that there is no double-counting

- More details in the original papers

Notation

We consider $2 \to n$ processes. $K_\oplus$ and $K_\ominus$ are the momenta of the incoming hadrons. Momentum conservation is enforced by

$$x_\oplus K_\oplus + x_\ominus K_\ominus \equiv k_\oplus + k_\ominus = k_1 + \ldots + k_n$$

$\mathbf{\Phi}_n$ is the set of variables

$$\mathbf{\Phi}_n = \{x_\oplus, x_\ominus, k_1, \ldots, k_n\}$$

If $\mathcal{B} = |M(2 \to n)|^2$ is the Born squared matrix element, then

$$\int d\mathbf{\Phi}_n \, B(\mathbf{\Phi}_n) \ldots \equiv \int dx_\oplus \, dx_\ominus \, d\Phi_n \, (k_\oplus + k_\ominus; k_1, \ldots, k_n) \; \mathrm{PDF}_\oplus(x_\oplus) \, \mathrm{PDF}_\ominus(x_\ominus) \, \mathcal{B}(\mathbf{\Phi}_n) \ldots$$

$$d\Phi_n \, (q; k_1, \ldots, k_n) = (2\pi)^4 \, \delta^4 \left(q - \sum_{i=1}^{n} k_i \right) \prod_{i=1}^{n} \frac{d^3 k_i}{(2\pi)^3 \, 2k_i^0}$$

and similar ones for the integral over the virtual contribution V, the integral of the real squared amplitude R and its counterterms C.

NLO calculations

We can always parametrize the $(n+1)$-body phase space $\mathbf{\Phi}_{n+1}$ in terms of the Born phase space $\mathbf{\Phi}_n$ and three radiation variables Φ_r: $\mathbf{\Phi}_{n+1} = \{\mathbf{\Phi}_n, \Phi_r\}$

$$\langle O \rangle = \int O \, d\sigma = \int d\mathbf{\Phi}_n \, O(\mathbf{\Phi}_n) \left[B(\mathbf{\Phi}_n) + V_b(\mathbf{\Phi}_n) \right] + \int d\mathbf{\Phi}_n \, d\Phi_r \, O(\mathbf{\Phi}_n, \Phi_r) \, R(\mathbf{\Phi}_n, \Phi_r)$$

where V_b is the (divergent) virtual differential cross section. The virtual and real-radiation integrals are separate divergent. Their sum is finite (for any infra-red safe observable).
A typical subtraction method re-organize the integrals in the form

$$\begin{aligned} \langle O \rangle \quad &= \quad \int d\mathbf{\Phi}_n \, O(\mathbf{\Phi}_n) \left[B(\mathbf{\Phi}_n) + V_b(\mathbf{\Phi}_n) + \int d\Phi_r \, C(\mathbf{\Phi}_n, \Phi_r) \right] \\ &+ \quad \int d\mathbf{\Phi}_n \, d\Phi_r \underbrace{\left[O(\mathbf{\Phi}_n, \Phi_r) \, R(\mathbf{\Phi}_n, \Phi_r) - O(\mathbf{\Phi}_n) \, C(\mathbf{\Phi}_n, \Phi_r) \right]}_{\text{finite}} \end{aligned}$$

Defining

$$V(\mathbf{\Phi}_n) = V_b(\mathbf{\Phi}_n) + \int d\Phi_r \, C(\mathbf{\Phi}_n, \Phi_r) \qquad \Longleftarrow \text{ finite}$$

we have

$$\langle O \rangle = \int d\mathbf{\Phi}_n \, O(\mathbf{\Phi}_n) \left[B(\mathbf{\Phi}_n) + V(\mathbf{\Phi}_n) \right] + \int d\mathbf{\Phi}_n \, d\Phi_r \left[O(\mathbf{\Phi}_n, \Phi_r) \, R(\mathbf{\Phi}_n, \Phi_r) - O(\mathbf{\Phi}_n) \, C(\mathbf{\Phi}_n, \Phi_r) \right]$$

NLO in SMC

Shower Monte Carlo (SMC) cross section for first emission ($d\Phi_r = dt\, dz\, d\varphi$)

$$\langle O \rangle = \int d\mathbf{\Phi}_n\, B(\mathbf{\Phi}_n) \left\{ O(\mathbf{\Phi}_n)\Delta_{t_0} + \int_{t_0} \frac{dt}{t}\, dz\, d\varphi\, O(\mathbf{\Phi}_n, \Phi_r)\, \Delta_t\, \frac{\alpha_s}{2\pi} P(z) \right\}$$

with

$$\Delta_t = \exp\left[-\int_t \frac{dt'}{t'}\, dz'\, d\varphi'\, \frac{\alpha_s}{2\pi} P(z') \right]$$

The expansion at order α_s gives the NLO_{SMC}

$$\langle O \rangle = \int d\mathbf{\Phi}_n\, B(\mathbf{\Phi}_n) \left\{ O(\mathbf{\Phi}_n) + \int_{t_0} \frac{dt}{t}\, dz\, d\varphi\, [O(\mathbf{\Phi}_n, \Phi_r) - O(\mathbf{\Phi}_n)]\, \frac{\alpha_s}{2\pi} P(z) \right\}$$

This is the inexact NLO correction implemented by the SMC

How do we reach exact NLO accuracy?

In the following, a very simplified version of the whole story: no demonstration that we can alter the shower to generate the hardest emission first, truncated shower (see [Nason, hep-ph/0409146] for more details).

Towards NLO accuracy

$$\begin{aligned}\langle O\rangle &= \int d\mathbf{\Phi}_n\, O(\mathbf{\Phi}_n)\,[B(\mathbf{\Phi}_n)+V(\mathbf{\Phi}_n)] \\ &+ \int d\mathbf{\Phi}_n\, d\Phi_r\, [O(\mathbf{\Phi}_n,\Phi_r)\,R(\mathbf{\Phi}_n,\Phi_r) - O(\mathbf{\Phi}_n)\,C(\mathbf{\Phi}_n,\Phi_r)] \\ &= \int d\mathbf{\Phi}_n\, O(\mathbf{\Phi}_n)\left\{B(\mathbf{\Phi}_n)+V(\mathbf{\Phi}_n)+\int d\Phi_r\,[R(\mathbf{\Phi}_n,\Phi_r)-C(\mathbf{\Phi}_n,\Phi_r)]\right\} \\ &+ \int d\mathbf{\Phi}_n\, d\Phi_r\, R(\mathbf{\Phi}_n,\Phi_r)\,[O(\mathbf{\Phi}_n,\Phi_r)-O(\mathbf{\Phi}_n)]\end{aligned}$$

Define

$$\overline{B}(\mathbf{\Phi}_n) = B(\mathbf{\Phi}_n)+V(\mathbf{\Phi}_n)+\int d\Phi_r\,[R(\mathbf{\Phi}_n,\Phi_r)-C(\mathbf{\Phi}_n,\Phi_r)]$$

$$\langle O\rangle = \int d\mathbf{\Phi}_n\, O(\mathbf{\Phi}_n)\,\overline{B}(\mathbf{\Phi}_n) + \int d\mathbf{\Phi}_n\, d\Phi_r\, R(\mathbf{\Phi}_n,\Phi_r)\,[O(\mathbf{\Phi}_n,\Phi_r)-O(\mathbf{\Phi}_n)]$$

In NLO_{SMC}, it was

$$\langle O\rangle = \int d\mathbf{\Phi}_n\, O(\mathbf{\Phi}_n)\,B(\mathbf{\Phi}_n) + \int d\mathbf{\Phi}_n\, d\Phi_r\, B(\mathbf{\Phi}_n)\,\frac{\alpha_s}{2\pi}\,P(z)\,\frac{1}{t}\,[O(\mathbf{\Phi}_n,\Phi_r)-O(\mathbf{\Phi}_n)]$$

POWHEG

$$\text{NLO}_{\text{SMC}} \leftrightarrow \text{NLO}: \qquad B(\mathbf{\Phi}_n) \leftrightarrow \overline{B}(\mathbf{\Phi}_n) \qquad B(\mathbf{\Phi}_n)\frac{\alpha_s}{2\pi}P(z)\frac{1}{t} \leftrightarrow R(\mathbf{\Phi}_n,\Phi_r)$$

All-order emission probability in SMC

$$\langle O\rangle = \int d\mathbf{\Phi}_n\, B(\mathbf{\Phi}_n)\left\{O(\mathbf{\Phi}_n)\,\Delta_{t_0} + \int_{t_0} d\Phi_r\, O(\mathbf{\Phi}_n,\Phi_r)\,\Delta_t\,\frac{\alpha_s}{2\pi}\,P(z)\,\frac{1}{t}\right\}$$

with

$$\Delta_t = \exp\left[-\int d\Phi'_r\,\frac{\alpha_s}{2\pi}\,P(z')\,\frac{1}{t'}\,\theta(t'-t)\right]$$

All order emission probability in POWHEG

$$\langle O\rangle = \int d\mathbf{\Phi}_n\, \overline{B}(\mathbf{\Phi}_n)\left\{O(\mathbf{\Phi}_n)\,\Delta_{t_0} + \int d\Phi_r\, O(\mathbf{\Phi}_n,\Phi_r)\,\Delta_t\,\frac{R(\mathbf{\Phi}_n,\Phi_r)}{B(\mathbf{\Phi}_n)}\right\}$$

$$\Delta_t = \exp\left[-\int d\Phi'_r\,\frac{R(\mathbf{\Phi}_n,\Phi'_r)}{B(\mathbf{\Phi}_n)}\,\theta(t'-t)\right]$$

with $t = k_T(\mathbf{\Phi}_n,\Phi_r)$ and $\overline{B}(\mathbf{\Phi}_n) = B(\mathbf{\Phi}_n) + V(\mathbf{\Phi}_n) + \int d\Phi_r\left[R(\mathbf{\Phi}_n,\Phi_r) - C(\mathbf{\Phi}_n,\Phi_r)\right]$

POSITIVE if $\overline{B}$ is positive (i.e. NLO < LO).

Accuracy of the Sudakov form factor

POWHEG Sudakov form factor has the form (with $c \approx 1$)

$$\Delta_t = \exp\left[-\int_t^{Q^2} \frac{dk_T^2}{k_T^2} \frac{\alpha_s(c\, k_T^2)}{\pi} \left\{ A \log\frac{E^2}{k_T^2} + B \right\}\right]$$

The next-to-leading log (NLL) Sudakov form factor has the form

$$\Delta_t^{\mathrm{NLL}} = \exp\left[-\int_t^{Q^2} \frac{dk_T^2}{k_T^2} \frac{\alpha_s(k_T^2)}{\pi} \left\{ \left(A_1 + A_2 \frac{\alpha_s(k_T^2)}{\pi}\right) \log\frac{E^2}{k_T^2} + B \right\}\right]$$

provided the color structure of the process is sufficiently simple ($\leqslant$ 3 colored legs). Can use this to fix c in POWHEG Sudakov form factor as suggested in Catani, Webber, Marchesini, (1991). HERWIG uses this.
For colored legs $\geqslant 4$, exponentiation only holds at leading-log (LL) or LL + NLL in the large-N_c limit (i.e. planar color structure of Feynman diagrams)

POWHEG Sudakov form factor is always LL accurate. NLL accurate for $\leqslant 3$ colored legs, NLL accurate in leading N_c in all cases.

POWHEG differential cross section

$$d\sigma_{\text{NLO}} = d\mathbf{\Phi}_n \Big\{ B(\mathbf{\Phi}_n) + V(\mathbf{\Phi}_n) + \big[R(\mathbf{\Phi}_n, \Phi_r) - C(\mathbf{\Phi}_n, \Phi_r) \big] \, d\Phi_r \Big\}$$

$$d\mathbf{\Phi}_{n+1} = d\mathbf{\Phi}_n \, d\Phi_r \qquad\qquad d\Phi_r \div dt \, dz \, d\varphi$$

$$V(\mathbf{\Phi}_n) = V_b(\mathbf{\Phi}_n) + \int d\Phi_r \, C(\mathbf{\Phi}_n, \Phi_r) \qquad \Longleftarrow \text{ finite}$$

$$d\sigma_{\text{SMC}} = B(\mathbf{\Phi}_n) \, d\mathbf{\Phi}_n \left\{ \Delta_{t_0} + \frac{\alpha_s}{2\pi} P(z) \frac{1}{t} \Delta_t \, d\Phi_r \right\}$$

$$\Delta_t = \exp\left[- \int d\Phi'_r \, \frac{\alpha_s}{2\pi} P(z') \frac{1}{t'} \theta(t' - t) \right] \qquad \text{SMC Sudakov form factor}$$

$$d\sigma_{\text{POWHEG}} = \overline{B}(\mathbf{\Phi}_n) \, d\mathbf{\Phi}_n \left\{ \Delta\left(\mathbf{\Phi}_n, p_T^{min}\right) + \frac{R(\mathbf{\Phi}_n, \Phi_r)}{B(\mathbf{\Phi}_n)} \Delta(\mathbf{\Phi}_n, p_T) \, d\Phi_r \right\}$$

$$\overline{B}(\mathbf{\Phi}_n) = B(\mathbf{\Phi}_n) + V(\mathbf{\Phi}_n) + \int d\Phi_r \big[R(\mathbf{\Phi}_n, \Phi_r) - C(\mathbf{\Phi}_n, \Phi_r) \big]$$

$$\Delta(\mathbf{\Phi}_n, p_T) = \exp\left[- \int d\Phi'_r \, \frac{R(\mathbf{\Phi}_n, \Phi'_r)}{B(\mathbf{\Phi}_n)} \, \theta\left(k_T(\mathbf{\Phi}_n, \Phi'_r) - p_T \right) \right] \qquad \text{POWHEG Sudakov}$$

POWHEG is even more flexible

We have great flexibility to deal with the real contribution

$$d\sigma = \overline{B}(\mathbf{\Phi}_n)\left\{\Delta\left(p_T^{min}\right)+\Delta(p_T)\,\frac{R(\mathbf{\Phi}_{n+1})}{B(\mathbf{\Phi}_n)}\,d\Phi_r\right\}d\mathbf{\Phi}_n$$

$$\overline{B}(\mathbf{\Phi}_n) = B(\mathbf{\Phi}_n)+V(\mathbf{\Phi}_n)+\int d\Phi_r\left[R(\mathbf{\Phi}_n,\Phi_r)-C(\mathbf{\Phi}_n,\Phi_r)\right]$$

$$\Delta(p_T) = \exp\left[-\int d\Phi_r'\,\frac{R(\mathbf{\Phi}_n,\Phi_r')}{B(\mathbf{\Phi}_n)}\,\theta\left({p_T}'-p_T\right)\right]$$

Break $R = R_s + R_f$ with $R_s > 0$, $R_f > 0$, R_s **s**ingular in the infrared regions, R_f **f**inite in collinear and soft limit. Define

$$d\sigma' = \overline{B}_s(\mathbf{\Phi}_n)\left\{\Delta_s\left(p_T^{min}\right)+\Delta_s(p_T)\,\frac{R_s(\mathbf{\Phi}_{n+1})}{B(\mathbf{\Phi}_n)}\,d\Phi_r\right\}d\mathbf{\Phi}_n + R_f(\mathbf{\Phi}_{n+1})\,d\mathbf{\Phi}_{n+1}$$

$$\overline{B}_s(\mathbf{\Phi}_n) = B(\mathbf{\Phi}_n)+V(\mathbf{\Phi}_n)+\int d\Phi_r\left[R_s(\mathbf{\Phi}_n,\Phi_r)-C(\mathbf{\Phi}_n,\Phi_r)\right]$$

$$\Delta_s(p_T) = \exp\left[-\int d\Phi_r'\,\frac{R_s(\mathbf{\Phi}_n,\Phi_r')}{B(\mathbf{\Phi}_n)}\,\theta\left({p_T}'-p_T\right)\right]$$

Easy to prove that $d\sigma'$ is equivalent to $d\sigma$. In other words, the part of the real cross section that is treated with the shower technique can be varied.

MC@NLO in the POWHEG language

Write the MC@NLO hardest jet cross section in the POWHEG language. Hardest emission can be written as [Nason 2004]

$$d\sigma = \underbrace{\overline{B}_{\rm HW}\, d\mathbf{\Phi}_n}_{\mathbb{S}\ \text{event}} \underbrace{\left[\Delta_{\rm HW}(p_T^{min}) + \Delta_{\rm HW}(p_T)\frac{R_{\rm HW}(\mathbf{\Phi}_{n+1})}{B(\mathbf{\Phi}_n)}\, d\Phi_r\right]}_{\text{HERWIG event}} + \underbrace{\left[R(\mathbf{\Phi}_{n+1}) - R_{\rm HW}(\mathbf{\Phi}_{n+1})\right] d\mathbf{\Phi}_{n+1}}_{\mathbb{H}\ \text{event}}$$

$$\overline{B}_{\rm HW}(\mathbf{\Phi}_n) = B(\mathbf{\Phi}_n) + V(\mathbf{\Phi}_n) + \int \left[R_{\rm HW}(\mathbf{\Phi}_n, \Phi_r) - C(\mathbf{\Phi}_n, \Phi_r)\right] d\Phi_r$$

$$\Delta_{\rm HW}(p_T) = \exp\left[-\int d\Phi'_r \frac{R_{\rm HW}(\mathbf{\Phi}_n, \Phi'_r)}{B(\mathbf{\Phi}_n)}\, \theta({p_T}' - p_T)\right]$$

Like POWHEG with

$$\begin{cases} R_s = R_{\rm HW} & \\ R_f = R - R_{\rm HW} & \Longleftarrow \ \text{can be negative} \end{cases}$$

This formula illustrates why MC@NLO and POWHEG are equivalent at NLO!
But differences can arise at NNLO. More on this later.

In summary

$$d\sigma = \overline{B}_s(\mathbf{\Phi}_n)\left\{\Delta_s\left(p_T^{min}\right) + \Delta_s(p_T)\,\frac{R_s(\mathbf{\Phi}_{n+1})}{B(\mathbf{\Phi}_n)}\,d\Phi_r\right\}d\mathbf{\Phi}_n + R_f(\mathbf{\Phi}_{n+1})\,d\mathbf{\Phi}_{n+1}$$

$$\overline{B}_s(\mathbf{\Phi}_n) = B(\mathbf{\Phi}_n) + V(\mathbf{\Phi}_n) + \int d\Phi_r\left[R_s(\mathbf{\Phi}_n, \Phi_r) - C(\mathbf{\Phi}_n, \Phi_r)\right]$$

$$\Delta_s(p_T) = \exp\left[-\int d\Phi_r'\,\frac{R_s(\mathbf{\Phi}_n, \Phi_r')}{B(\mathbf{\Phi}_n)}\,\theta\left({p_T}' - p_T\right)\right]$$

1. First, according to the POWHEG method, one generates an underlying Born configuration, i.e. the kinematics $\mathbf{\Phi}_n$ is generated with probability distribution according to the $\overline{B}_s(\mathbf{\Phi}_n)$ function and the flavour of the underlying Born configuration is chosen according to its contribution to the integral of $\overline{B}_s(\mathbf{\Phi}_n)$ over the whole Born phase space
2. Then the radiation Φ_r is generated distributed according to $\Delta_s \times R_s/B$. Together with the underlying Born kinematics $\mathbf{\Phi}_n$, the kinematics of the real-emission event $\mathbf{\Phi}_{n+1}$ is then completely determined.
3. If needed, generate the kinematics according to the finite contribution R_f. Since this is finite and positive, no problem in the generation of $\mathbf{\Phi}_{n+1}$ for this kind of contributions. N.B. The R_f term is necessary when the real-emission term has not an underlying Born. This is the case for example of Higgs boson production in gluon fusion, $gg\to H$, where the $q\bar{q}\to Hg$ real diagrams cannot be built from an underlying Born term

Mathematical tricks

✓ To generate the underlying Born kinematics ($\boldsymbol{\Phi}_n$), distributed according to $\overline{B}_s(\boldsymbol{\Phi}_n)$, one uses programs like BASES/SPRING or MINT, that, after a single integration, can generate points distributed according to the integrand function.

✓ Use the veto technique and the highest-p_T bid procedure, to generate the radiation variables, distributed according to $d\Delta_s(p_T)$.

These tricks are well known to Monte Carlo experts.

We have collected a few of them in the appendixes of our paper [Frixione, Nason and Oleari, arXiv:0709.2092 [hep-ph]].

POWHEG / POWHEG BOX

- ✓ it can generate events with positive weights. NO negative weights to handle
- ✓ it is independent from parton-shower programs. Can be interfaced with PYTHIA, HERWIG, SHERPA...
  It is then possible to compare the different outputs
- ✓ No need to implement new interfaces
  Two possible ways to interface to shower Monte Carlo programs
  1. Les Houches Event format. The event is written on a file that is subsequently showered by HERWIG, PYTHIA...
  2. on the fly. We provide UPINIT and UPEVNT directly running in HERWIG and PYTHIA
- ✓ As far as the hardest emission is concerned, POWHEG guarantees:
  - NLO accuracy on integrated quantities
  - collinear, double-log (soft-collinear), large-N_c-soft single-log of the Sudakov (in fact, corrections that exponentiates are obviously OK)
- ✓ As far as subsequent (less hard) emissions, the output has the accuracy of the SMC one is using.

A few questions

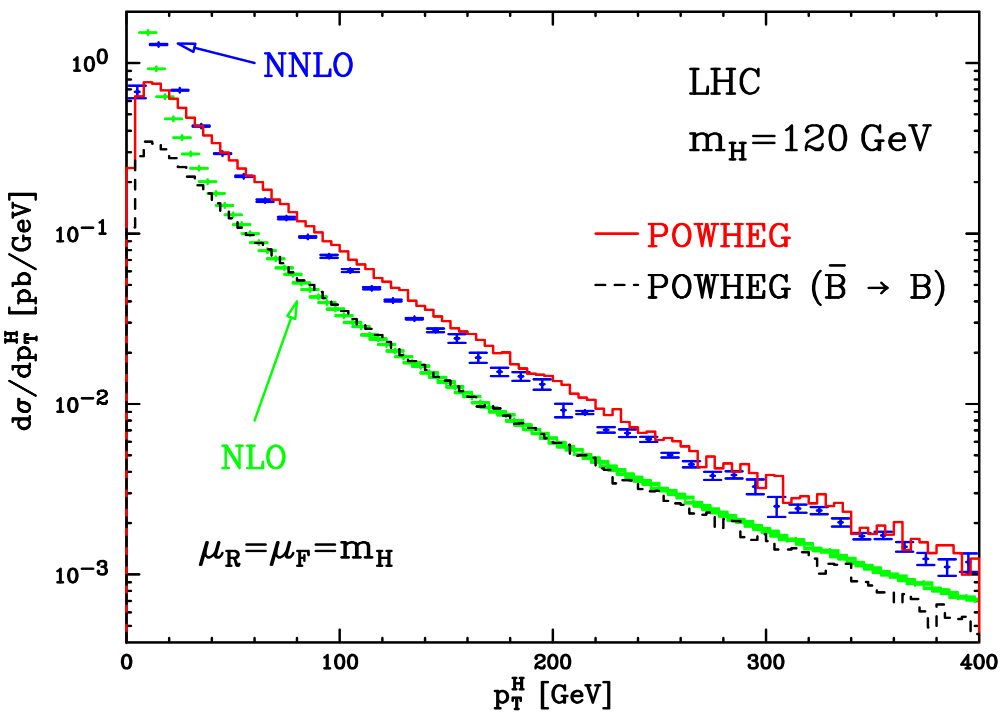
- Can we estimate the size of NNLO corrections, at least in the high p_T tail?

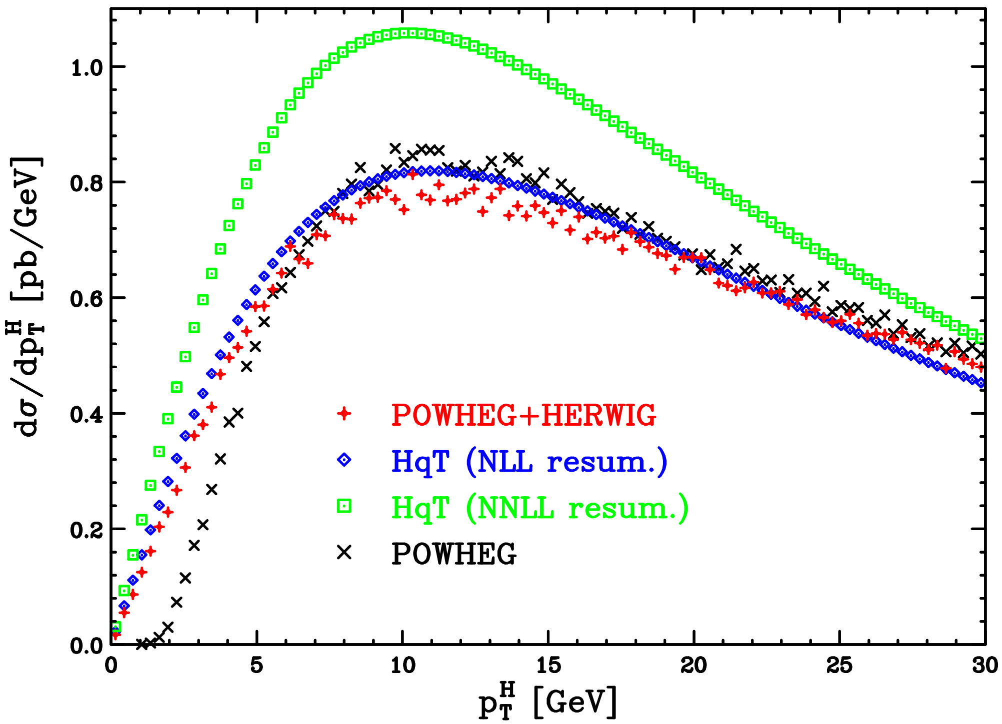
- What happens if the Born term B is zero in some kinematic configurations?
  This happens, for example, for Drell-Yan hadroproduction $pp \to W \to l\nu$: there is a zero in the Born term if the outgoing lepton is anti-parallel to the incoming quark (due to the left-handed nature of the W boson coupling, we have a violation of angular-momentum conservation along the incoming beam)

- How can we compute the renormalization and factorization scale dependence of the POWHEG result?

- What happens if the Born term B is divergent?
  This happens, for example, for $pp \to$ jets, V+jet production...

NNLO contributions: Higgs boson production





$$\overline{B}_s(\mathbf{\Phi}_n) = B(\mathbf{\Phi}_n) + V(\mathbf{\Phi}_n) + \int d\Phi_r \left[R_s(\mathbf{\Phi}_n, \Phi_r) - C(\mathbf{\Phi}_n, \Phi_r)\right]$$

$$d\sigma = \overline{B}_s(\mathbf{\Phi}_n) \left\{ \Delta_s(\mathbf{\Phi}_n, p_T^{min}) + \Delta_s(\mathbf{\Phi}_n, p_T) \frac{R_s(\mathbf{\Phi}_{n+1})}{B(\mathbf{\Phi}_n)}\, d\Phi_r \right\} d\mathbf{\Phi}_n + R_f(\mathbf{\Phi}_{n+1})\, d\mathbf{\Phi}_{n+1}$$

$$d\sigma_{\rm rad} \approx \frac{\overline{B}_s(\mathbf{\Phi}_n)}{B(\mathbf{\Phi}_n)} R_s(\mathbf{\Phi}_{n+1})\, d\mathbf{\Phi}_{n+1} + R_f(\mathbf{\Phi}_{n+1})\, d\mathbf{\Phi}_{n+1}$$

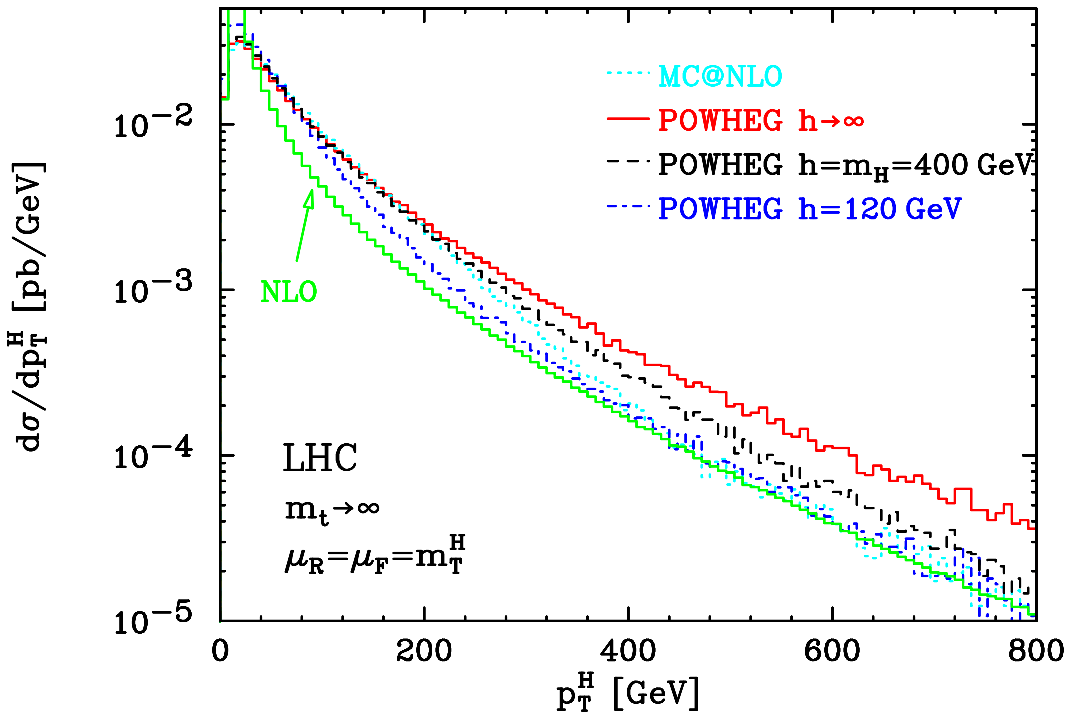
$$= \left\{ \left[1 + \mathcal{O}(\alpha_s)\right] R_s(\mathbf{\Phi}_{n+1}) + R_f(\mathbf{\Phi}_{n+1}) \right\} d\mathbf{\Phi}_{n+1} = R(\mathbf{\Phi}_{n+1})\, d\mathbf{\Phi}_{n+1} + \mathcal{O}(\alpha_s) R_s(\mathbf{\Phi}_{n+1})$$

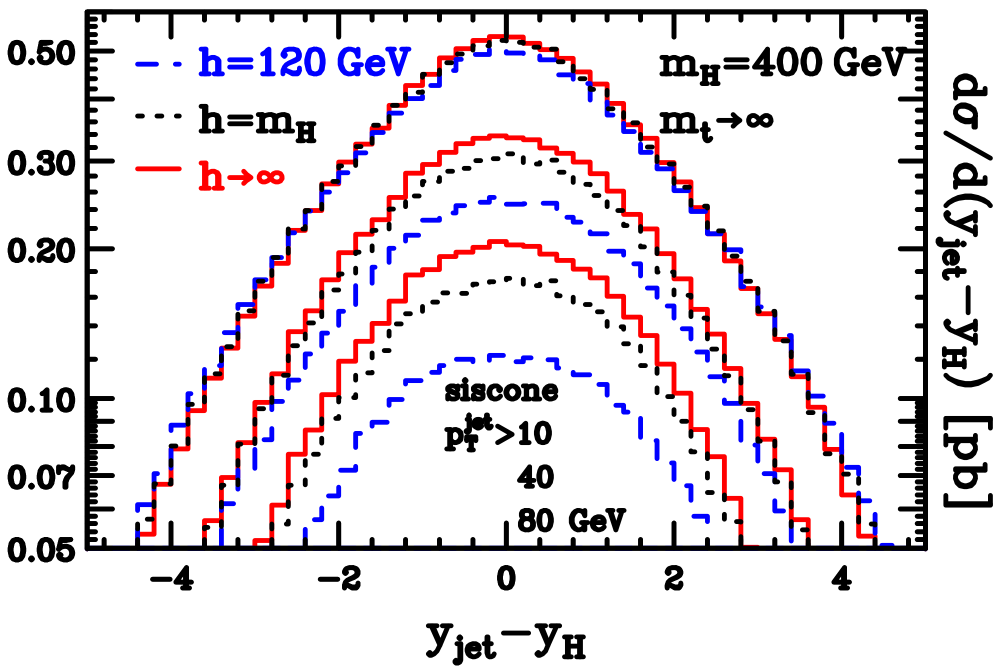
$$R_s = \frac{h^2}{p_T^2 + h^2} R$$

$$R_f = \frac{p_T^2}{p_T^2 + h^2} R$$

$$R = R_s + R_f$$

agrees with NLO at high p_T

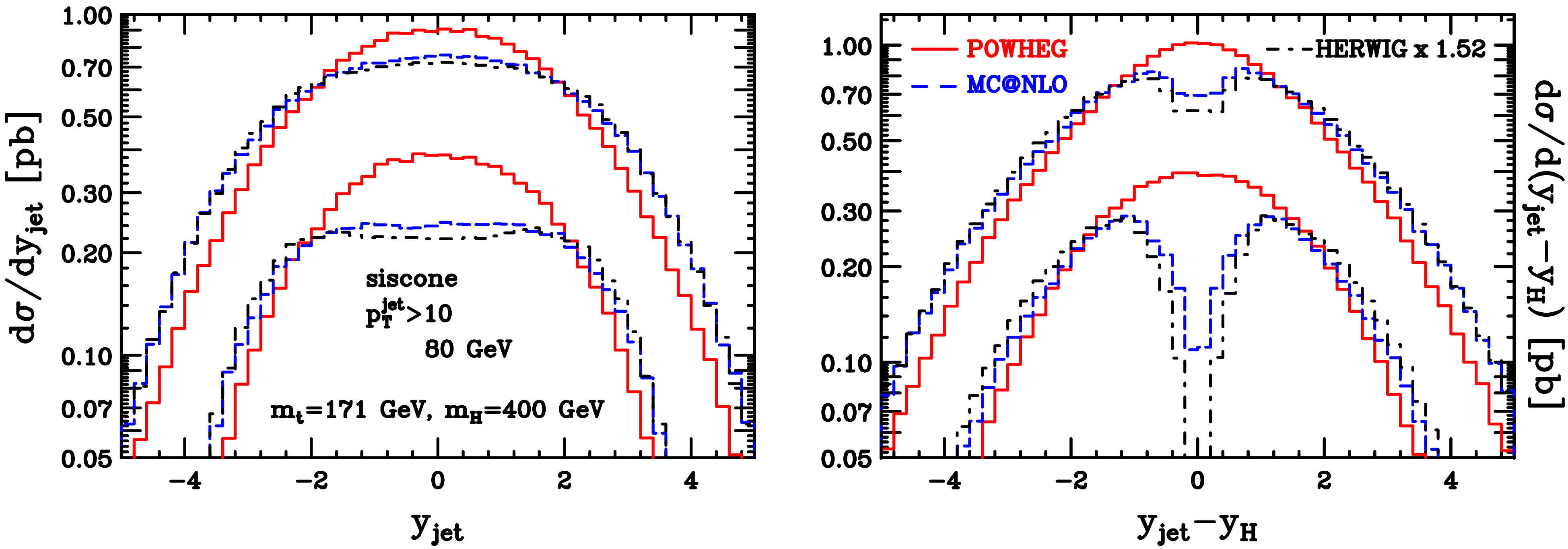




No new features appear in all the other distributions

When $h \to 0$, we recover the pure NLO cross section

NNLO contributions: the dip in MC@NLO



- Dip inherited from the deeper dip of HERWIG. MC@NLO fills partially the dip.
- It gets worse for large $p_T^{\rm jet}$
- Why MC@NLO has a dip in the hardest jet rapidity?
- Why POWHEG has no dip? Is that because of the hardest p_T spectrum?

NNLO contributions: the dip in MC@NLO

Write the MC@NLO hardest jet cross section in the POWHEG language. Hardest emission can be written as [Nason 2004]

$$d\sigma = \underbrace{\overline{B}_{\rm HW}\, d\boldsymbol{\Phi}_n}_{\mathbb{S}\text{ event}} \underbrace{\left[\Delta_{\rm HW}(p_T^{min}) + \Delta_{\rm HW}(p_T)\frac{R_{\rm HW}(\boldsymbol{\Phi}_{n+1})}{B(\boldsymbol{\Phi}_n)}\, d\Phi_r\right]}_{\text{HERWIG event}} + \underbrace{\left[R(\boldsymbol{\Phi}_{n+1}) - R_{\rm HW}(\boldsymbol{\Phi}_{n+1})\right] d\boldsymbol{\Phi}_{n+1}}_{\mathbb{H}\text{ event}}$$

$$\overline{B}_{\rm HW}(\boldsymbol{\Phi}_n) = B(\boldsymbol{\Phi}_n) + V(\boldsymbol{\Phi}_n) + \int \left[R_{\rm HW}(\boldsymbol{\Phi}_n, \Phi_r) - C(\boldsymbol{\Phi}_n, \Phi_r)\right] d\Phi_r$$

$$\Delta_{\rm HW}(p_T) = \exp\left[-\int d\Phi_r' \frac{R_{\rm HW}(\boldsymbol{\Phi}_n, \Phi_r')}{B(\boldsymbol{\Phi}_n)}\, \theta(p_T{}' - p_T)\right]$$

Like POWHEG with
$$\begin{cases} R_s = R_{\rm HW} \\ R_f = R - R_{\rm HW} \quad \Longleftarrow \quad \text{can be negative} \end{cases}$$

This formula illustrates why MC@NLO and POWHEG are equivalent at NLO!
But differences can arise at NNLO…

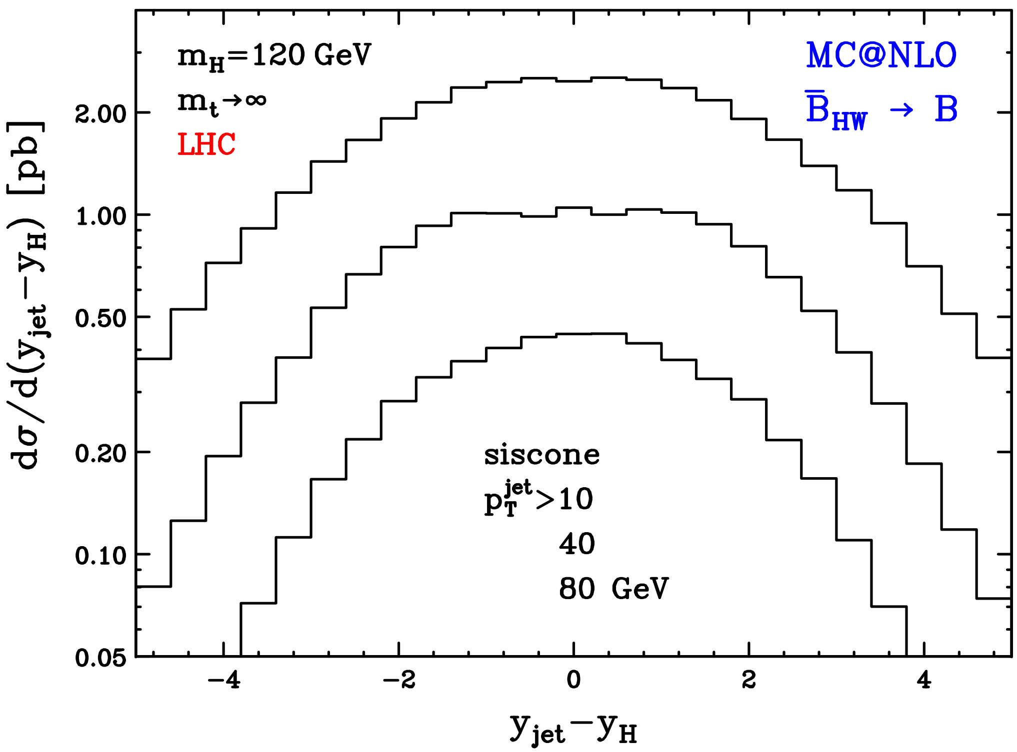
At high p_T the cross section goes as

$$d\sigma \approx \left[\frac{\overline{B}_{\rm HW}(\mathbf{\Phi}_n)}{B(\mathbf{\Phi}_n)} R_{\rm HW}(\mathbf{\Phi}_{n+1}) + R(\mathbf{\Phi}_{n+1}) - R_{\rm HW}(\mathbf{\Phi}_{n+1})\right] d\mathbf{\Phi}_{n+1}$$

$$= \underbrace{R(\mathbf{\Phi}_{n+1})}_{\text{no dip}} d\mathbf{\Phi}_{n+1} + \underbrace{\left(\frac{\overline{B}_{\rm HW}(\mathbf{\Phi}_n)}{B(\mathbf{\Phi}_n)} - 1\right)}_{\mathcal{O}(\alpha_s) \text{ but large for Higgs}} \underbrace{R_{\rm HW}(\mathbf{\Phi}_{n+1})}_{\text{pure HERWIG dip}} d\mathbf{\Phi}_{n+1}$$

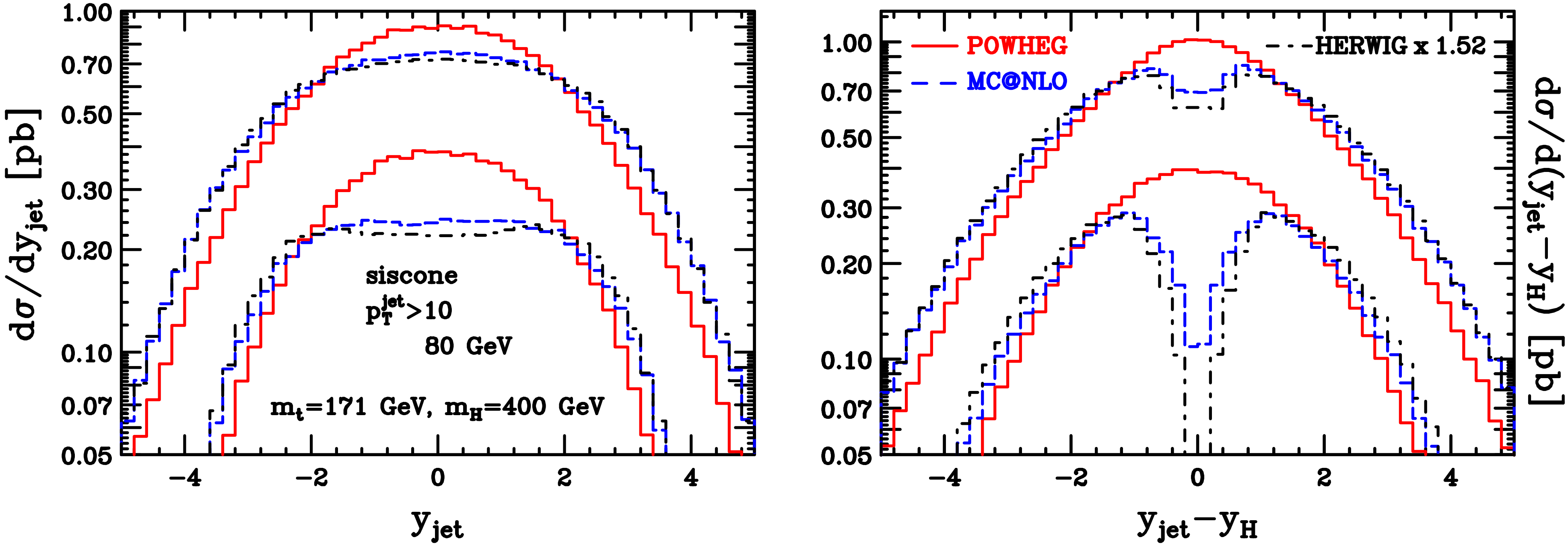
So: a contribution with a dip is added to the exact NLO result.
The contribution is $\mathcal{O}(\alpha_s R)$, i.e. NNLO

Can we test this hypothesis?
Replace $\overline{B}_{\rm HW} \to B$ in MC@NLO.
The dip should disappear...



No visible dip is present.

NNLO contributions: the dip in MC@NLO



- Why MC@NLO has a dip in the hardest jet rapidity?
  ANSWER: because it is very sensitive to the dead zone in the HERWIG phase space
- Why POWHEG has no dip? Is that because of the hardest p_T spectrum?
  ANSWER: NO, it does not depend on the hardest p_T spectrum. POWHEG generate by itself the hardest radiation.

Summary of MC@NLO and POWHEG comparisons

- Fairly good agreement on most distributions
- Areas of disagreement can be tracked back to NNLO terms, arising mostly because of the use of an NLO inclusive cross section (the $\bar{B}$ function) to shower out the hardest radiation.
- In POWHEG, since the hardest radiation is generated by POWHEG itself, one has high flexibility in tuning the magnitude of these NNLO terms.
- For MC@NLO, these NNLO terms can generate unphysical behavior in physical distributions, reflecting the dead zones structure of the underlying shower Monte Carlo.

Since MC@NLO uses the underlying Monte Carlo to generate the hardest emission, to remedy to these problems one has to intervene on the Monte Carlo itself

Born zeros

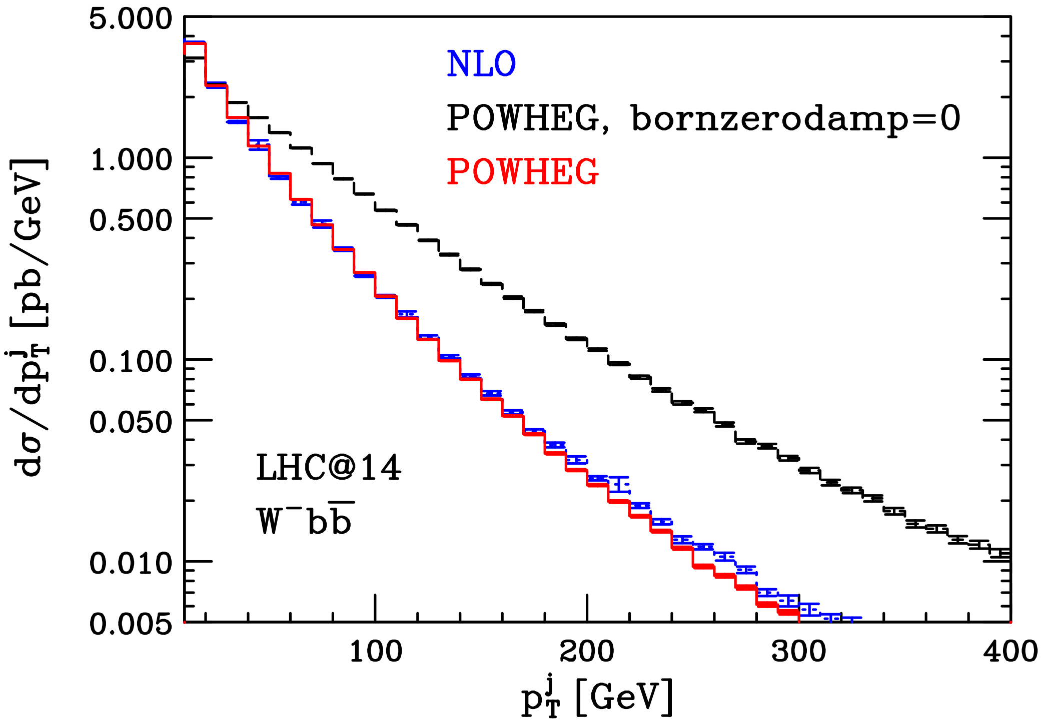
- Born kinematics configurations with a vanishing Born may be generated if the $\bar{B}$ term is different from zero.

- At the stage of radiation generation, one would find very large ratios of $R/B \Longrightarrow$ difficult to find a reasonable upper bound for this ratio.

- In the limit of hardness (p_T) of the radiation going to zero, R too approaches 0 (soft and collinear limit). The problem arises when the distance of the underlying Born configuration from the zero configuration is smaller than the distance of the real-emission cross section from the singular (i.e. zero hardness) configuration

Born zeros

The POWHEG BOX has a built-in mechanism to deal with Born terms that can become zero in some kinematic points of the phase space. This mechanism is activated by the `bornzerodamp` flag set to 1 in the input file



Enhancement of the high-p_T tail by a factor $\bar{B}/B$ ($\bar{B}$ different from 0, $B \to 0$)

$$\text{if} \begin{cases} \dfrac{R}{R_{\rm coll}} > N \\ \dfrac{R}{R_{\rm soft}} > N \end{cases} \quad N \approx 5 \;\; \text{then} \begin{cases} R_s = 0 \\ R_f = R \end{cases}$$

R is far from the collinear and soft regions $\Longrightarrow$ it is finite and can be safely treated as separate from the shower, in the R_f term

Scale dependence

$$d\sigma = \overline{B}_s(\mathbf{\Phi}_n, \mu_R)\, d\mathbf{\Phi}_n \left\{ \Delta_s\left(\mathbf{\Phi}_n, p_T^{min}\right) + \Delta_s(\mathbf{\Phi}_n, p_T)\, \frac{R_s(\mathbf{\Phi}_n, \Phi_r, \alpha_s(k_T))}{B(\mathbf{\Phi}_n)}\, d\Phi_r \right\}$$

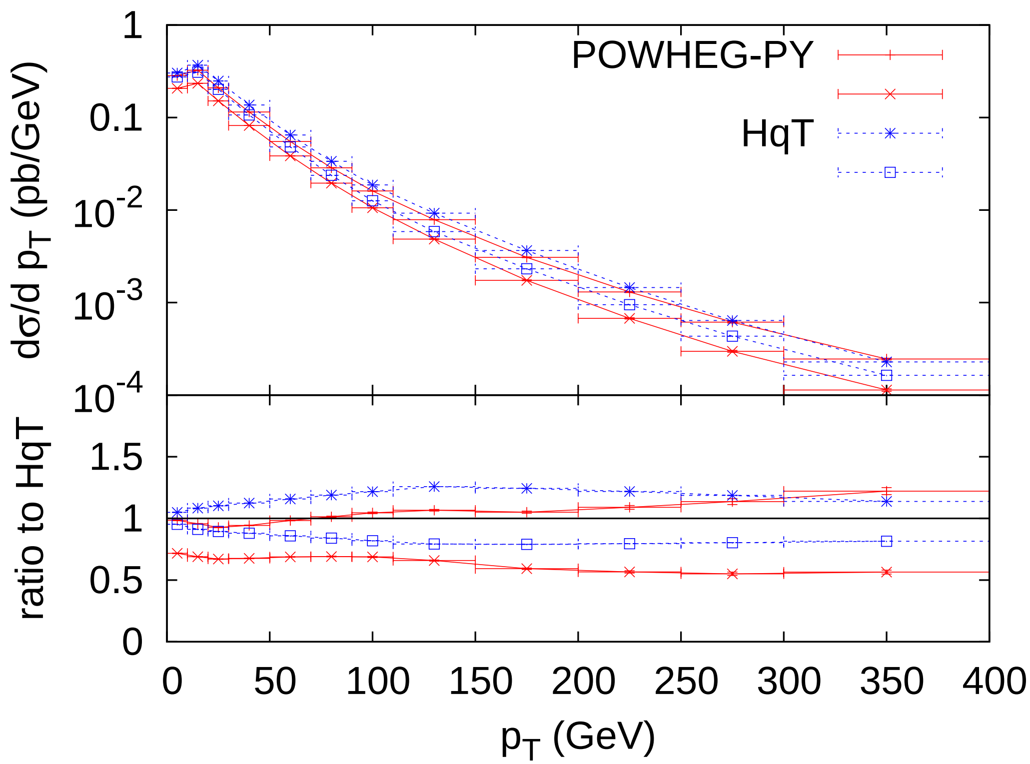
$$+ R_f(\mathbf{\Phi}_{n+1}, \alpha_s(\mu_R))\; d\mathbf{\Phi}_{n+1}$$

$$\overline{B}_s(\mathbf{\Phi}_n, \mu_R) = B(\mathbf{\Phi}_n) + V(\mathbf{\Phi}_n, \alpha_s(\mu_R)) + \int d\Phi_r \left[R_s(\mathbf{\Phi}_n, \Phi_r, \alpha_s(\mu_R)) - C(\mathbf{\Phi}_n, \Phi_r, \alpha_s(\mu_R))\right]$$

$$\Delta_s(\mathbf{\Phi}_n, p_T) \exp\left[-\int d\Phi'_r\, \frac{R_s(\mathbf{\Phi}_n, \Phi'_r, \alpha_s(k_T))}{B(\mathbf{\Phi}_n)}\, \theta\left(k_T(\mathbf{\Phi}_n, \Phi'_r) - p_T\right)\right]$$

- A scale variation in the curly braces {} is in practice never performed (in order not to spoil the NLL accuracy of the Sudakov form factor)
- Scale dependence affects $\overline{B}_s$ and R_f differently: $\overline{B}_s$ is a quantity integrated over the radiation kinematics $\Longrightarrow$ milder scale dependence

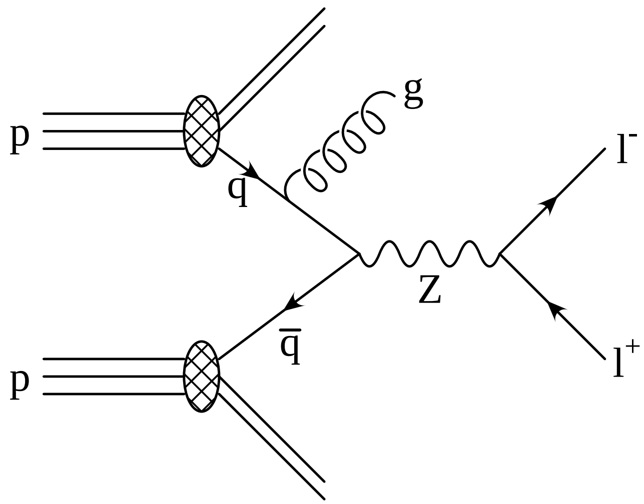
Similar conclusions for the factorization scale μ_F



- $gg \to H$ at NLO
- HqT Catani, Grazzini et al.: NNLL+NNLO the "switched" result, with resummation scale $Q = m_H$
- $0.5 < \mu_R/\mu_F < 2$ around central value m_H
- R_f/R_s separation done automatically by the POWHEG BOX, on an event-by-event basis.

- The error band of POWHEG is relatively small at small p_T and becomes larger at larger p_T. The p_T of the Higgs boson is a LO quantity. $H+1$ jet starts at order α_s^3. Its scale variation is of order $\alpha_s^4 \Longrightarrow$ its relative scale variation is of order $\alpha_s^4/\alpha_s^3 \propto \alpha_s$
- On the other hand, the total cross section (the integral of the curve) or the Higgs boson rapidity distribution, that are obtained by integrating over all transverse momenta, are given by a term of order α_s^2 plus a term of order α_s^3, and their scale variation is also of order α_s^4. Thus, their relative scale variation is of order $\alpha_s^4/\alpha_s^2 \propto \alpha_s^2$

Divergent Born



$Z + 1$ jet, dijet production... were the first cases we had to face with a divergent Born.

POWHEG starts from a Born diagram and attaches radiation.

First solution: introduce a cutoff, i.e. generate events starting from partonic Born events with $p_T^{\rm B} > p_T^{\rm gen}$, called generation cut

- Study the effect of the cutoff at the partonic Born level on showered events

- Check that there is no sensitivity to the cut after the analysis of the hadronic events. If $p_T^{\rm an}$ is the analysis cut, taking $p_T^{\rm an} \gtrsim p_T^{\rm gen}$ is not enough to get a realistic sample. In fact, in an event generated at the Born level with a given $p_T^{\rm B} < p_T^{\rm gen}$, the shower may increase the transverse momentum of the jet so that the final transverse momentum p_T can be bigger than $p_T^{\rm an}$.

Divergent Born

Second solution: generate weighted events, rather than unweighted ones. Generate the underlying Born kinematics not according to $\bar{B}$ but according to

$$\bar{B} \longrightarrow \bar{B} \times F(p_T^{\mathrm{B}})$$

where $F(p_T^{\mathrm{B}})$ is a suppression function such that

$$\lim_{p_T^{\mathrm{B}} \to 0} F(p_T^{\mathrm{B}}) = 0 \qquad \text{and} \qquad \lim_{p_T^{\mathrm{B}} \to 0} \bar{B} \times F(p_T^{\mathrm{B}}) = \text{finite}$$

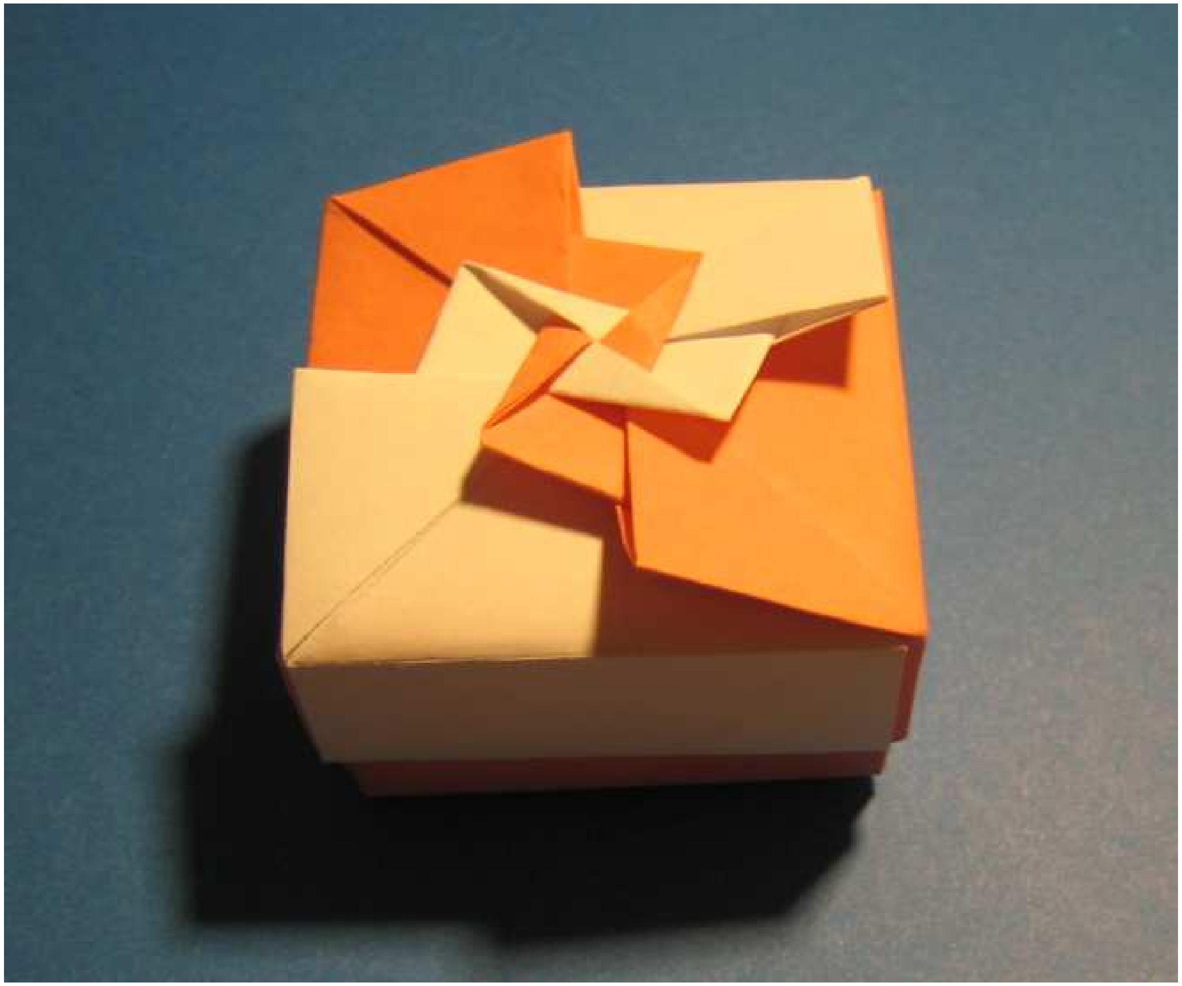
The generated events, however, should be given a weight $1/F(p_T^{\mathrm{B}})$, rather than 1, in order to compensate for the initial $F(p_T^{\mathrm{B}})$ suppression factor.

Example

$$F(p_T^{\mathrm{B}}) = \frac{\left[\left(p_T^{\mathrm{B}}\right)^2\right]^{\alpha}}{\left(p_T^{\mathrm{B}}\right)^2 + \left(p_T^{\mathrm{supp}}\right)^2}$$

p_T^{supp} some numerical value and α such that $\bar{B} \times F(p_T^{\mathrm{B}})$ finite in the small transverse-momentum region

The POWHEG BOX



http://powhegbox.mib.infn.it

The POWHEG BOX

The POWHEG BOX is a public-available computer framework, presented in [Alioli, Nason, Oleari and Re, arXiv:1002.2581], that implements in practice the theoretical construction of the POWHEG formalism, for generic NLO processes, according to the general formulation of POWHEG given in [Frixione, Nason and Oleari, arXiv:0709.2092]

More precisely, the user should only supply:

- ✓ the lists of the Born and real processes (i.e. $sc \rightarrow gud \Longleftrightarrow [3,4,0,2,1]$)
- ✓ the Born phase space
- ✓ the Born squared amplitudes, the color-correlated and spin-correlated amplitudes, for all partonic subprocesses
  All these amplitudes are common ingredients of a NLO calculation
- ✓ the real squared amplitude for all the relevant real-emission subprocesses
- ✓ the finite part of the virtual corrections, computed in conventional dimensional regularization or in dimensional reduction
- ✓ the Born color structures in the limit of large number of colors.

All the rest will be done **automatically**!

The POWHEG BOX

The user should not worry about

- ✓ the phase space for initial-state radiation and final-state radiation (i.e. the phase space for real emission)
- ✓ the combinatorics, the identification of all singular regions in the real amplitude R, the soft and collinear limits, the calculation of all the counterterms
- ✓ the calculation of the differential NLO cross section
  Spinoff: NLO results using the FKS subtraction scheme
- ✓ the calculation of the upper bounds for the generation of radiation (for an efficient generation of the Sudakov-suppressed events)
- ✓ the generation of radiation
- ✓ writing the event into the Les Houches interface (to communicate with the LO Shower Monte Carlo programs)

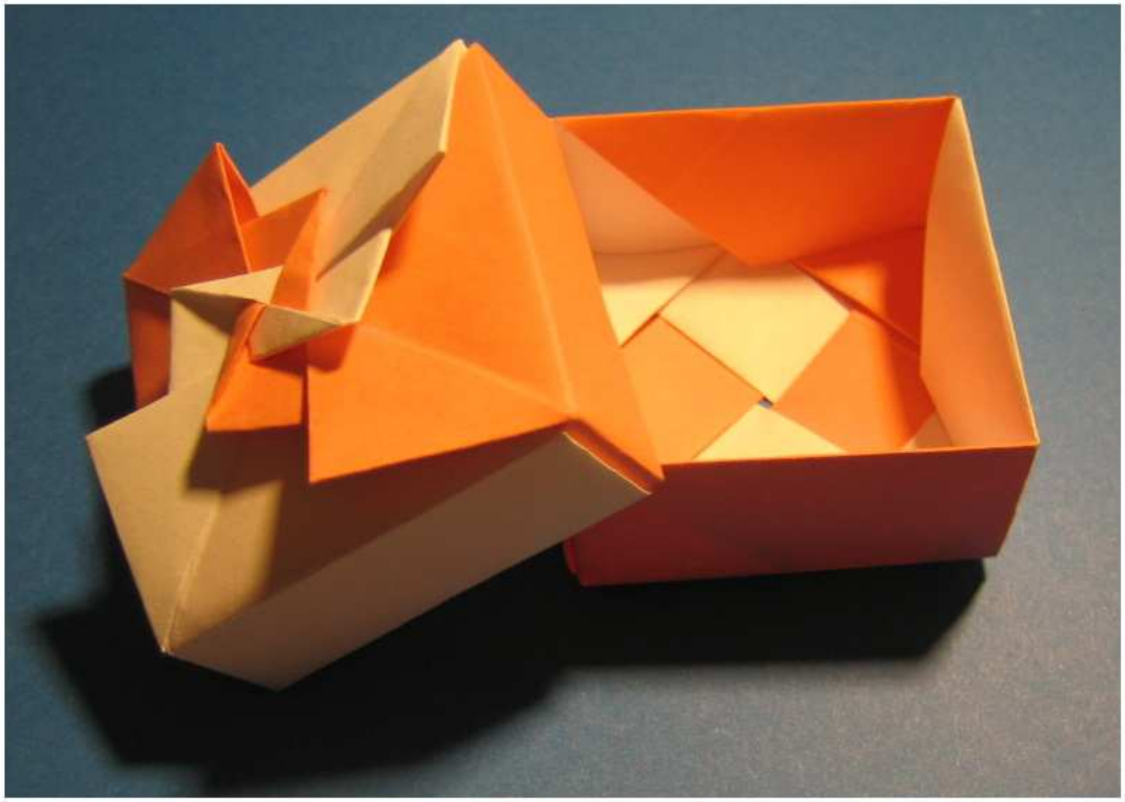
The user has only to know in which format to supply the ingredients listed before.

Recent improvements

- In collaboration with Rikkert Frederix, we have built an interface to MadGraph 4 that automatically builds the Born, Born color- and spin-correlated amplitudes, the real amplitude and the Born color structure in the large number of colors. Using this interface, the only missing ingredients are
  - the Born phase space
  - the virtual term
- Towards the automatization of the calculation of the virtual
  - MCFM [Williams, Campbell, Ellis]: build an interface to existing MCFM processes.
  - GoSam [Cullen, Greiner, Heinrich, Luisoni, Mastrolia, Ossola, Reiter, Tramontano]: interface this automatic generator of virtual contributions to the POWHEG BOX

After this, the only missing ingredient for a fully automated generator will be the Born phase space.

The POWHEG BOX



No need to open the BOX!

The POWHEG BOX

Use the FKS (Frixione-Kunszt-Signer) subtraction scheme according to the general formulation of POWHEG given in [Frixione, Nason and Oleari, 2007] (FNO), hiding all FKS implementation details.

In other words, the user needs not to know it!

It includes:

- ✓ the phase space for ISR and FSR, according to FNO.
- ✓ the combinatorics, the calculation of all singular regions in the real amplitude R, the soft and collinear limit
- ✓ the calculation of $\bar{B}$ (spinoff: NLO results using the FKS subtraction scheme)
- ✓ the calculation of the upper bounds for the generation of radiation
- ✓ the generation of radiation
- ✓ writing the event into the Les Houches interface

The POWHEG BOX How-To

- `parameter (nlegborn=5)` $[pp \to (Z \to e^+ e^-) + j]$ in included file `pwhg_flst.h`
  `flst_nborn` and `flst_nreal`
- `flst_born(k=1..nlegborn,j=1..flst_nborn)`: flavour of the k-th leg of the j-th Born graph
  `flst_real(k=1..nlegreal,j=1..flst_nreal)`: flavour of the k-th leg of the j-th real graph.
  It is required that legs in the Born and real processes have to be ordered as follows:
  - leg 1, incoming parton with positive rapidity
  - leg 2, incoming parton with negative rapidity
  - from leg 3 onward, final state particles, in the order: colorless particles first, massive coloured particles, massless coloured particles.

  The flavour is taken incoming for the two incoming particles and outgoing for the outgoing particles. The flavour index is assigned according to PDG conventions, except for gluons, where 0 is used instead of 21.
  Example: $pp \to (Z \to e^+ e^-) + 2j$, the string `[1,0,-11,11,1,0]` labels the process $dg \to e^+ e^- dg$
- `init_couplings`

The POWHEG BOX: example

Suppose that we are interested in $pp \to e^- e^+$ and that only the u quark and the gluon exist.

$u\bar{u} \to e^- e^+$	`flst_born(...,1)` = [2,-2,11,-11]
$\bar{u}u \to e^- e^+$	`flst_born(...,2)` = [-2, 2, 11, -11]

`nlegborn`=4

`flst_nborn` = 2

$u\bar{u} \to e^- e^+ g$	`flst_real(...,1)` = [2, -2, 11,-11,0]
$\bar{u}u \to e^- e^+ g$	`flst_real(...,2)` = [-2, 2, 11,-11, 0]
$g\bar{u} \to e^- e^+ \bar{u}$	`flst_real(...,3)` = [0,-2, 11,-11, -2]
$gu \to e^- e^+ u$	`flst_real(...,4)` = [0, 2, 11,-11, 2]
$\bar{u}g \to e^- e^+ \bar{u}$	`flst_real(...,5)` = [-2, 0, 11,-11, -2]
$ug \to e^- e^+ u$	`flst_real(...,6)` = [2, 0, 11,-11, 2]

`nlegreal` = `nlegborn + 1`

`flst_nreal` = 6

- `Born_phsp(xborn)` for Born phase space
  `xborn(1..ndim)` array of random numbers `ndim=(nlegborn-2)*3-4+2-1`
  - the Born Jacobian `kn_jacborn`, Born momenta in the laboratory frame `kn_pborn(0:3,1..nlegborn)`, Born momenta in the partonic CM frame `kn_cmpborn(0:3,1..nlegborn)` and Bjorken x (`kn_xb1` and `kn_xb2`).
- `set_ren_fac_scales(mur,muf)`
- `setborn(p,bflav,born,bornjk,bmunu)`
  - the momenta `p(0:3,1..nlegborn)`
  - the flavour string `bflav(1..nlegborn)`
  - `bornjk(1..nlegborn,1..nlegborn)`
  - the Born helicity-correlated squared amplitudes `bmunu(0:3,0:3,j=1..nlegborn)`
- `setvirtual(p,vflav,virtual)` returns finite part of the interference $2\,\mathrm{Re}\,(M_B \times M_V)$, after factorizing out ($d = 4 - 2\epsilon$)

$$\mathcal{N} = \frac{(4\pi)^\epsilon}{\Gamma(1-\epsilon)} \left(\frac{\mu^2}{Q^2}\right)^\epsilon \frac{\alpha_s}{2\pi}$$

- `real_ampsq(p,rflav,amp2)`
  - the momenta `p(0:3,1..nlegreal)`
  - the flavour string `rflav(1...nlegreal)`
  - `amp2`: spin and color summed and averaged real squared amplitudes

Processes implemented in the POWHEG BOX

- heavy-quark pair production (Frixione, Nason, Ridolfi, 2007)
- Z/W (with decay) (Alioli, Nason, Re, C.O., 2008)
- Higgs boson in gluon fusion (Alioli, Nason, Re, C.O., 2008)
- single top (Alioli, Nason, Re, C.O., 2009) and tW (Re, 2010)
- Higgs boson in VBF (Nason,C.O., 2010)
- Z/W (with decay) + 1 jet (Alioli, Nason, Re, C.O., 2010)
- dijet (Alioli, Hamilton Nason, Re, C.O., 2010)
- $t\bar{t}$ + 1 jet (Kardos, Papadopoulos, Trocsanyi, 2011) also (Alioli, Moch, Uwer, 2011)
- $t\bar{t}H$, $t\bar{t}Z/\gamma$ (Garzelli, Kardos, Papadopoulos, Trocsanyi, 2011)
- W^+W^+ plus two jets (Melia, Nason, Rontsch, Zanderighi, 2011)
- W^+W^+ plus two jets via VBF (Jäger, Zanderighi, 2011)
- $Wb\bar{b}$ (with approximated decay) (Reina, C.O., 2011)
- diboson production (with decay), (Melia, Nason, Rontsch, Zanderighi, 2011)
- tH^- (Klasen, Kovaric, Nason, Weydert, in preparation)

Running the code

The POWHEG BOX code can be downloaded from

http://powhegbox.mib.infn.it

- To download the code, you have to give the command (one single line)

  ```
  svn checkout --username anonymous --password anonymous
      svn://powhegbox.mib.infn.it/trunk/POWHEG-BOX
  ```

- Under POWHEG-BOX/Docs you can find the POWHEG BOX manual. Under POWHEG-BOX/***process-name***/Docs you can find the manual specific for each subprocess.
- Enter the ***process-name*** directory, if needed fix the Makefile and then compile the main code by giving: make pwhg_main.
  It is useful to have installed the LHAPDF and fastjet packages. If you don't have them, then fix the Makefile accordingly.
- Enter the template directory testrun-lhc and give ../pwhg_main. In this dir, you can find the powheg.input file that controls the POWHEG BOX running. Or create your own directory with your own powheg.input file, and do the runs in this directory.

Parameters in the input file

- Anything you want to be read into POWHEG can be put in the `powheg.input` file
- There is no pre-defined order of the input parameters listed in this file
- They can be read in the code by the function `powheginput('***string-to-be-read***')`. It returns a real value.
- If you want to know all the input parameters that POWHEG can handle, just search for `powheginput` thru the code
- Parameter read with # are optional and have a default value if not listed in the input file.
  For example `powheginput('#renscfact')` search the input file for the string `renscfact`. If found, then POWHEG reads the number on the same line, and returns this number
  `renscfact 2d0 !  (default 1d0) ren scale factor:  muren = muref * renscfact`

powheg.input file

```
numevts  100000   ! number of events to be generated
ih1   1           ! hadron 1 (1 for protons, -1 for antiprotons)
ih2   1           ! hadron 2 (1 for protons, -1 for antiprotons)
ebeam1 3500d0     ! energy of beam 1
ebeam2 3500d0     ! energy of beam 2

! To be set only if using internal (mlm) pdfs
! ndns1 131         ! pdf set for hadron 1 (mlm numbering)
! ndns2 131         ! pdf set for hadron 2 (mlm numbering)
! To be set only if using LHA pdfs
! 10550 cteq66
lhans1  10550         ! pdf set for hadron 1 (LHA numbering)
lhans2  10550         ! pdf set for hadron 2 (LHA numbering)
```

powheg.input file

```
! Parameters to allow or not the use of stored data
use-old-grid    1 ! if 1 use old grid if file pwggrids.dat is present
                  ! (<> 1 regenerate)
use-old-ubound  1 ! if 1 use norm of upper bounding function stored
                  ! in pwgubound.dat, if present; <> 1 regenerate

ncall1 1000000    ! number of calls for initializing the integration grid
itmx1    10       ! number of iterations for initializing the integration grid
ncall2 1000000    ! number of calls for computing the integral and finding
                  !   upper bound
itmx2    10       ! number of iterations for computing the integral and
                  !   finding upper bound
foldcsi   1       ! number of folds on csi integration
foldy     1       ! number of folds on  y  integration
foldphi   1       ! number of folds on phi integration
nubound 1000000   ! number of calls to set up the upper bounding norms
                  !   for radiation
```

powheg.input file

```
! OPTIONAL PARAMETERS

#flg_debug    1    ! activate the printing of extra info on the LHE file

withnegweights 1   ! (default 0) if on (1) use negative weights

#renscfact  1d0    ! (default 1d0) ren scale factor: muren  = muref * renscfact
#facscfact  1d0    ! (default 1d0) fac scale factor: mufact = muref * facscfact

#bornonly   1      ! (default 0) if 1 do Born only

#testplots  1      ! (default 0) if 1 plot NLO and POWHEG-alone distributions

#xupbound 2d0      ! increase upper bound for radiation generation
```

`powheg.input` **file**

```
#iseed    5437      !  Start the random number generator with seed iseed
#rand1     0        !  skipping  rand2*100000000+rand1 numbers.
#rand2     0        !  (see RM48 short writeup in CERNLIB)

#manyseeds 1        ! Used to perform multiple runs with different random
                    ! seeds in the same directory.
                    ! If set to 1, the program asks for an integer j;
                    ! The file pwgseeds.dat at line j is read, and the
                    ! integer at line j is used to initialize the random
                    ! sequence for the generation of the event.
                    ! The event file is called pwgevents-'j'.lhe
```

Comments

In the POWHEG-BOX/***process-name***/init_couplings.f file you can set the values of the physical parameters that enter this process: m_Z, m_W, m_b, $\sin^2\theta_W$, $\alpha_{\rm em}\ldots$

There are several output files. Among them:

- pwgstat.dat
  In general, the total cross section written in this file is NOT the true total cross section. It is the total cross section for unweighted events
  Check the
  negative weight fraction : ...
  in that file too. If you want only positive-weight events, then comment the corresponding line in the poweg.input file

  ```
  #  withnegweights 1   ! (default 0) if on (1) use negative weights
  ```

  and increase csi, y, phi folding to reduce the fraction of negative-weight events.
- Several topdrawer files that contain POWHEG BOX info and the user-defined histograms produced by the pwhg_analysis.f file
- pwgevents.lhe: the file that contains the events

Comments

Now the event file pwgevents.lhe is ready to be processed

- If you are interested in plotting the results from POWHEG alone, with no subsequent shower, then compile the lhef_analysis file and run it in the directory where the file of the events is
- If you want to study the results after the shower done by PYTHIA or HERWIG, then you may compile and run main-PYTHIA-lhef or main-HERWIG-lhef

The End