MLM Matching with MadGraph + Pythia

Johan Alwall (歐友涵)

National Taiwan University

MG/FR School, Beijing, May 22-26, 2013

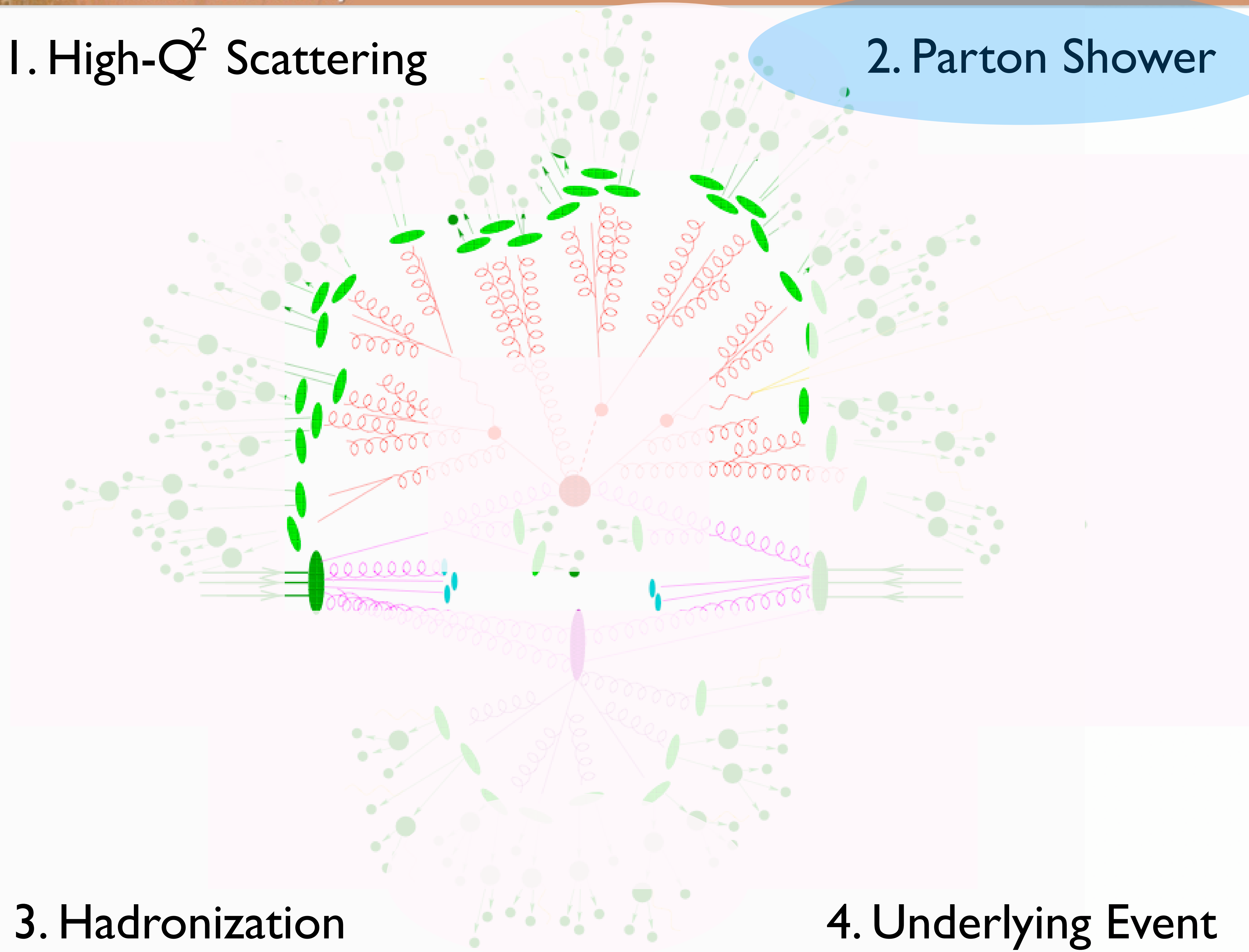
Lectures and exercises found at
https://server06.fynu.ucl.ac.be/projects/madgraph/wiki/SchoolBeijing

Outline of lectures

- Lecture I (Johan):
  - ➡ New Physics at hadron colliders
  - ➡ Monte Carlo integration and generation
  - ➡ Simulation of collider events
- Lecture II (Olivier):
  - ➡ MadGraph 5
- Lecture III (Johan):
  - ➡ MLM Matching with MadGraph and Pythia

1. High-Q^2 Scattering

2. Parton Shower



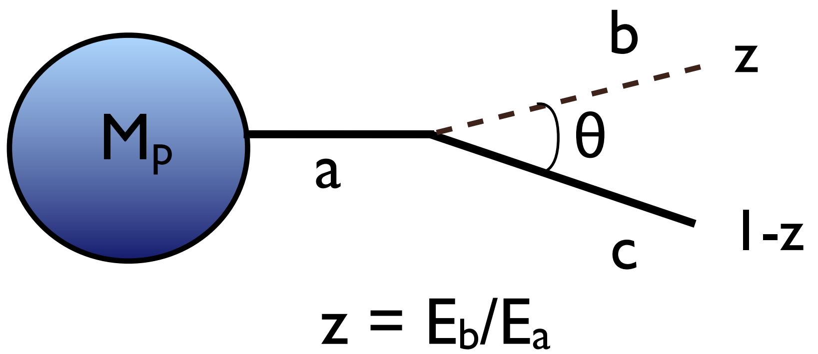
3. Hadronization

4. Underlying Event

Parton Shower basics

Matrix elements involving q →q g (or g → gg) are strongly enhanced when the final state particles are close in the phase space:

$$\frac{1}{(p_b+p_c)^2} \simeq \frac{1}{2E_bE_c(1-\cos\theta)} = \frac{1}{t}$$



soft and collinear divergencies

Collinear factorization:

$$|M_{p+1}|^2 d\Phi_{p+1} \simeq |M_p|^2 d\Phi_p \frac{dt}{t}\frac{\alpha_S}{2\pi}P(z)dzd\phi$$

when θ is small.

Parton Shower basics

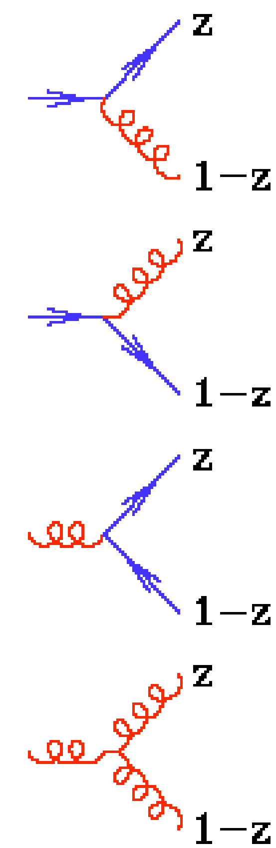
The spin averaged (unregulated) splitting functions for the various types of branching are:

$$\hat{P}_{qq}(z) = C_F \left[\frac{1+z^2}{(1-z)} \right],$$

$$\hat{P}_{gq}(z) = C_F \left[\frac{1+(1-z)^2}{z} \right],$$

$$\hat{P}_{qg}(z) = T_R \left[z^2 + (1-z)^2 \right],$$

$$\hat{P}_{gg}(z) = C_A \left[\frac{z}{(1-z)} + \frac{1-z}{z} + z(1-z) \right].$$



$$C_F = \frac{4}{3}, C_A = 3, T_R = \frac{1}{2}.$$

Parton Shower basics

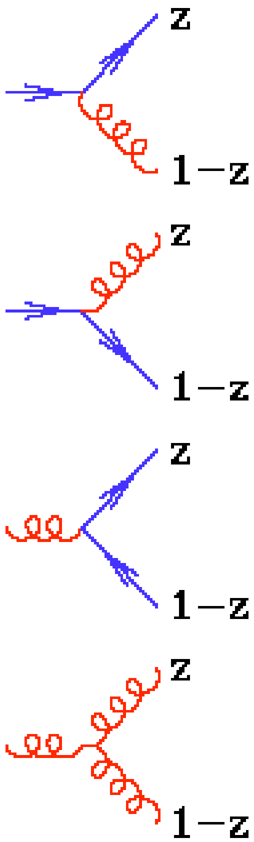
The spin averaged (unregulated) splitting functions for the various types of branching are:

$$\hat{P}_{qq}(z) = C_F \left[\frac{1+z^2}{(1-z)}\right],$$

$$\hat{P}_{gq}(z) = C_F \left[\frac{1+(1-z)^2}{z}\right],$$

$$\hat{P}_{qg}(z) = T_R \left[z^2+(1-z)^2\right],$$

$$\hat{P}_{gg}(z) = C_A \left[\frac{z}{(1-z)} + \frac{1-z}{z} + z(1-z)\right].$$



$$C_F = \frac{4}{3}, C_A = 3, T_R = \frac{1}{2}.$$

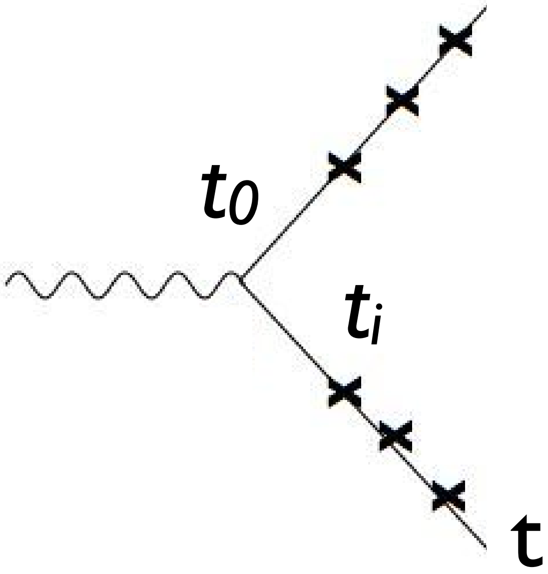
Comments:

* Gluons radiate the most

* There are soft divergences in z=1 and z=0.

* P_{qg} has no soft divergences.

Parton Shower basics



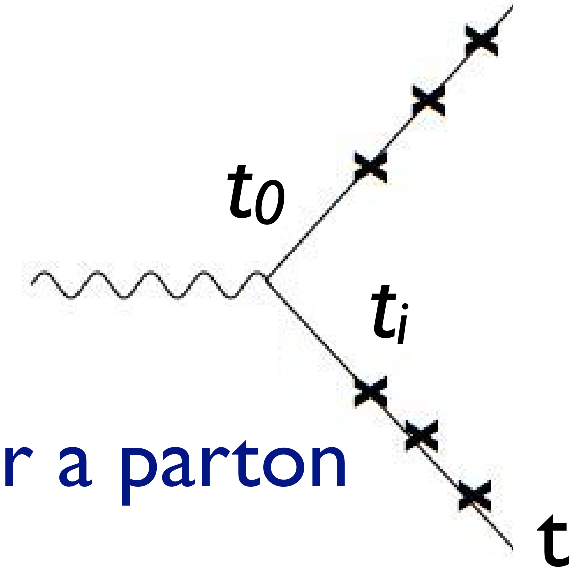
\mathcal{[
{\delta t]

{\color[rgb
(t,t_0)\sim
\int^1_z dz

{\color[rgb]
{\delta t}{

{\color[rgb]
{t_0}\frac{c
\Delta(t,t_0

Parton Shower basics



- Now, consider the non-branching probability for a parton at a given virtuality t_i:

$$\mathcal{P}_{\mathrm{non-branching}}(t_i) = 1 - \mathcal{P}_{\mathrm{branching}}(t_i) = 1 - \frac{\delta t}{t_i}\frac{\alpha_s}{2\pi}\int_z^1 dz \hat{P}(z)$$

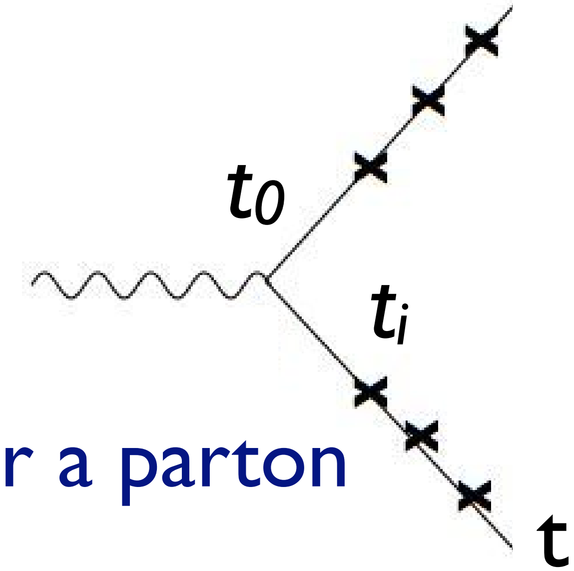
\mathcal{
\delta t

{\color[rgb
(t,t_0)\sim
\int^1_z dz

{\color[rgb]
{\delta t}{

{\color[rgb]
{t_0}\frac{
\Delta(t,t_

Parton Shower basics



- Now, consider the non-branching probability for a parton at a given virtuality t_i:

$$\mathcal{P}_{\rm non-branching}(t_i) = 1 - \mathcal{P}_{\rm branching}(t_i) = 1 - \frac{\delta t}{t_i}\frac{\alpha_s}{2\pi}\int_z^1 dz\hat{P}(z)$$

- The total non-branching probability between virtualities t and t_0:

$$\mathcal{P}_{\rm non-branching}(t,t_0) \simeq \prod_{i=0}^{N}\left(1 - \frac{\delta t}{t_i}\frac{\alpha_s}{2\pi}\int_z^1 dz\hat{P}(z)\right)$$
$$\simeq e^{\sum_{i=0}^{N}\left(-\frac{\delta t}{t_i}\frac{\alpha_s}{2\pi}\int_z^1 dz\hat{P}(z)\right)}$$
$$\simeq e^{-\int_t^{t_0}\frac{dt'}{t'}\frac{\alpha_s}{2\pi}\int_z^1 dz\hat{P}(z)} = \Delta(t,t_0)$$

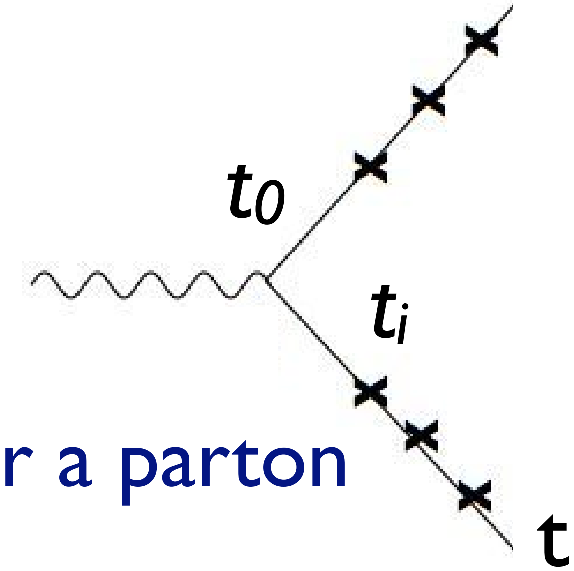
\mathcal{
\delta t

{\color[rgb
(t,t_0)\sim
\int^1_z dz

{\color[rgb]
{\delta t}{

{\color[rgb]
{t_0}\frac{
\Delta(t,t_

Parton Shower basics



- Now, consider the non-branching probability for a parton at a given virtuality t_i:

$$\mathcal{P}_{\mathrm{non-branching}}(t_i) = 1 - \mathcal{P}_{\mathrm{branching}}(t_i) = 1 - \frac{\delta t}{t_i}\frac{\alpha_s}{2\pi}\int_z^1 dz\hat{P}(z)$$

\mathcal{[
\delta t

- The total non-branching probability between virtualities t and t_0:

$$\mathcal{P}_{\mathrm{non-branching}}(t,t_0) \simeq \prod_{i=0}^{N}\left(1 - \frac{\delta t}{t_i}\frac{\alpha_s}{2\pi}\int_z^1 dz\hat{P}(z)\right)$$

$$\simeq e^{\sum_{i=0}^{N}\left(-\frac{\delta t}{t_i}\frac{\alpha_s}{2\pi}\int_z^1 dz\hat{P}(z)\right)}$$

$$\simeq e^{-\int_t^{t_0}\frac{dt'}{t'}\frac{\alpha_s}{2\pi}\int_z^1 dz\hat{P}(z)} = \Delta(t,t_0)$$

{\color[rgb
(t,t_0)\sim
\int^1_z dz

{\color[rgb]
{\delta t}{t

{\color[rgb]
{t_0}\frac{d
\Delta(t,t_0

- This is the famous "Sudakov form factor"

Final-state parton showers

Final-state parton showers

With the Sudakov form factor, we can now implement a final-state parton shower in a Monte Carlo event generator!

Final-state parton showers

With the Sudakov form factor, we can now implement a final-state parton shower in a Monte Carlo event generator!

1. Start the evolution at the virtual mass scale t_0 (e.g. the mass of the decaying particle) and momentum fraction $z_0 = 1$

Final-state parton showers

With the Sudakov form factor, we can now implement a final-state parton shower in a Monte Carlo event generator!

1. Start the evolution at the virtual mass scale t_0 (e.g. the mass of the decaying particle) and momentum fraction $z_0 = 1$
2. Given a virtual mass scale t_i and momentum fraction x_i at some stage in the evolution, generate the scale of the next emission t_{i+1} according to the Sudakov probability $\Delta(t_i,t_{i+1})$ by solving
   $\Delta(t_{i+1},t_i) = R$
   where R is a random number (uniform on [0, 1]).

Final-state parton showers

With the Sudakov form factor, we can now implement a final-state parton shower in a Monte Carlo event generator!

1. Start the evolution at the virtual mass scale t_0 (e.g. the mass of the decaying particle) and momentum fraction $z_0 = 1$

2. Given a virtual mass scale t_i and momentum fraction x_i at some stage in the evolution, generate the scale of the next emission t_{i+1} according to the Sudakov probability $\Delta(t_i,t_{i+1})$ by solving
$\Delta(t_{i+1},t_i) = R$
where R is a random number (uniform on [0, 1]).

3. If $t_{i+1} < t_{cut}$ it means that the shower has finished.

Final-state parton showers

With the Sudakov form factor, we can now implement a final-state parton shower in a Monte Carlo event generator!

1. Start the evolution at the virtual mass scale t_0 (e.g. the mass of the decaying particle) and momentum fraction $z_0 = 1$
2. Given a virtual mass scale t_i and momentum fraction x_i at some stage in the evolution, generate the scale of the next emission t_{i+1} according to the Sudakov probability $\Delta(t_i,t_{i+1})$ by solving
$\Delta(t_{i+1},t_i) = R$
where R is a random number (uniform on [0, 1]).
3. If $t_{i+1} < t_{cut}$ it means that the shower has finished.
4. Otherwise, generate $z = z_i/z_{i+1}$ with a distribution proportional to $(\alpha_s/2\pi)P(z)$, where $P(z)$ is the appropriate splitting function.

Final-state parton showers

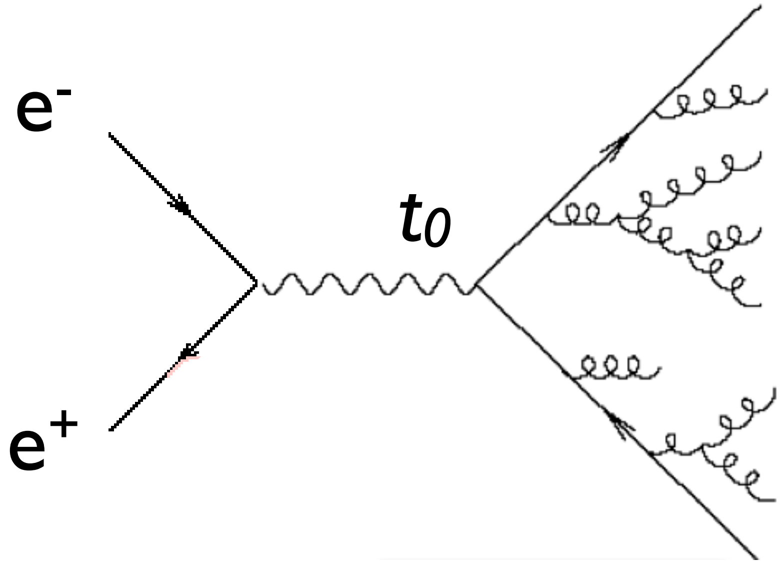
With the Sudakov form factor, we can now implement a final-state parton shower in a Monte Carlo event generator!

1. Start the evolution at the virtual mass scale t_0 (e.g. the mass of the decaying particle) and momentum fraction $z_0 = 1$
2. Given a virtual mass scale t_i and momentum fraction x_i at some stage in the evolution, generate the scale of the next emission t_{i+1} according to the Sudakov probability $\Delta(t_i,t_{i+1})$ by solving
 $\Delta(t_{i+1},t_i) = R$
 where R is a random number (uniform on [0, 1]).
3. If $t_{i+1} < t_{cut}$ it means that the shower has finished.
4. Otherwise, generate $z = z_i/z_{i+1}$ with a distribution proportional to $(\alpha_s/2\pi)P(z)$, where $P(z)$ is the appropriate splitting function.
5. For each emitted particle, iterate steps 2-4 until branching stops.

Final-state parton showers

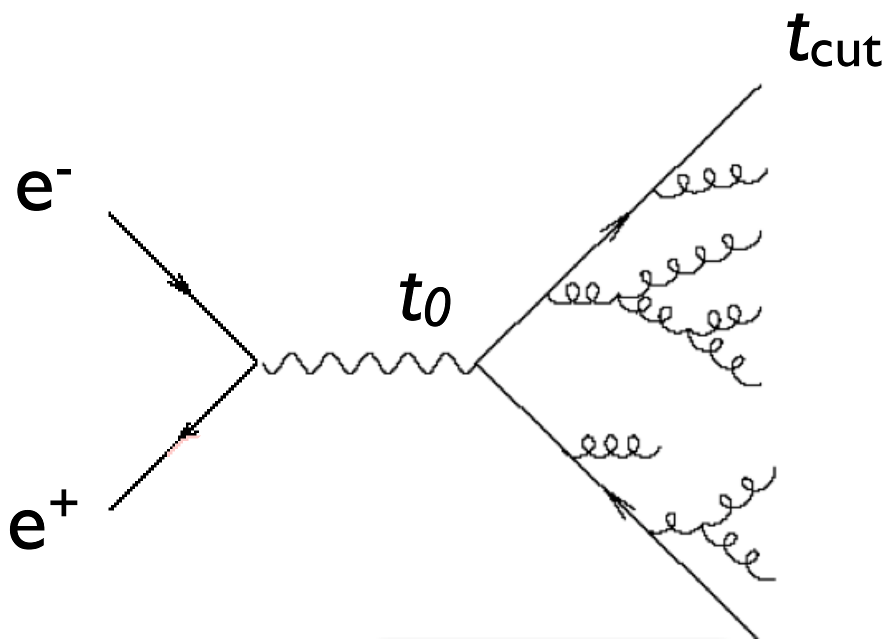
Final-state parton showers

- The result is a "cascade" or "shower" of partons with ever smaller virtualities.



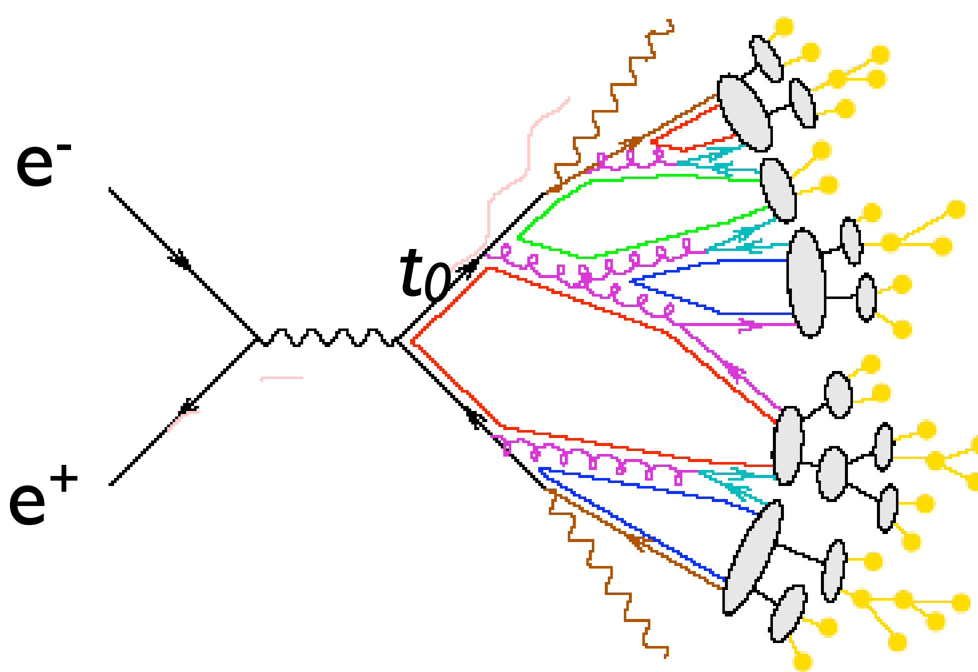
Final-state parton showers

- The result is a "cascade" or "shower" of partons with ever smaller virtualities.
- The cutoff scale t_{cut} is usually set close to 1 GeV, the scale where non-perturbative effects start dominating over the perturbative parton shower.



Final-state parton showers

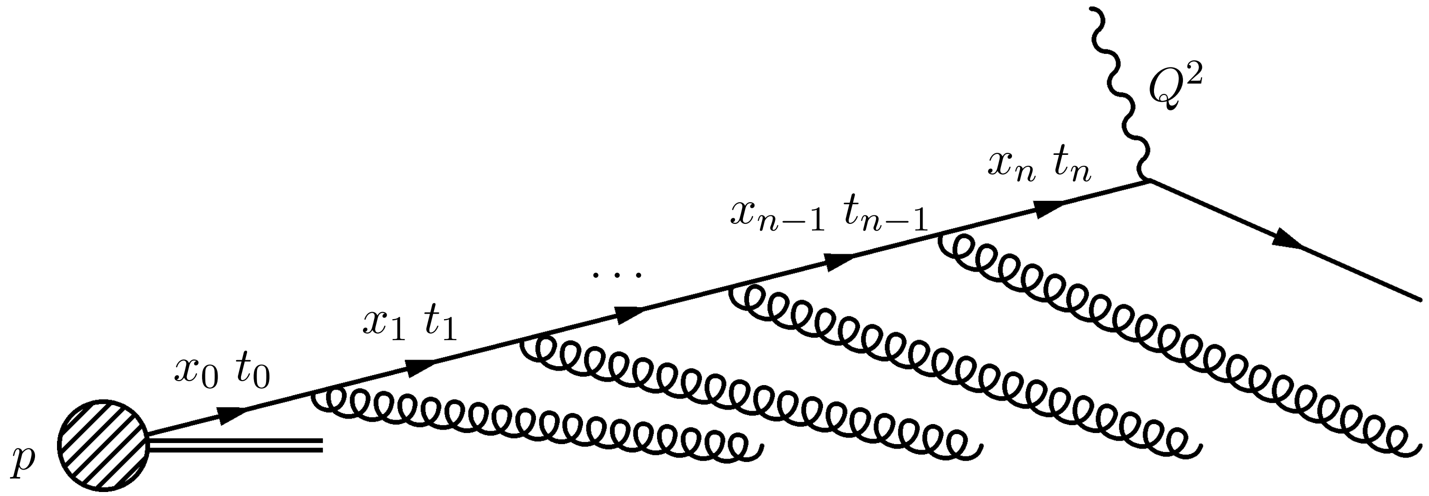
- The result is a "cascade" or "shower" of partons with ever smaller virtualities.
- The cutoff scale t_{cut} is usually set close to 1 GeV, the scale where non-perturbative effects start dominating over the perturbative parton shower.
- At this point, phenomenological models are used to simulate how the partons turn into color-neutral hadrons. Hadronization not sensitive to the physics at scale t_0, but only t_{cut}! (can be tuned once and for all!)



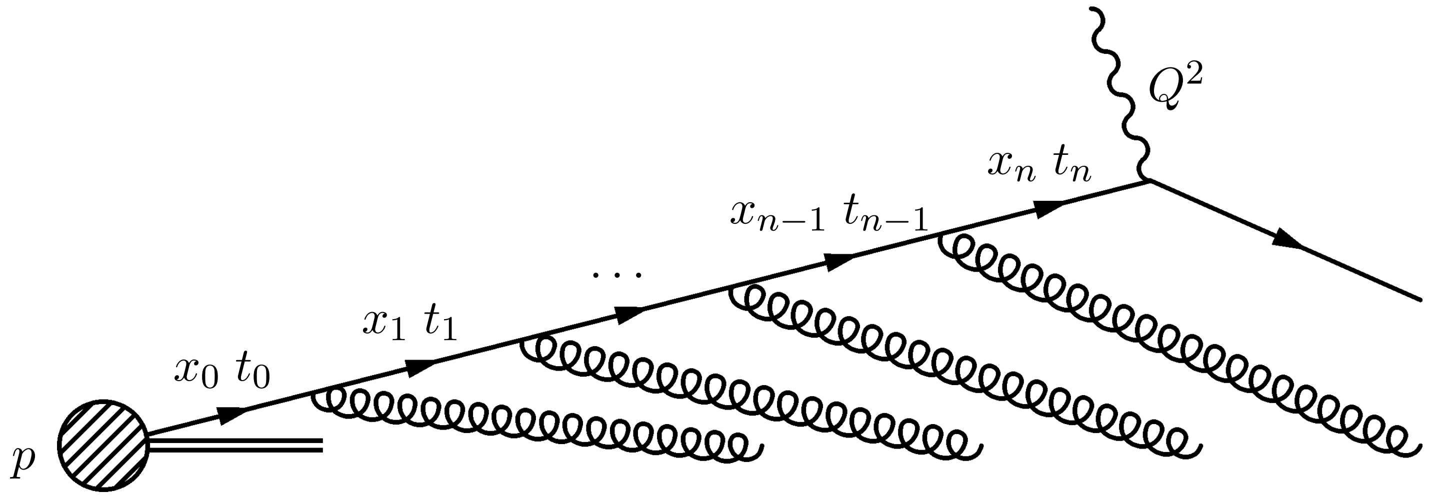
Initial-state parton splittings

- So far, we have looked at final-state (time-like) splittings
- For initial state, the splitting functions are the same
- However, there is another ingredient - the parton density (or distribution) functions (PDFs)
  - ➡ Probability to find a given parton in a hadron at a given momentum fraction $x = p_z/P_z$ and scale t
- How do the PDFs evolve with increasing t?

Initial-state parton splittings



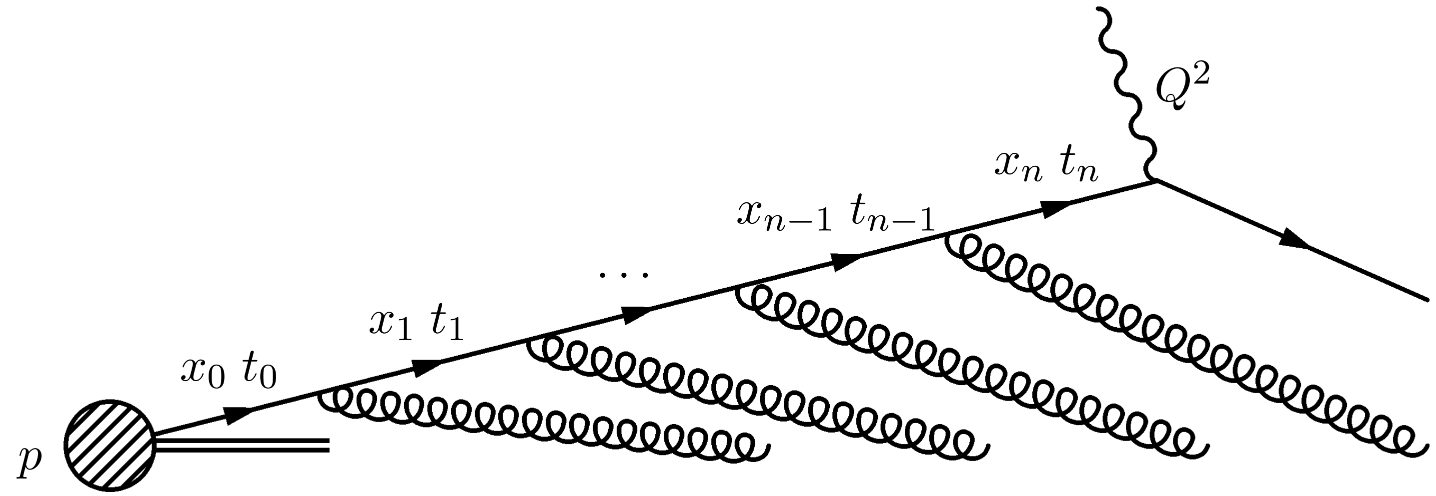
Initial-state parton splittings



- Start with a quark PDF $f_0(x)$ at scale t_0. After a single parton emission, the probability to find the quark at virtuality $t > t_0$ is

$$f(x,t) = f_0(x) + \int_{t_0}^{t} \frac{dt'}{t'} \frac{\alpha_s}{2\pi} \int_x^1 \frac{dz}{z} P(z) f_0\left(\frac{x}{z}\right)$$

Initial-state parton splittings



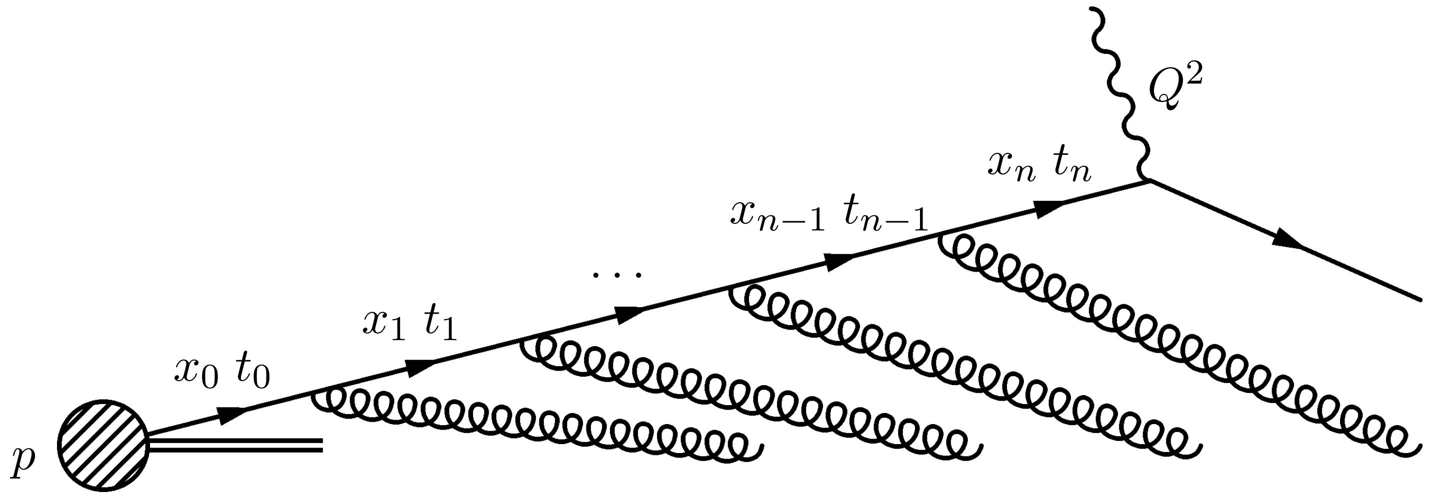
- Start with a quark PDF $f_0(x)$ at scale t_0. After a single parton emission, the probability to find the quark at virtuality $t > t_0$ is

$$f(x,t) = f_0(x) + \int_{t_0}^{t} \frac{dt'}{t'} \frac{\alpha_s}{2\pi} \int_x^1 \frac{dz}{z} P(z) f_0\left(\frac{x}{z}\right)$$

- After a second emission, we have

$$f(x,t) = f_0(x) + \int_{t_0}^{t} \frac{dt'}{t'} \frac{\alpha_s}{2\pi} \int_x^1 \frac{dz}{z} P(z) \Big\{ f_0\left(\frac{x}{z}\right) + \int_{t_0}^{t'} \frac{dt''}{t''} \frac{\alpha_s}{2\pi} \int_{x/z}^1 \frac{dz'}{z'} P(z') f_0\left(\frac{x}{zz'}\right) \Big\}$$

Initial-state parton splittings



- Start with a quark PDF $f_0(x)$ at scale t_0. After a single parton emission, the probability to find the quark at virtuality $t > t_0$ is

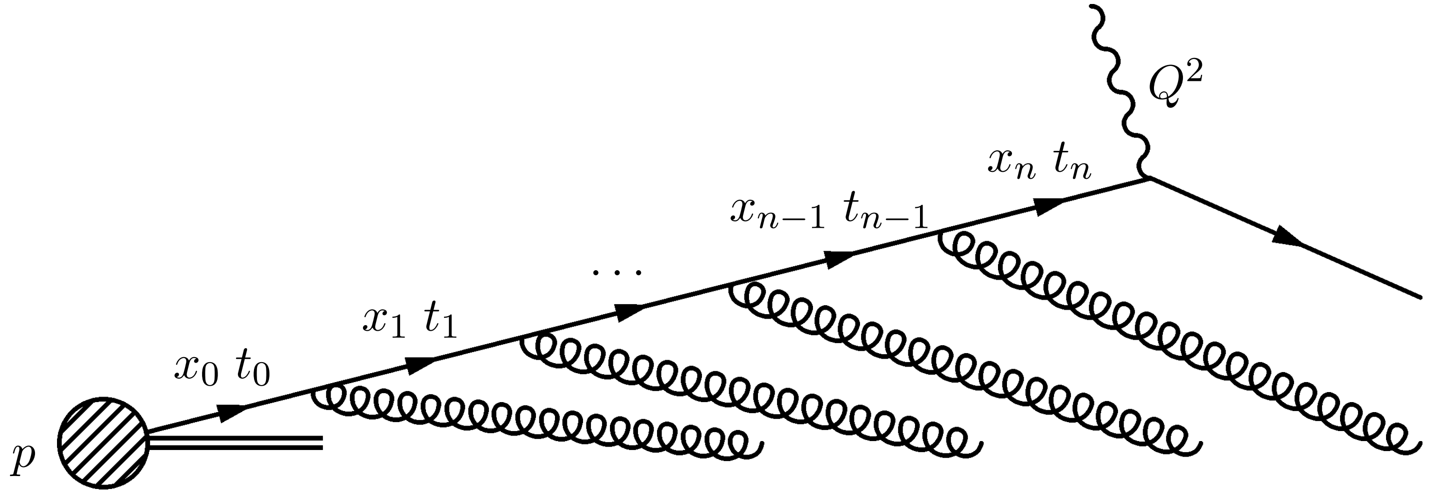
$$f(x,t) = f_0(x) + \int_{t_0}^{t} \frac{dt'}{t'} \frac{\alpha_s}{2\pi} \int_x^1 \frac{dz}{z} P(z) f_0\left(\frac{x}{z}\right)$$

- After a second emission, we have

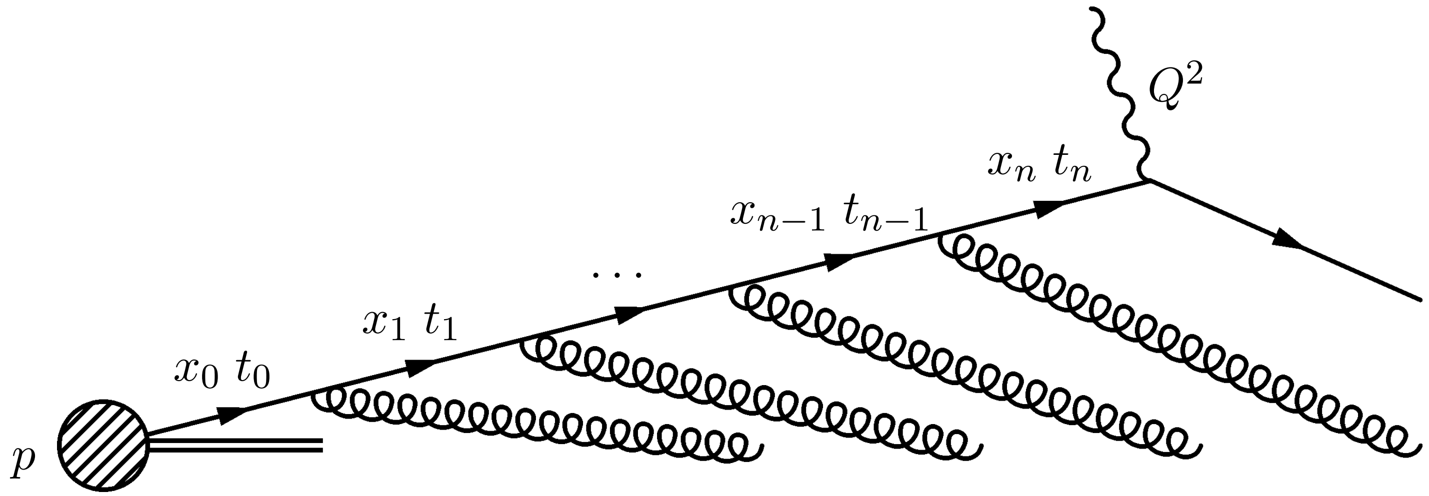
$$f(x,t) = f_0(x) + \int_{t_0}^{t} \frac{dt'}{t'} \frac{\alpha_s}{2\pi} \int_x^1 \frac{dz}{z} P(z) \left\{ f_0\left(\frac{x}{z}\right) + \int_{t_0}^{t'} \frac{dt''}{t''} \frac{\alpha_s}{2\pi} \int_{x/z}^1 \frac{dz'}{z'} P(z') f_0\left(\frac{x}{zz'}\right) \right\}$$

$f(x/z, t')$

The DGLAP equation

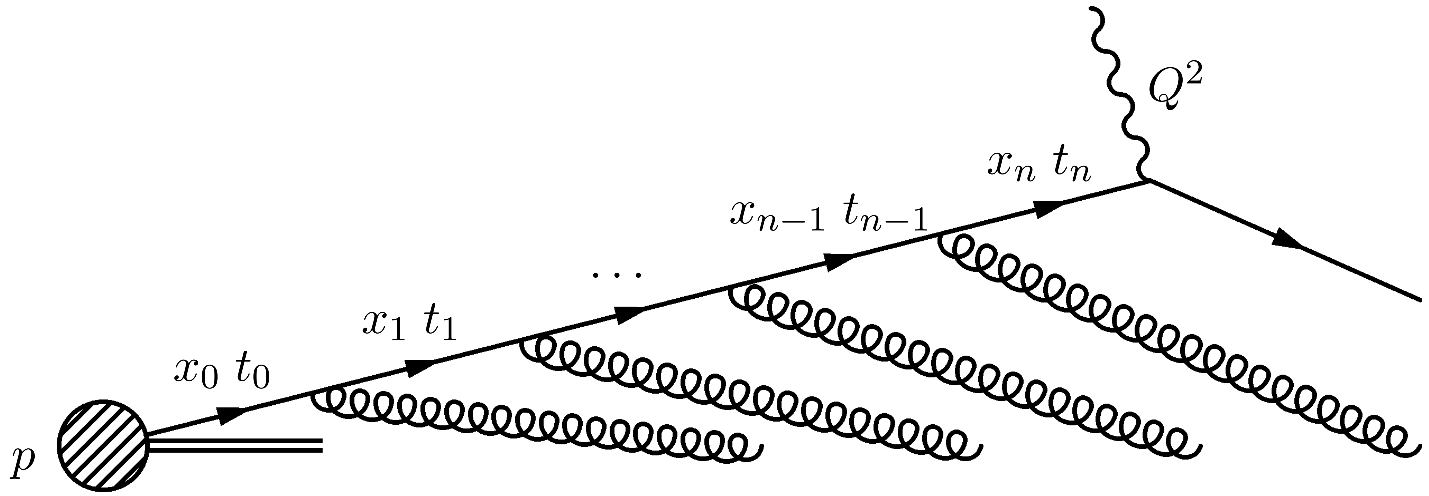


The DGLAP equation



- So for multiple parton splittings, we arrive at an integral equation:

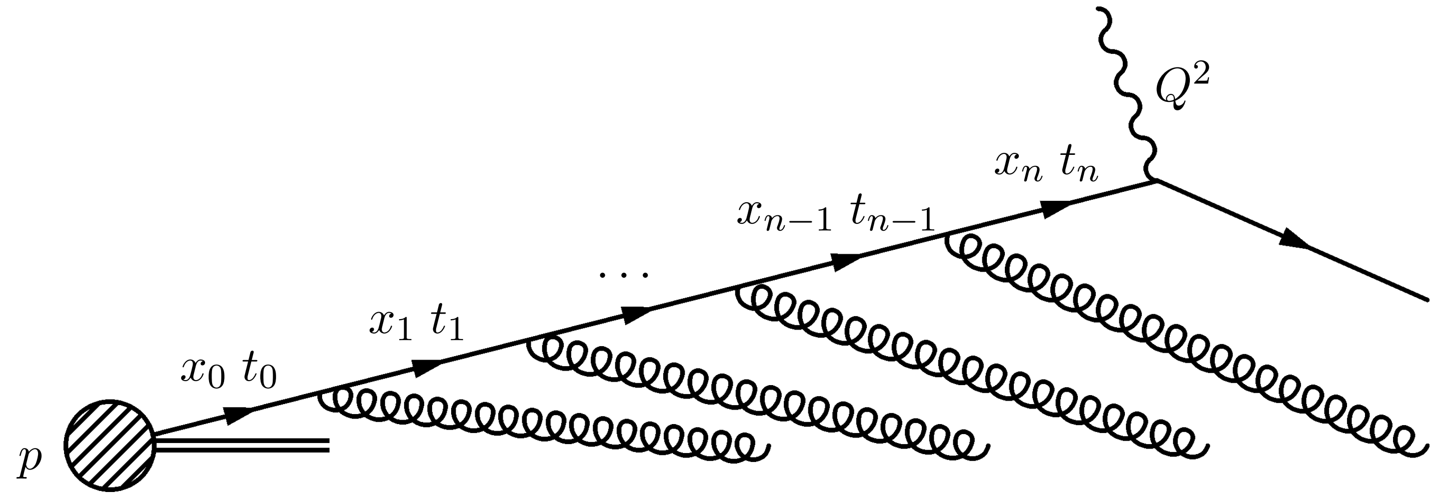
The DGLAP equation



- So for multiple parton splittings, we arrive at an integral equation:

$$t\frac{\partial}{\partial t}f_i(x,t) = \int_x^1 \frac{dz}{z}\frac{\alpha_s}{2\pi}P_{ij}(z)f_j\left(\frac{x}{z}\right)$$

The DGLAP equation

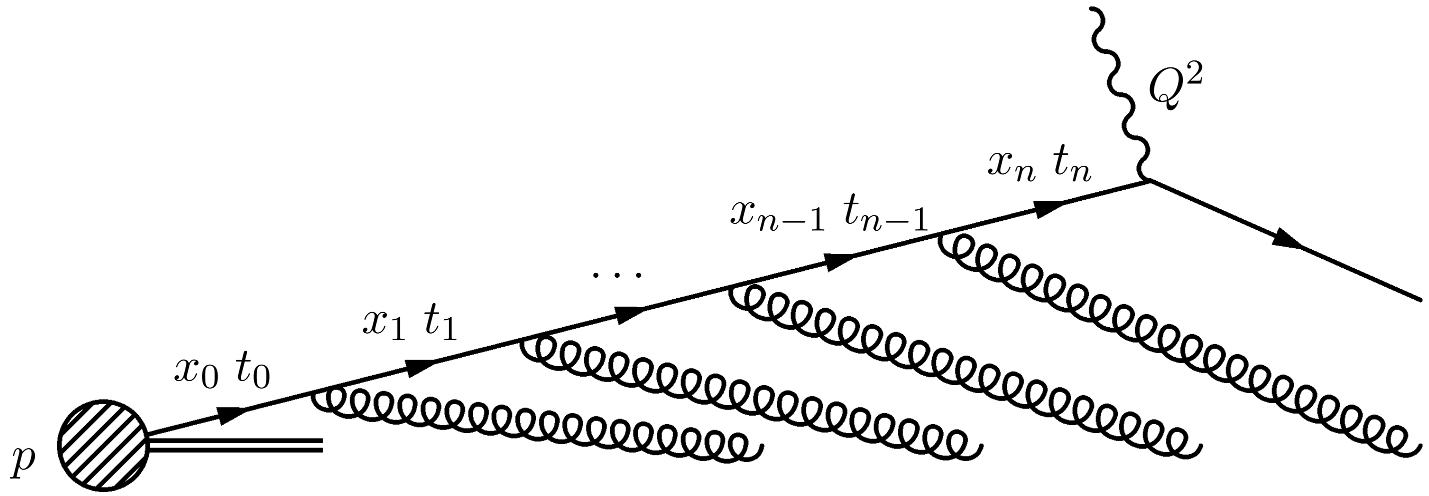


- So for multiple parton splittings, we arrive at an integral equation:

$$t\frac{\partial}{\partial t}f_i(x,t) = \int_x^1 \frac{dz}{z}\frac{\alpha_s}{2\pi}P_{ij}(z)f_j\left(\frac{x}{z}\right)$$

- This is the famous DGLAP equation (where we have taken into account the multiple parton species *i*, *j*). The boundary condition for the equation is the initial PDFs $f_{i0}(x)$ at a starting scale t_0 (again around 1 GeV).

The DGLAP equation



- So for multiple parton splittings, we arrive at an integral equation:

$$t\frac{\partial}{\partial t}f_i(x,t) = \int_x^1 \frac{dz}{z}\frac{\alpha_s}{2\pi}P_{ij}(z)f_j\left(\frac{x}{z}\right)$$

- This is the famous DGLAP equation (where we have taken into account the multiple parton species *i*, *j*). The boundary condition for the equation is the initial PDFs $f_{i0}(x)$ at a starting scale t_0 (again around 1 GeV).

- These starting PDFs are fitted to experimental data.

Initial-state parton showers

- To simulate parton radiation from the initial state, we start with the hard scattering, and then "devolve" the DGLAP evolution to get back to the original hadron: Backwards evolution!

- In backwards evolution, the Sudakovs include also the PDFs - this follows from the DGLAP equation and ensures conservation of probability:

$$\Delta_{Ii}(x,t_1,t_2) = \exp\left\{-\int_{t_1}^{t_2} dt' \sum_j \int_x^1 \frac{dx'}{x'} \frac{\alpha_s(t')}{2\pi} P_{ij}\left(\frac{x}{x'}\right) \frac{f_i(x',t')}{f_j(x,t')}\right\}$$

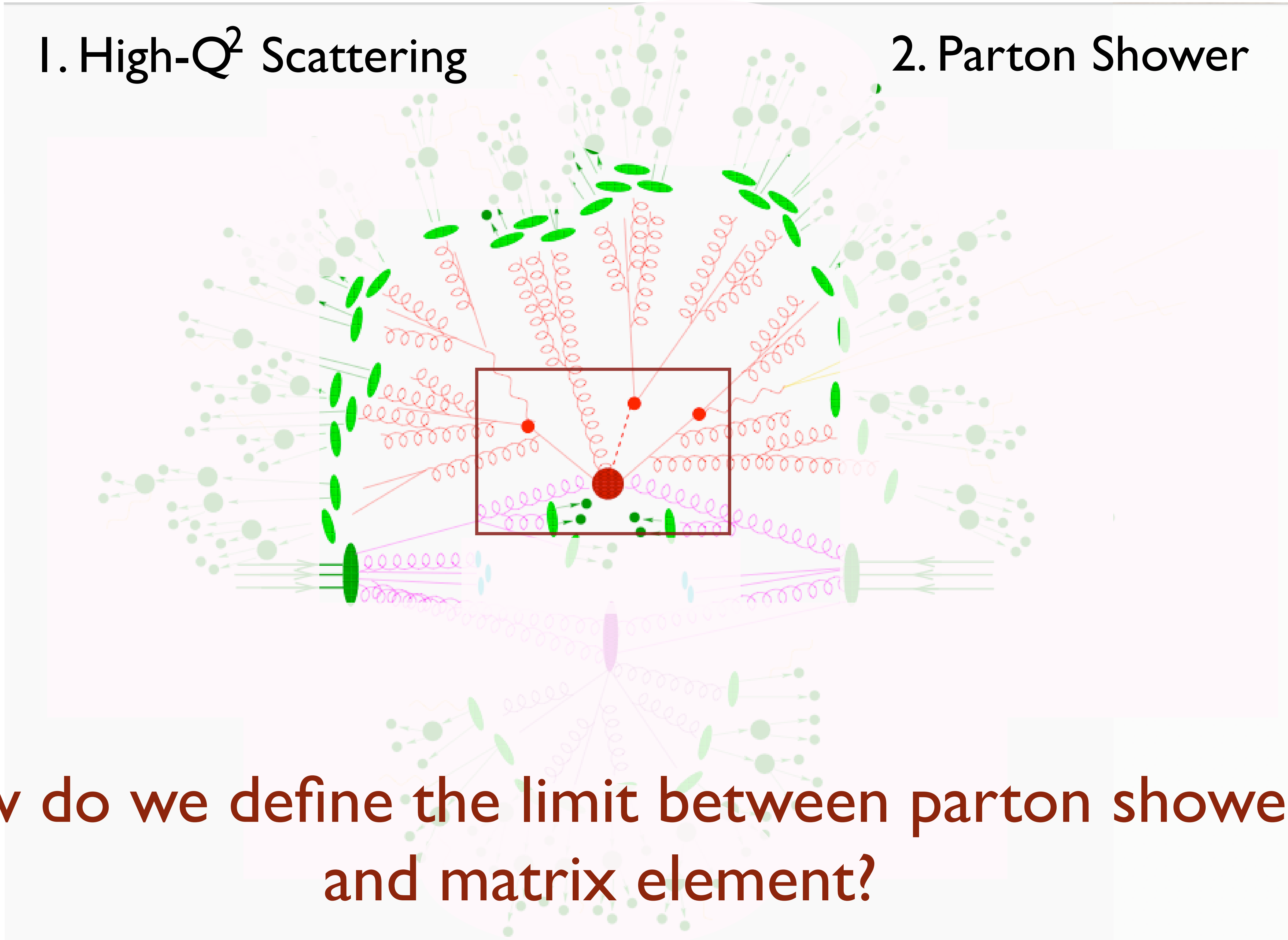
  This represents the probability that parton *i* will stay at the same *x* (no splittings) when evolving from t_1 to t_2.

- The shower simulation is now done as in FS shower!

Parton Shower MC event generators

- In both initial-state and final-state showers, the definition of t is not unique, as long as it has the dimension of scale:
- Different parton shower generators have made different choices:
  - ➡ Ariadne: "dipole p_T"
  - ➡ Herwig: $E \cdot \theta$
  - ➡ Pythia (old): virtuality q^2
  - ➡ Pythia 6.4 and Pythia 8: p_T
  - ➡ Sherpa: v. 1.1 virtuality q^2, v. 1.2 "dipole p_T"
- Note that all of the above are complete MC event generators with matrix elements, parton showers, hadronization, decay, and underlying event simulation.

Back to our favorite piece of art!



How do we define the limit between parton shower and matrix element?

Matrix Elements vs. Parton Showers

Matrix Elements vs. Parton Showers

ME

1. Fixed order calculation
2. Computationally expensive
3. Limited number of particles
4. Valid when partons are hard and well separated
5. Quantum interference correct
6. Needed for multi-jet description

Matrix Elements vs. Parton Showers

ME	Shower MC
1. Fixed order calculation 2. Computationally expensive 3. Limited number of particles 4. Valid when partons are hard and well separated 5. Quantum interference correct 6. Needed for multi-jet description	1. Resums logs to all orders 2. Computationally cheap 3. No limit on particle multiplicity 4. Valid when partons are collinear and/or soft 5. Partial interference through angular ordering 6. Needed for hadronization

Matrix Elements vs. Parton Showers

ME	Shower MC
1. Fixed order calculation 2. Computationally expensive 3. Limited number of particles 4. Valid when partons are hard and well separated 5. Quantum interference correct 6. Needed for multi-jet description	1. Resums logs to all orders 2. Computationally cheap 3. No limit on particle multiplicity 4. Valid when partons are collinear and/or soft 5. Partial interference through angular ordering 6. Needed for hadronization

Approaches are complementary: merge them!

Matrix Elements vs. Parton Showers

ME

1. Fixed order calculation
2. Computationally expensive
3. Limited number of particles
4. Valid when partons are hard and well separated
5. Quantum interference correct
6. Needed for multi-jet description

Shower MC

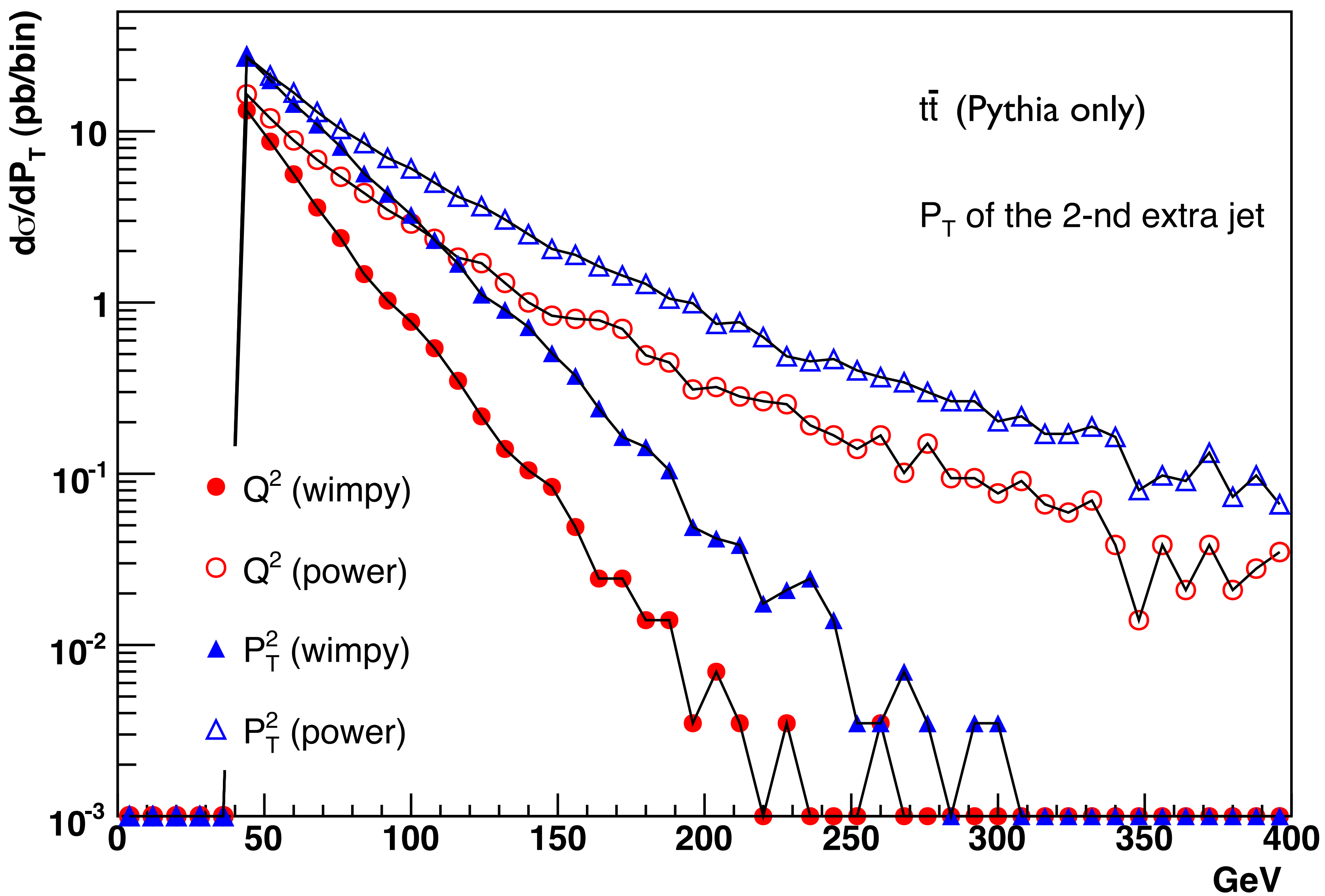
1. Resums logs to all orders
2. Computationally cheap
3. No limit on particle multiplicity
4. Valid when partons are collinear and/or soft
5. Partial interference through angular ordering
6. Needed for hadronization

Approaches are complementary: merge them!

Difficulty: avoid double counting, ensure smooth distributions

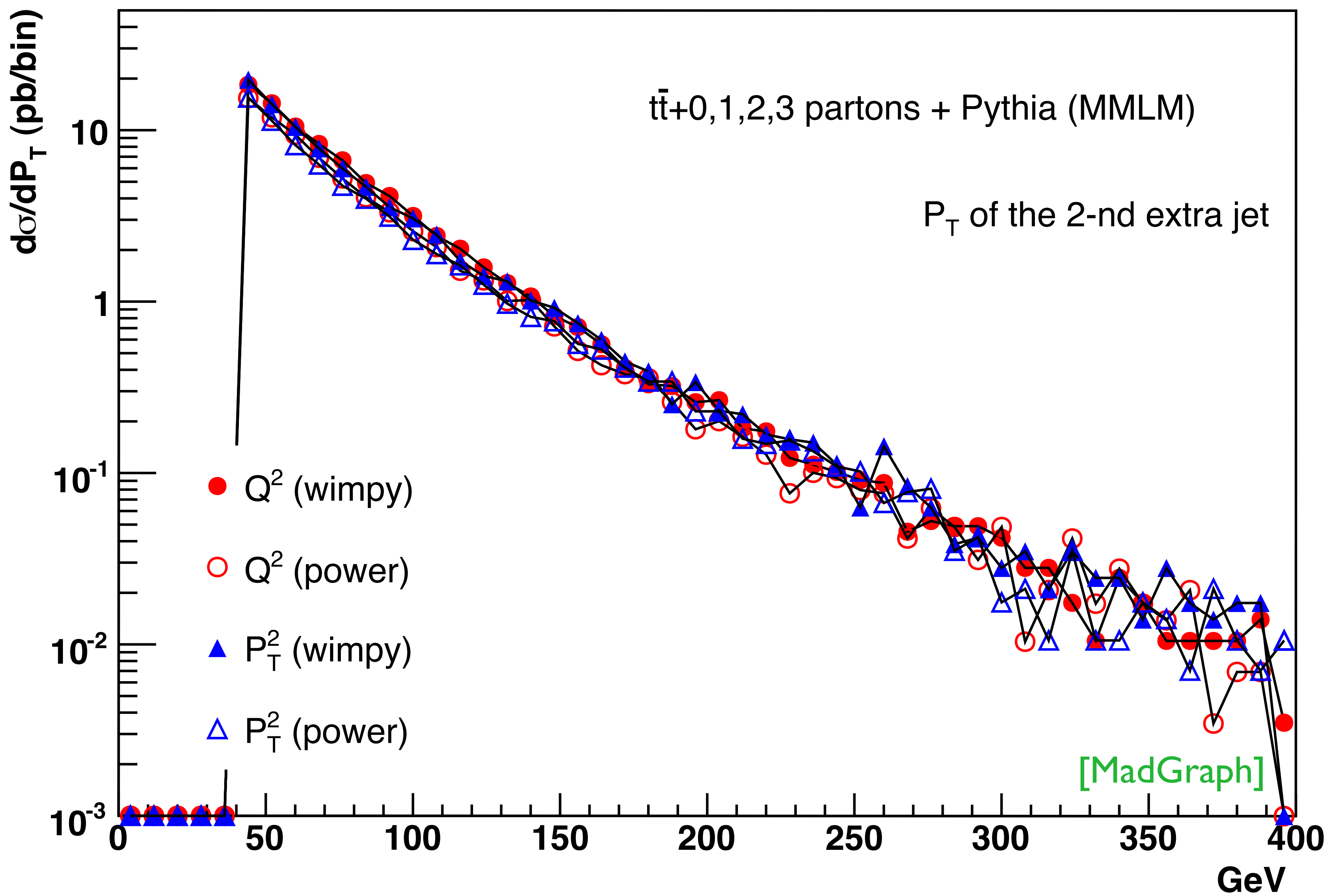
PS alone vs matched samples

In the soft-collinear approximation of Parton Shower MCs, parameters are used to tune the result ⇒ Large variation in results (small prediction power)

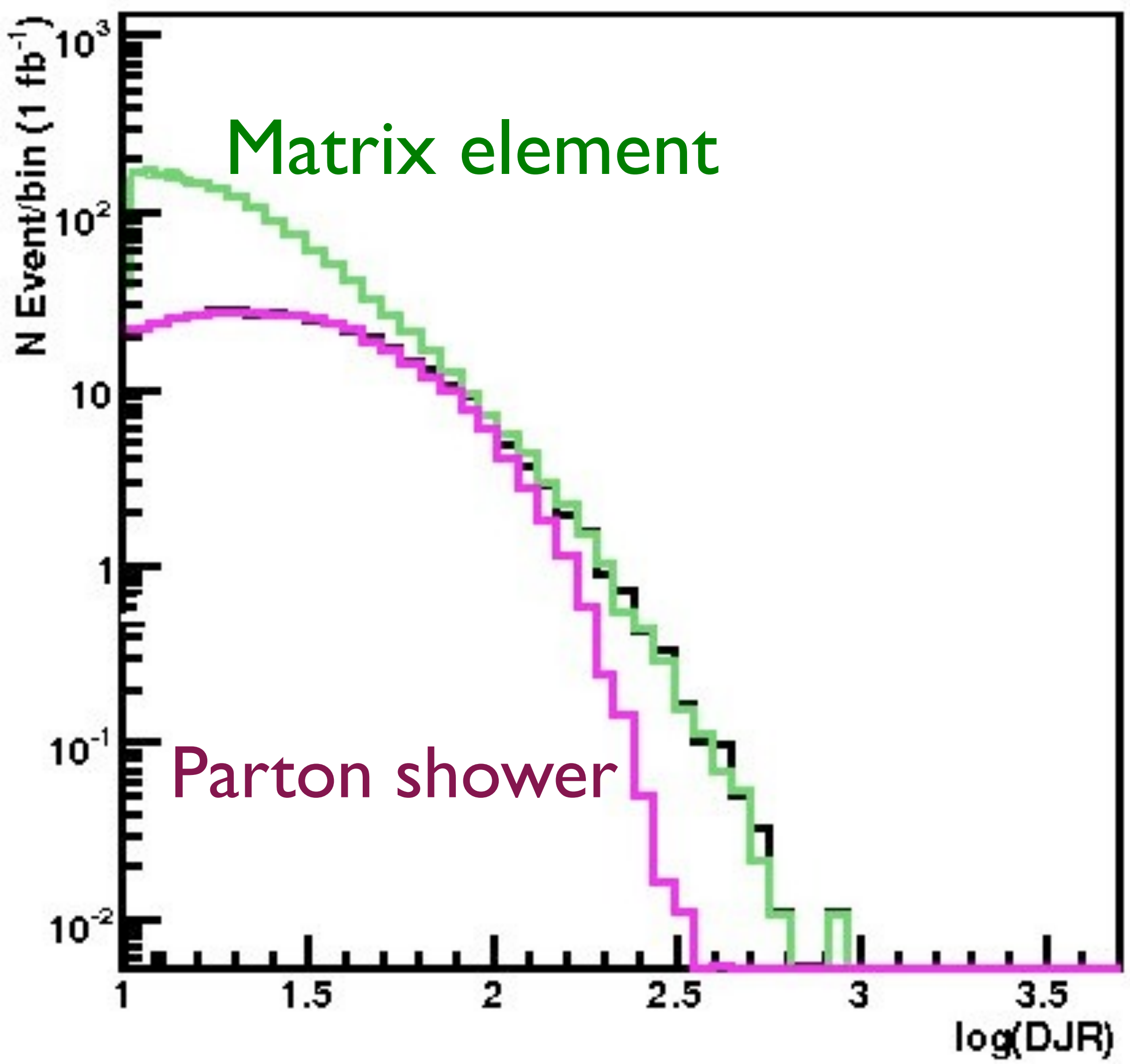


PS alone vs ME matching

In a matched sample these differences are irrelevant since the behavior at high pt is dominated by the matrix element.



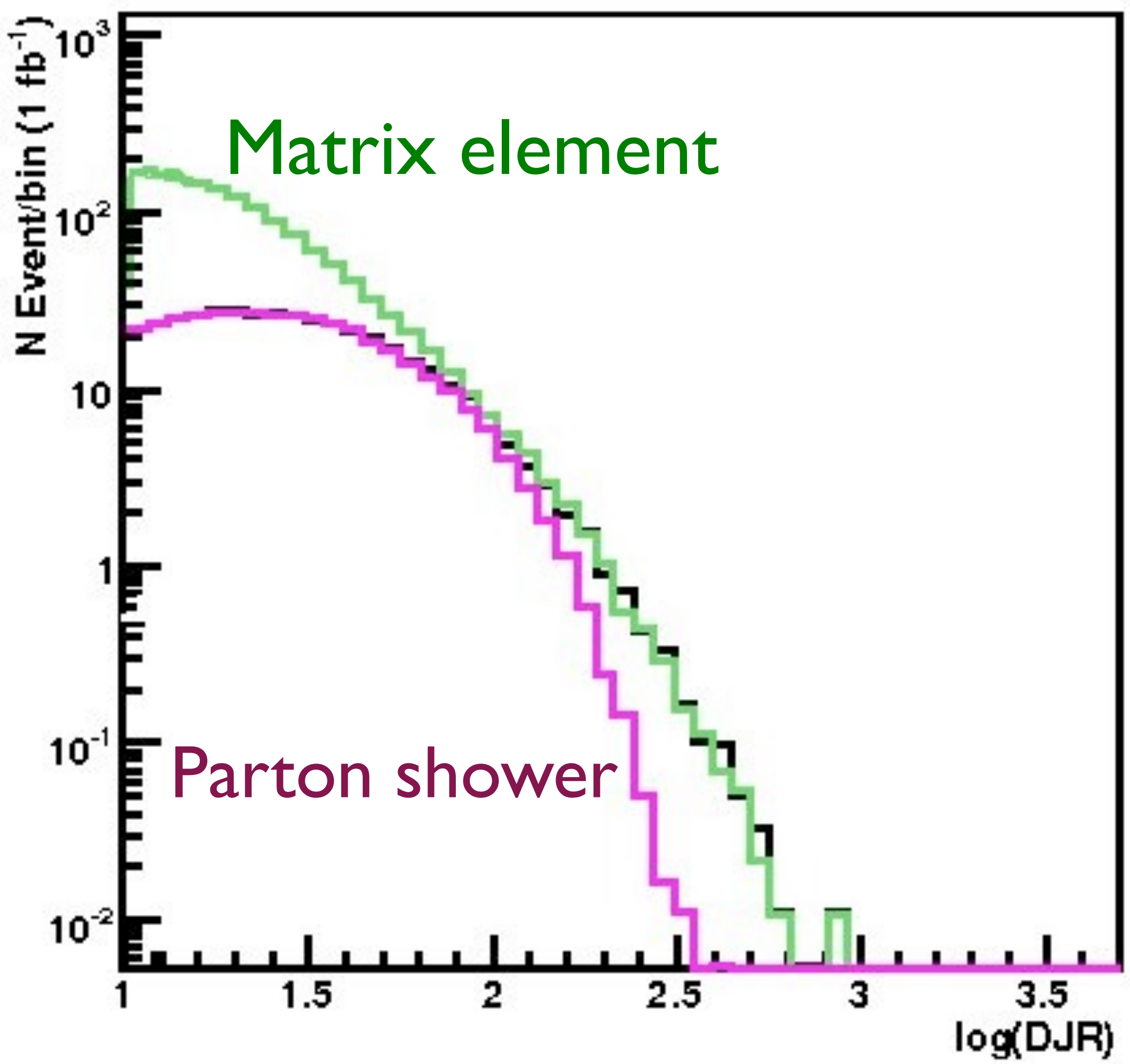
Goal for ME-PS merging/matching



2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Goal for ME-PS merging/matching

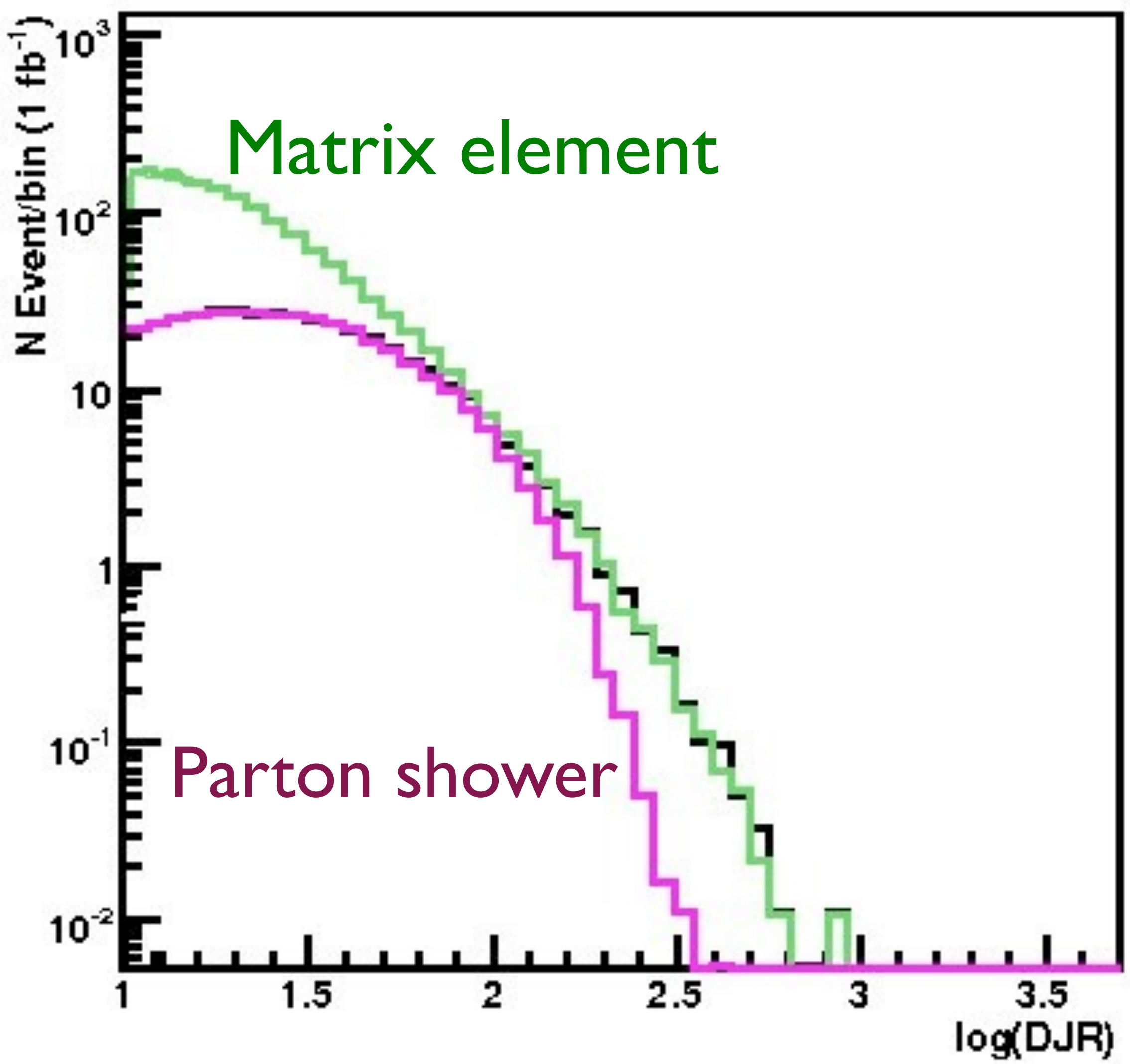
- Regularization of matrix element divergence



2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Goal for ME-PS merging/matching

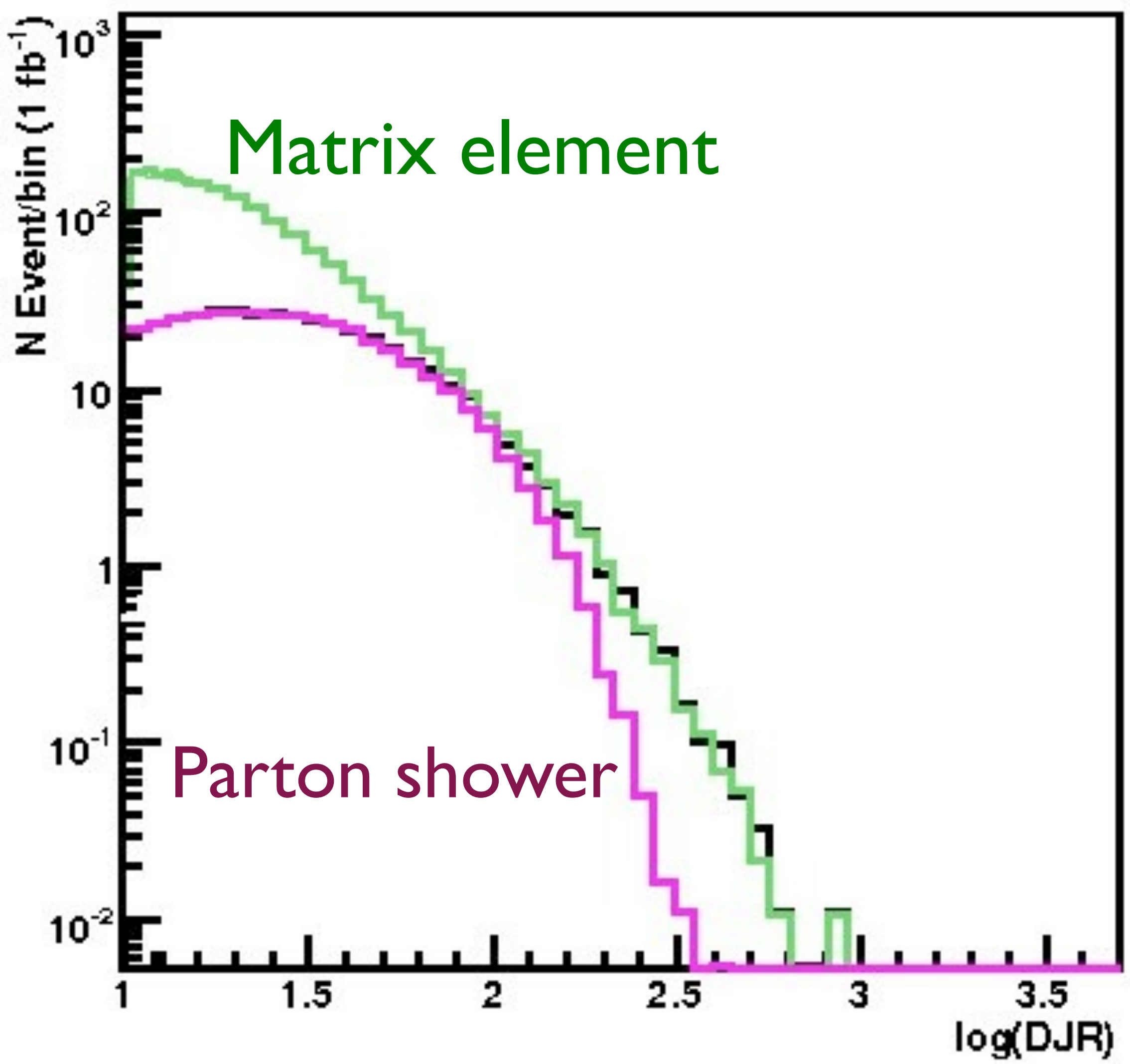
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta



2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Goal for ME-PS merging/matching

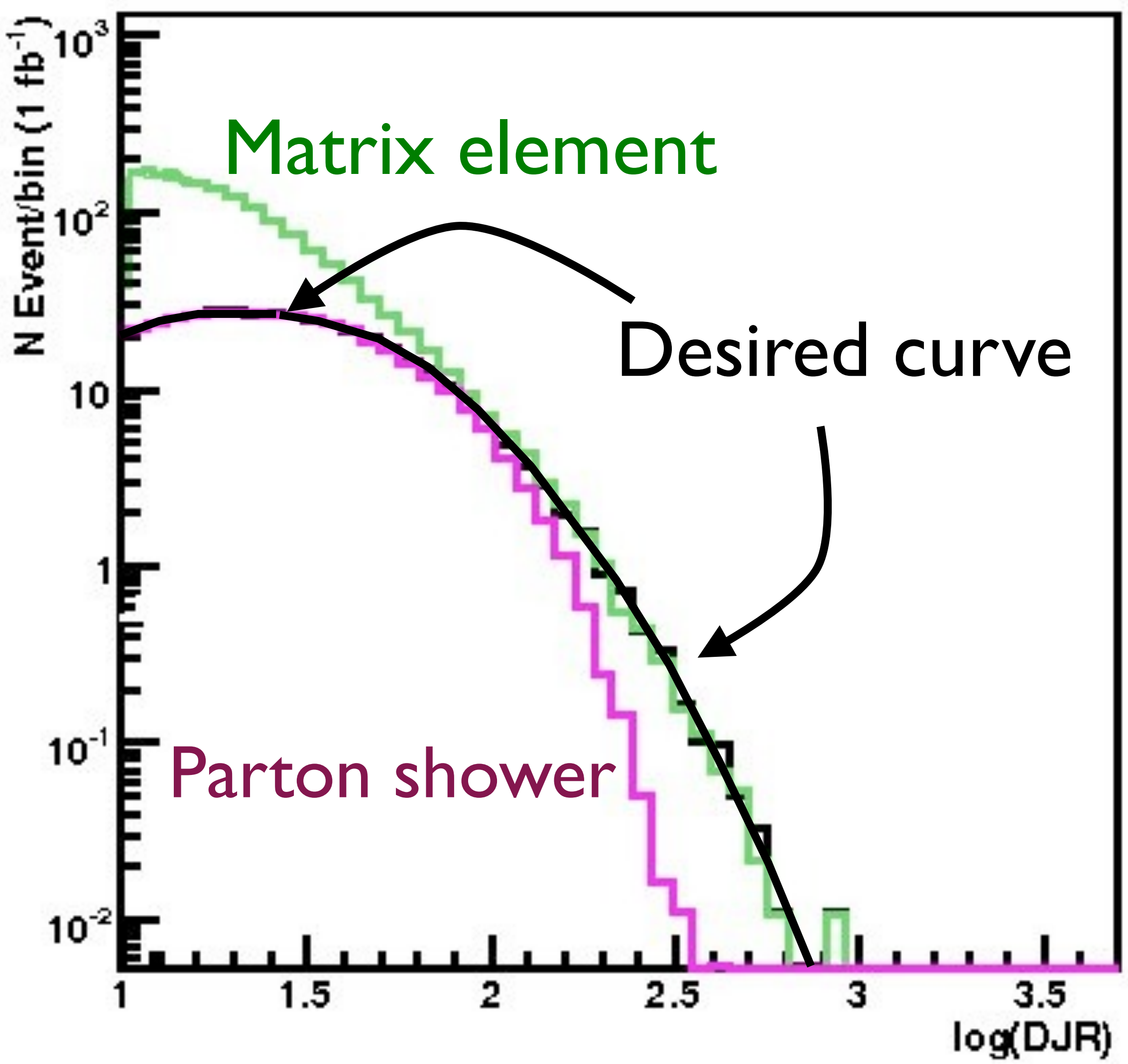
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Goal for ME-PS merging/matching

- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



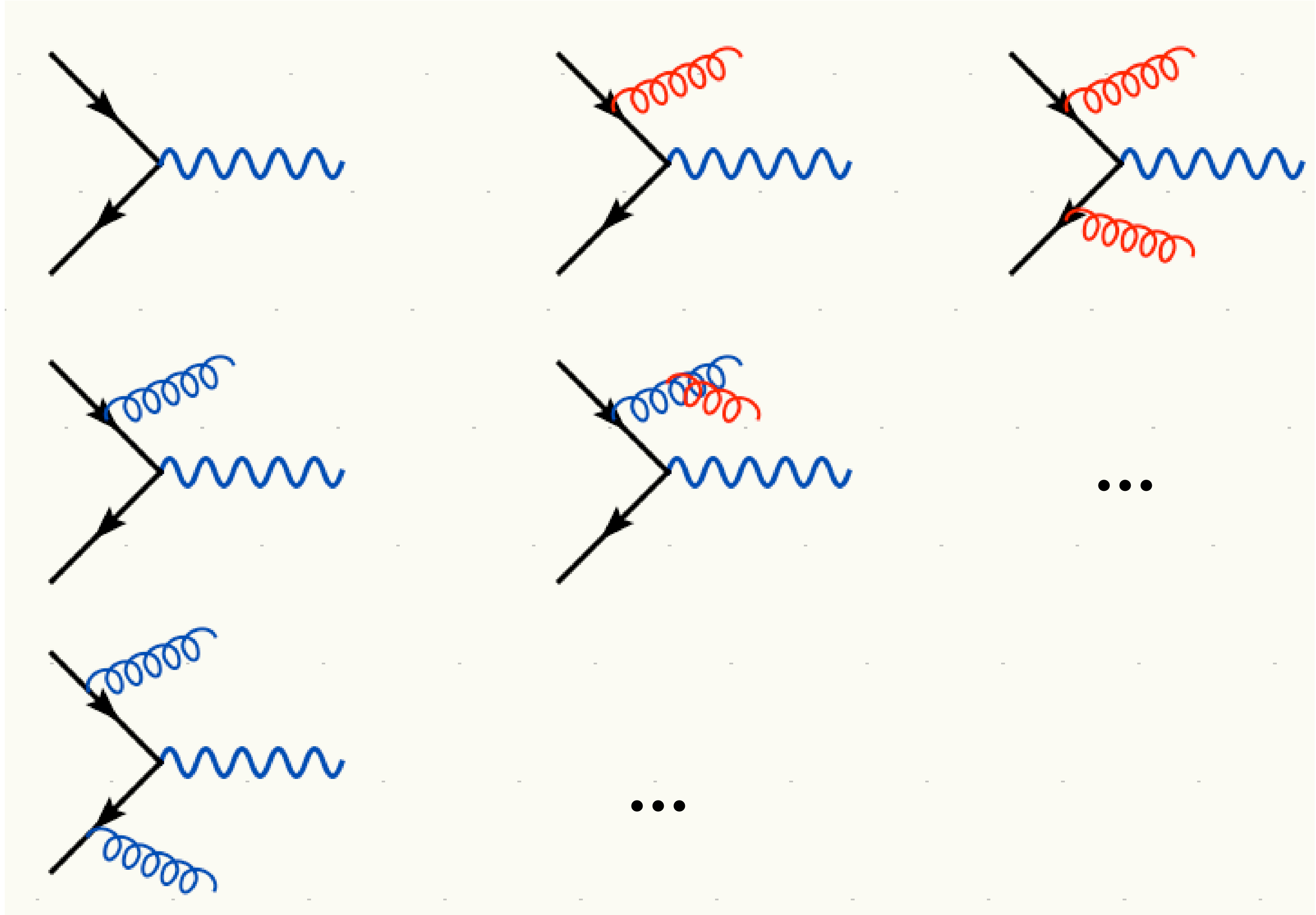
2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Merging ME with PS

[Mangano]
[Catani, Krauss, Kuhn, Webber]
[Lönnblad]

PS →

ME ↓

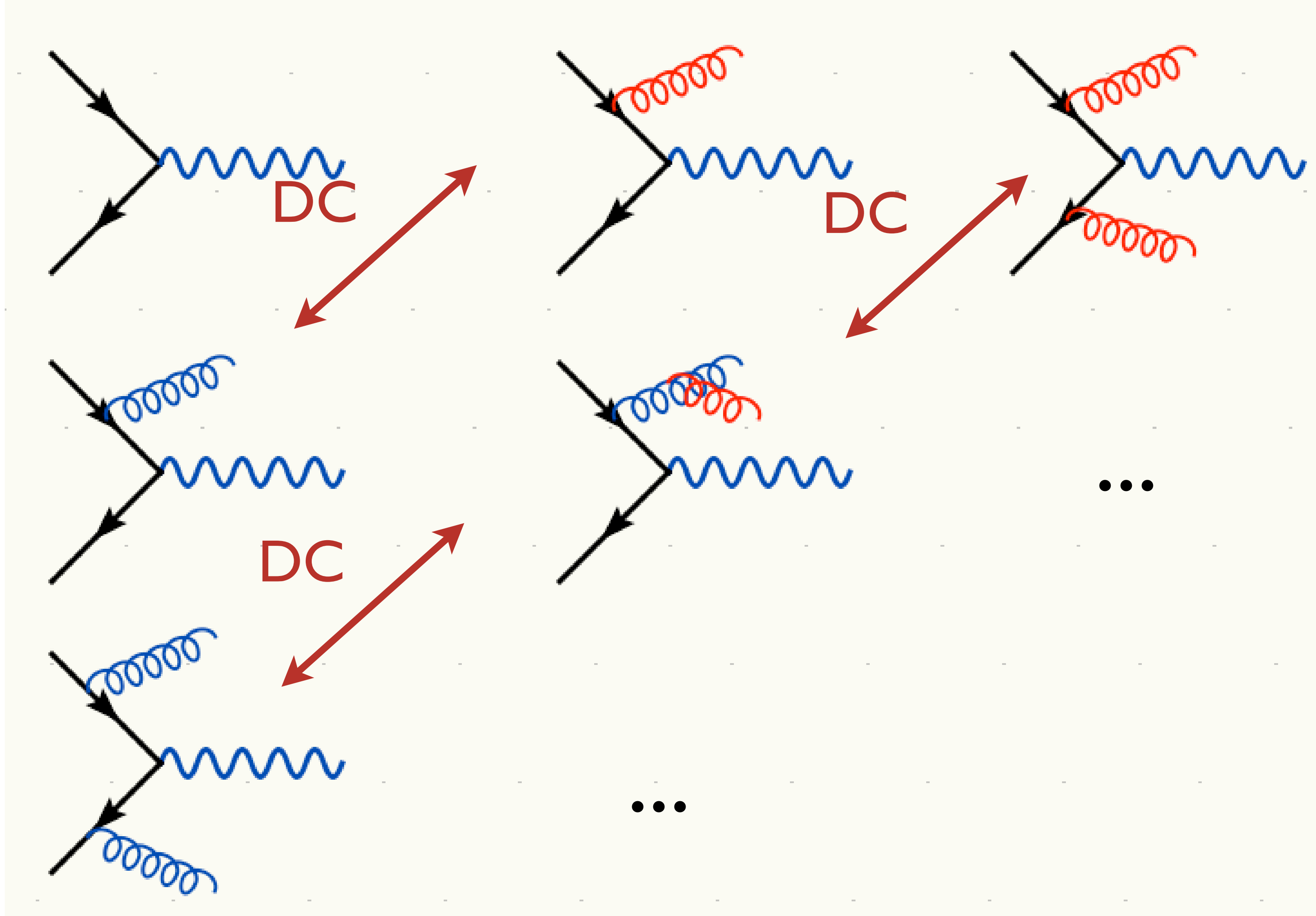


Merging ME with PS

[Mangano]
[Catani, Krauss, Kuhn, Webber]
[Lönnblad]

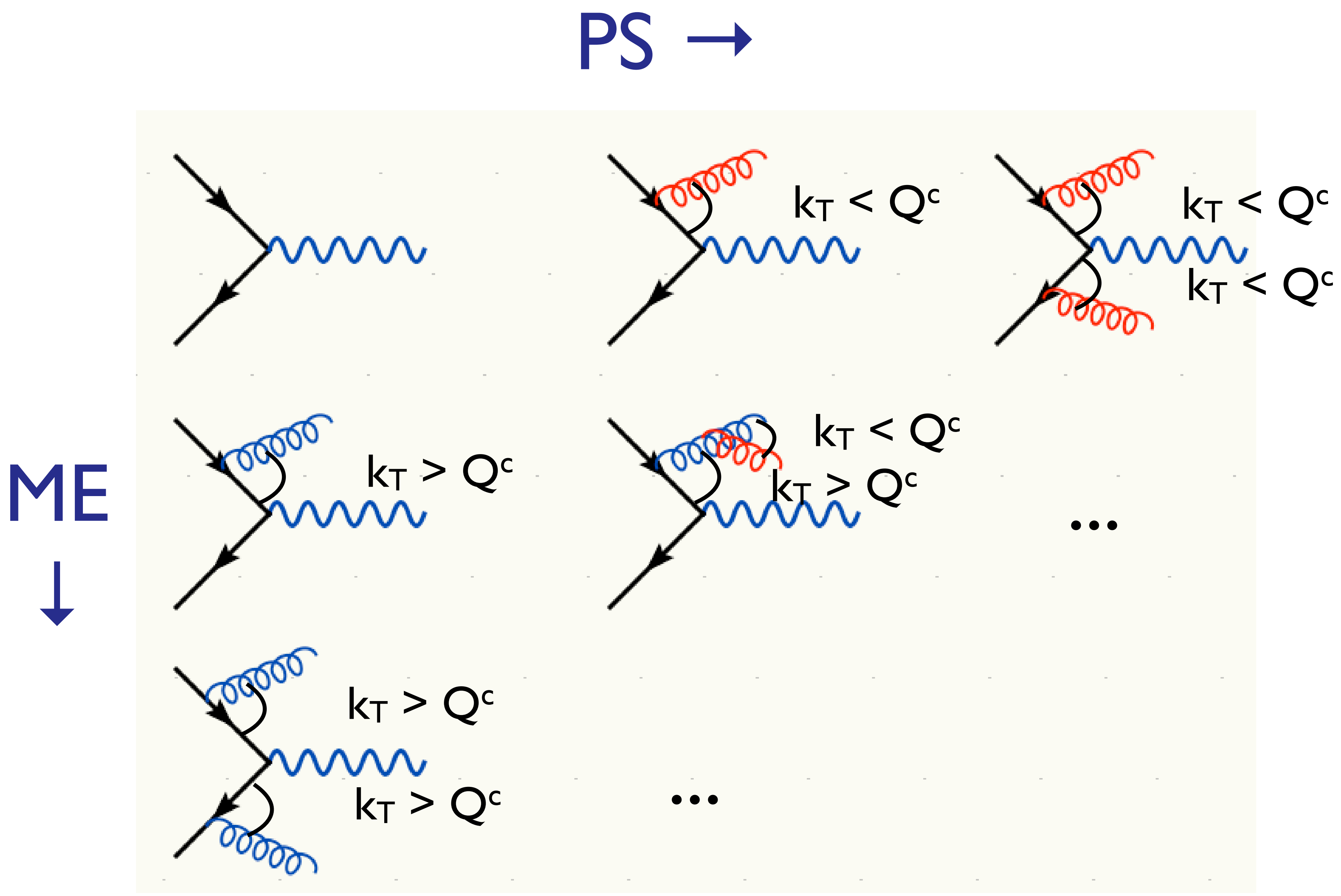
PS →

ME
↓



Merging ME with PS

[Mangano]
[Catani, Krauss, Kuhn, Webber]
[Lönnblad]

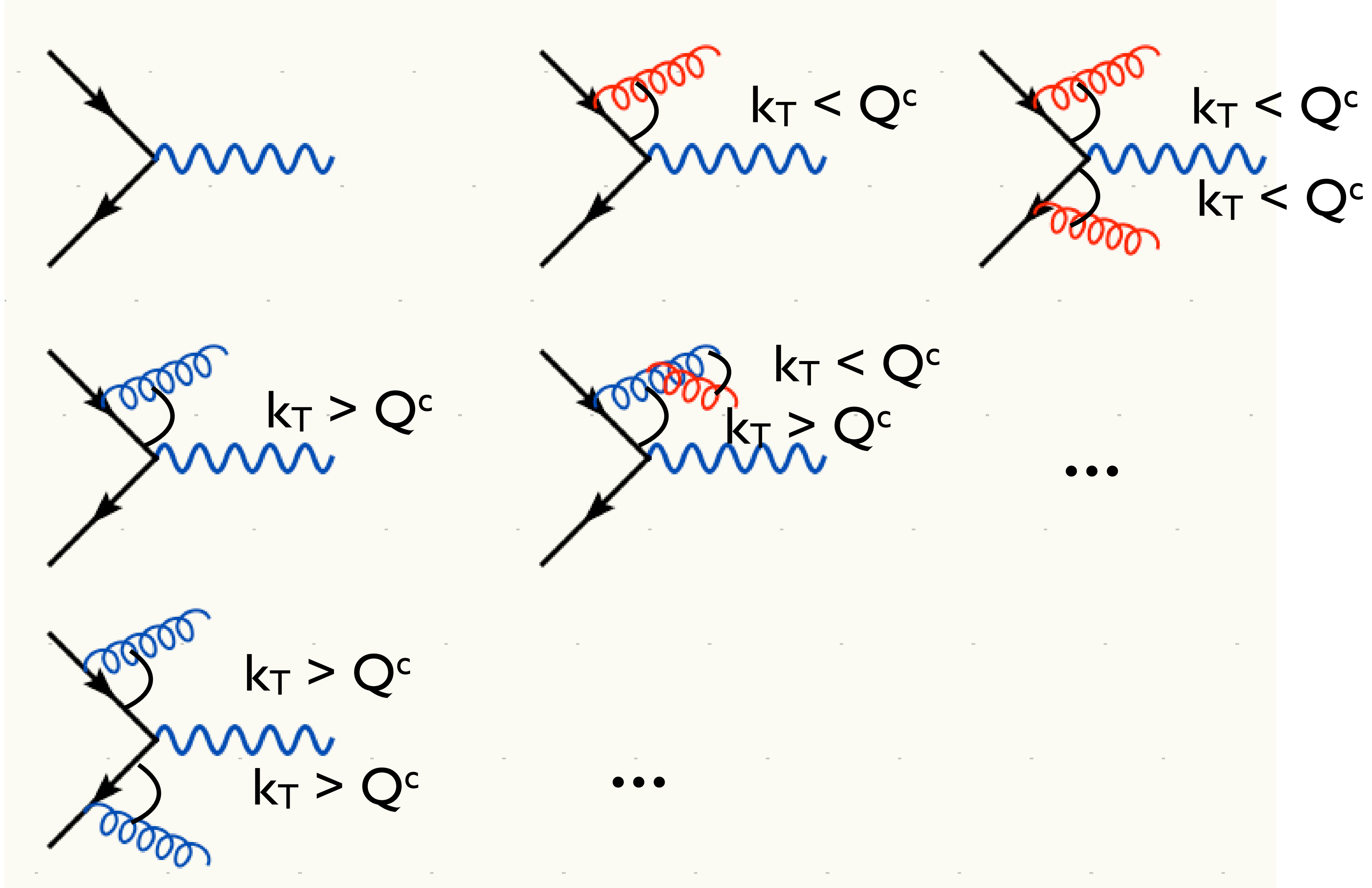


Merging ME with PS

[Mangano]
[Catani, Krauss, Kuhn, Webber]
[Lönnblad]

PS →

ME ↓

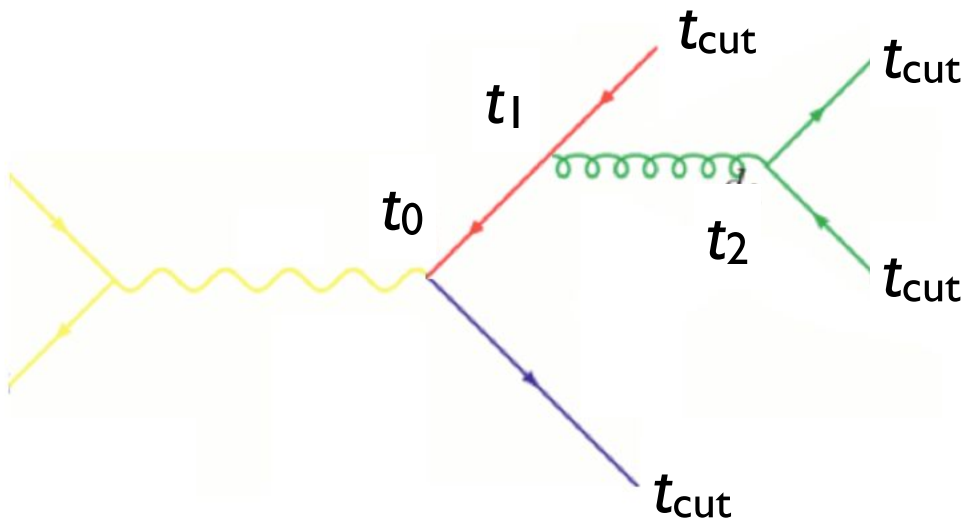


Double counting between ME and PS easily avoided using phase space cut between the two: PS below cutoff, ME above cutoff.

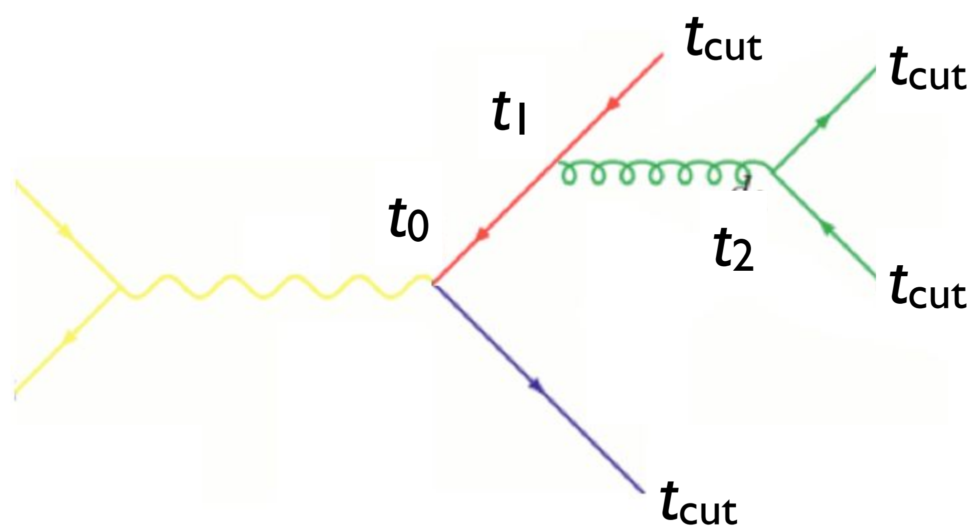
Merging ME with PS

- So double counting problem easily solved, but what about getting smooth distributions that are independent of the precise value of Q^c?
- Below cutoff, distribution is given by PS
  - need to make ME look like PS near cutoff
- Let's take another look at the PS!

Merging ME with PS

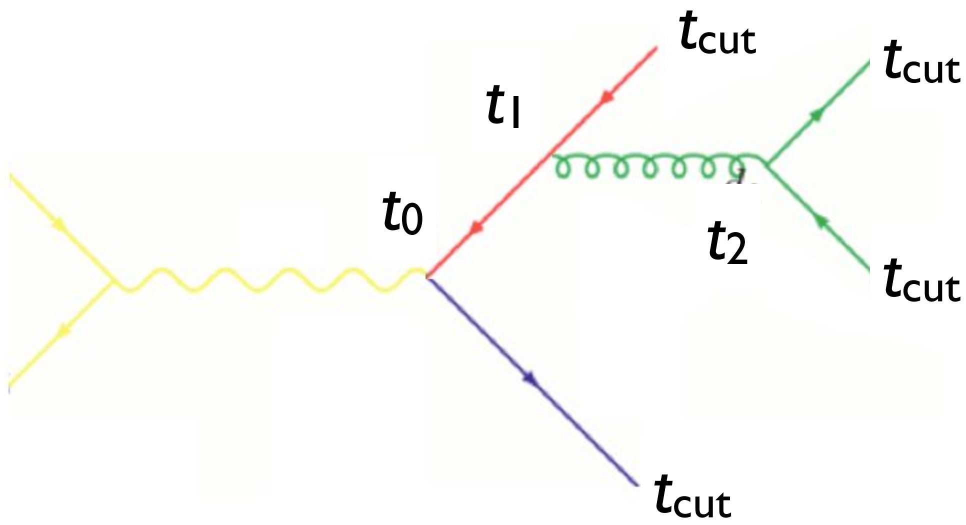


Merging ME with PS



- How does the PS generate the configuration above?

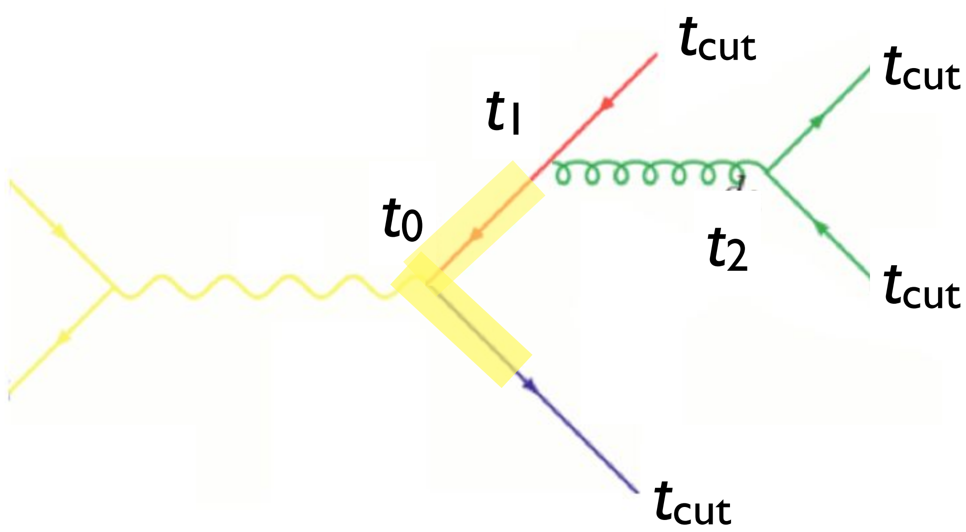
Merging ME with PS



- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

$$(\Delta_q(t_1, t_0))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z)$$

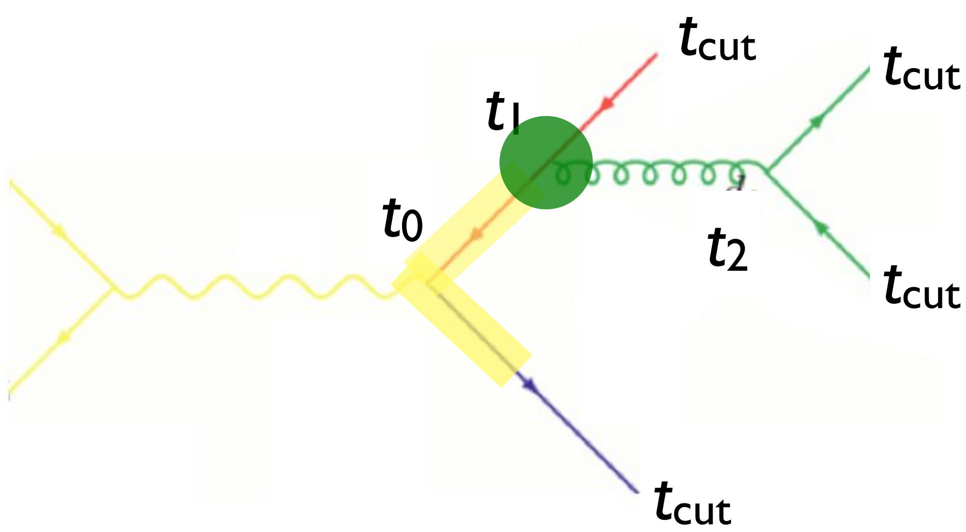
Merging ME with PS



- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

$$(\Delta_q(t_1, t_0))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z)$$

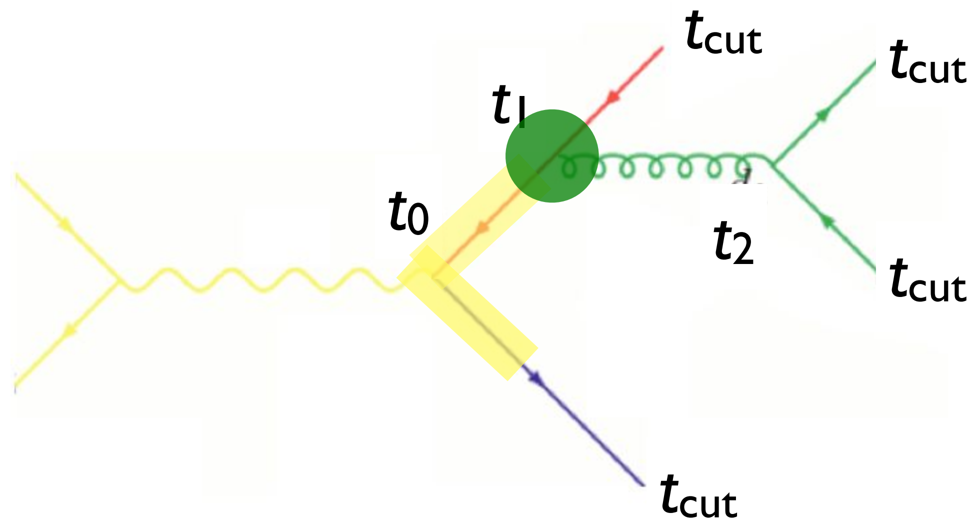
Merging ME with PS



- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

$$(\Delta_q(t_1, t_0))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z)$$

Merging ME with PS



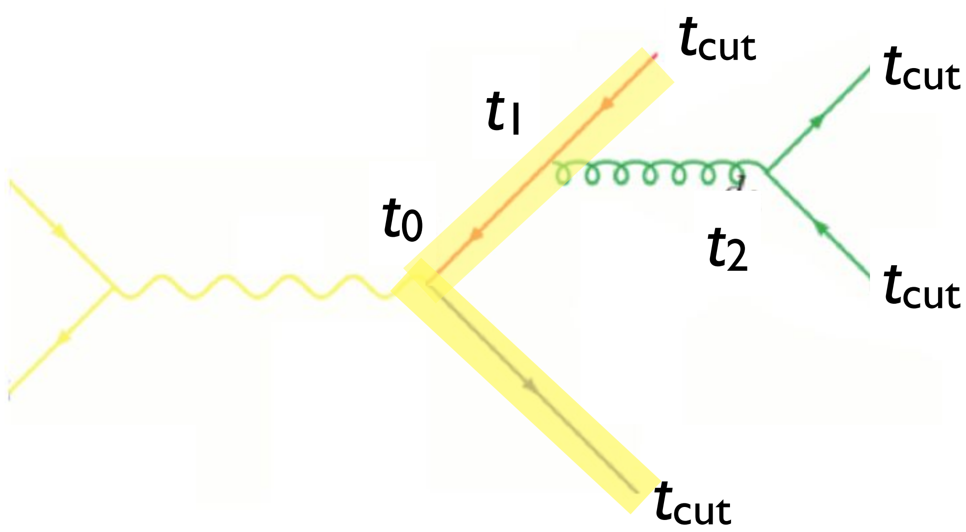
- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

$$(\Delta_q(t_1,t_0))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z)$$

and for the whole tree

$$(\Delta_q(t_{\mathrm{cut}},t_0))^2 \Delta_g(t_2,t_1) (\Delta_q(_{\mathrm{cut}},t_2))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z) \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$

Merging ME with PS



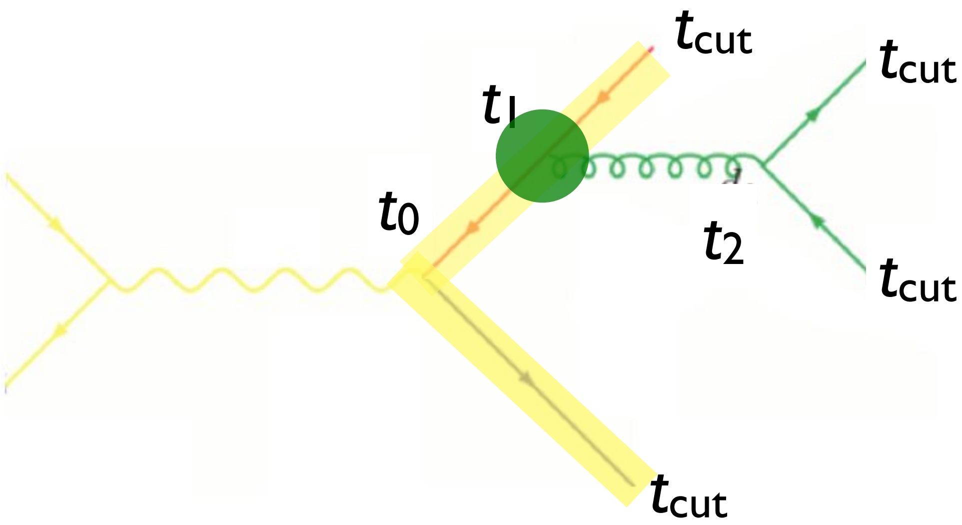
- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

$$(\Delta_q(t_1, t_0))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z)$$

and for the whole tree

$$(\Delta_q(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q({\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z) \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$

Merging ME with PS



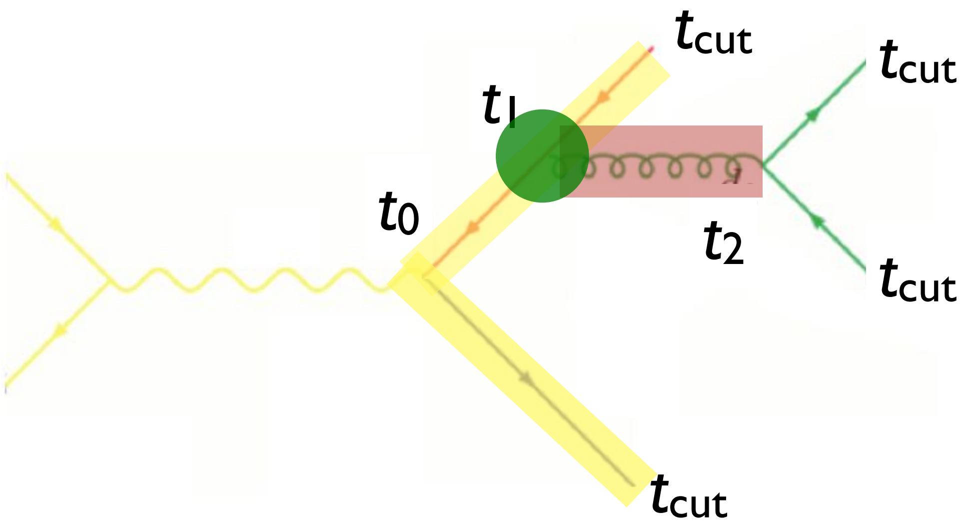
- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

$$(\Delta_q(t_1, t_0))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z)$$

and for the whole tree

$$(\Delta_q(t_{cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(_{cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z) \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$

Merging ME with PS



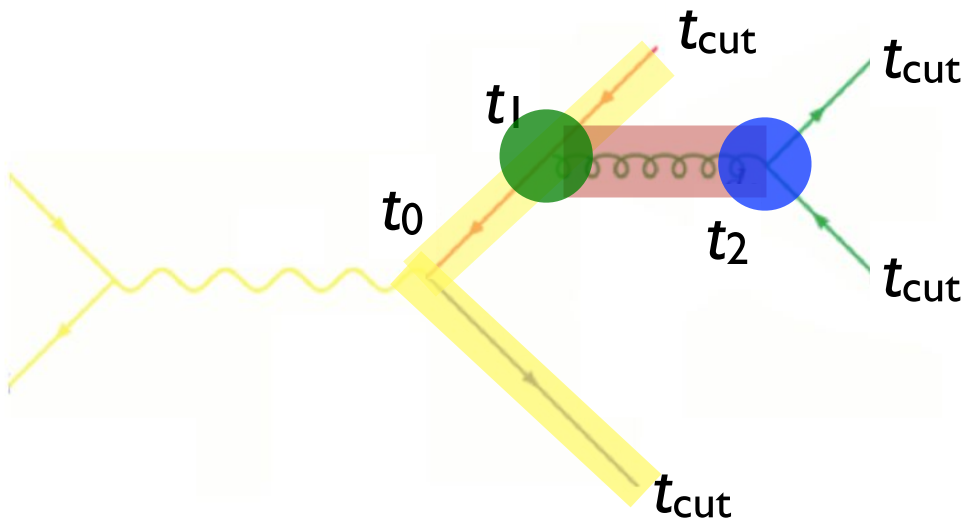
- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

$$(\Delta_q(t_1, t_0))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z)$$

and for the whole tree

$$(\Delta_q(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1) (\Delta_q({\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z) \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$

Merging ME with PS



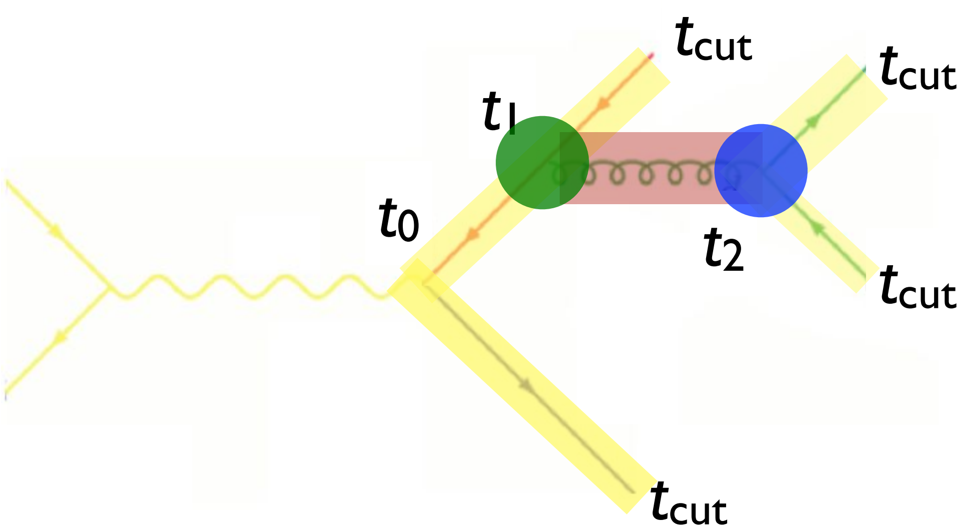
- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

$$(\Delta_q(t_1, t_0))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z)$$

and for the whole tree

$$(\Delta_q(t_{cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(_{cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z) \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$

Merging ME with PS



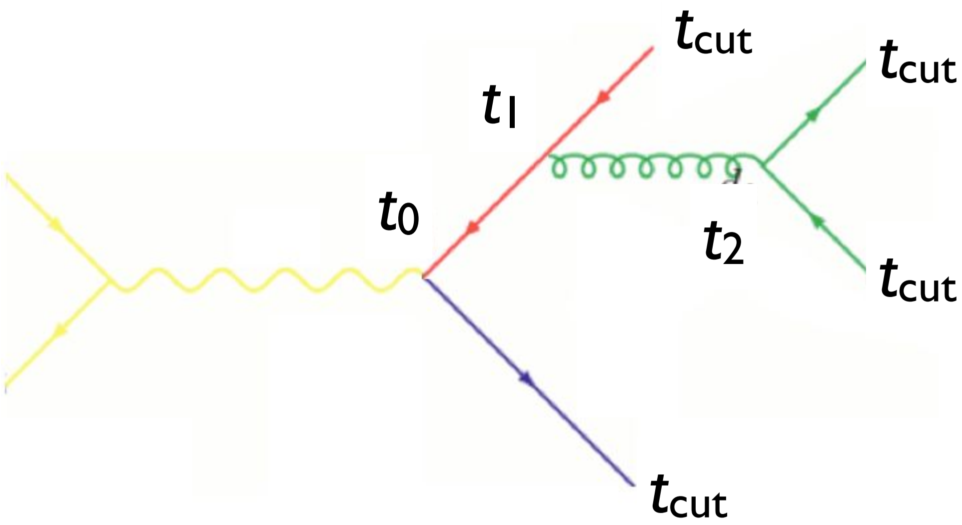
- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

$$(\Delta_q(t_1, t_0))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z)$$

and for the whole tree

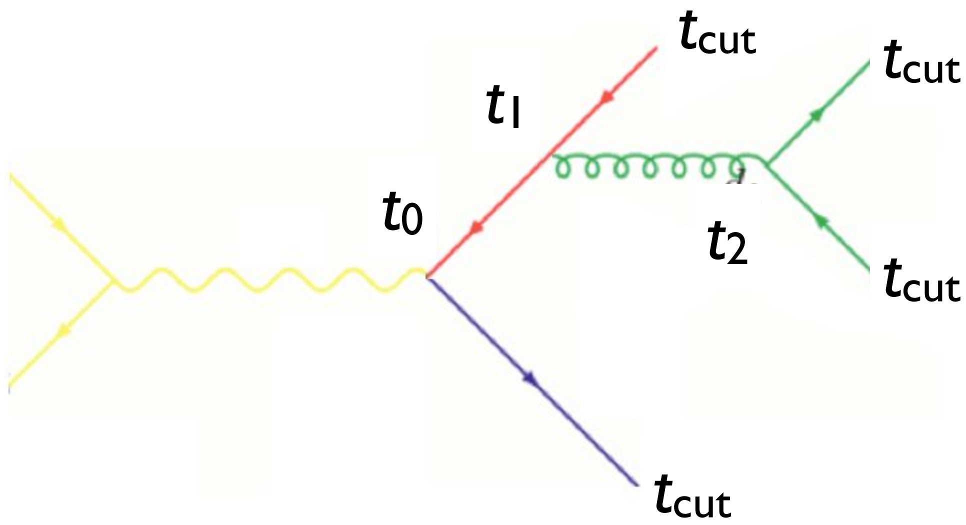
$$(\Delta_q(t_{\mathrm{cut}}, t_0))^2 \Delta_g(t_2, t_1) (\Delta_q(\mathrm{cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z) \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$

Merging ME with PS



$$(\Delta_q(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z) \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$

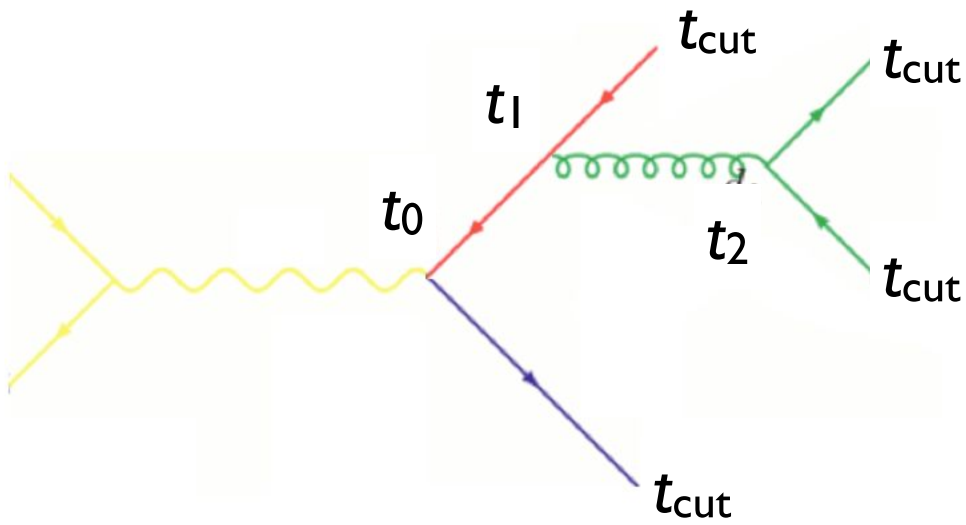
Merging ME with PS



$$(\Delta_q(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q({\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z) \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$

Corresponds to the matrix element
BUT with α_s evaluated at the scale of each splitting

Merging ME with PS

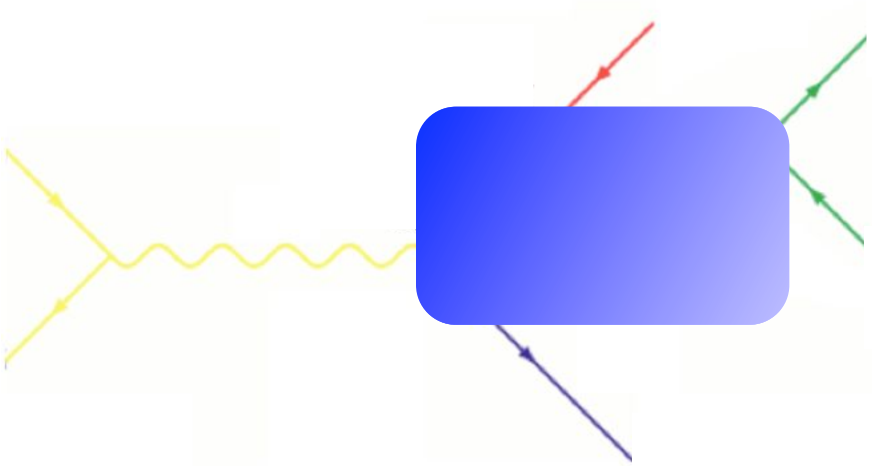


$$(\Delta_q(t_{\rm cut},t_0))^2\Delta_g(t_2,t_1)(\Delta_q(_{\rm cut},t_2))^2\frac{\alpha_s(t_1)}{2\pi}P_{gq}(z)\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$

Corresponds to the matrix element
BUT with α_s evaluated at the scale of each splitting

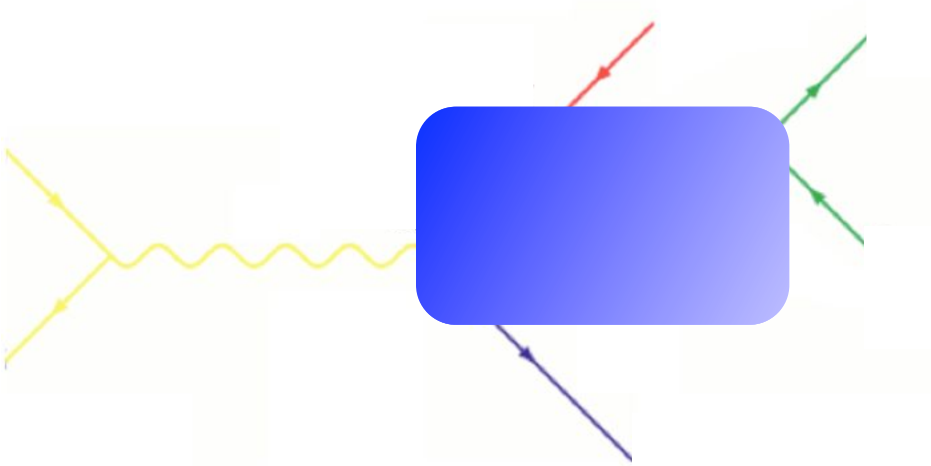
Sudakov suppression due to disallowing additional radiation above the scale t_{cut}

Merging ME with PS



$$|\mathcal{M}|^2(\hat{s}, p_3, p_4, \ldots)$$

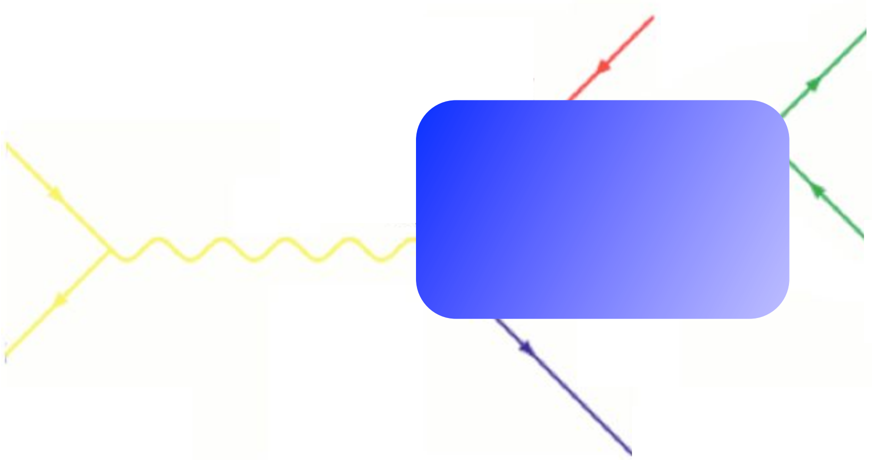
Merging ME with PS



$$|\mathcal{M}|^2(\hat{s}, p_3, p_4, \ldots)$$

- To get an equivalent treatment of the corresponding matrix element, do as follows:

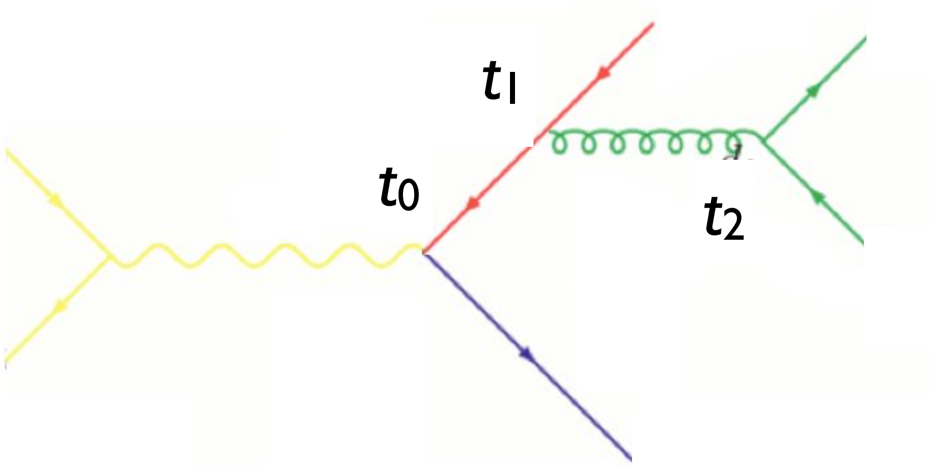
Merging ME with PS



$$|\mathcal{M}|^2(\hat{s}, p_3, p_4, \ldots)$$

- To get an equivalent treatment of the corresponding matrix element, do as follows:
  1. Cluster the event using some clustering algorithm
     - this gives us a corresponding "parton shower history"

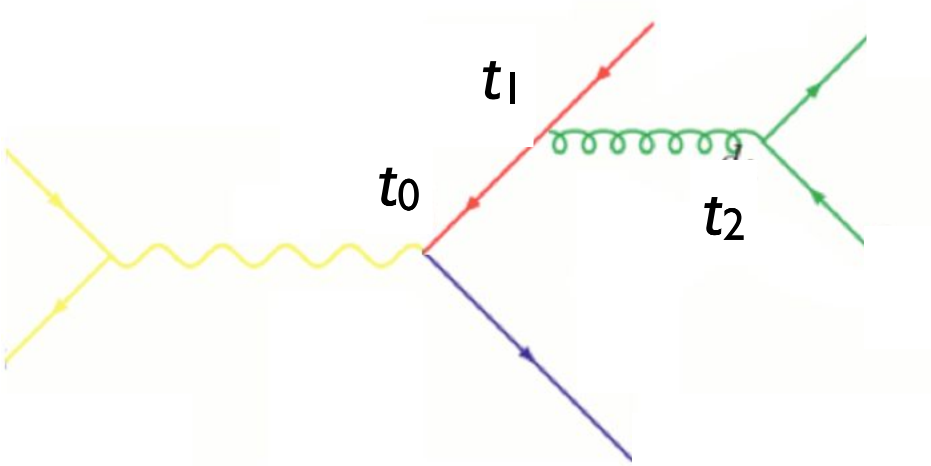
Merging ME with PS



$$|\mathcal{M}|^2(\hat{s}, p_3, p_4, \ldots)$$

- To get an equivalent treatment of the corresponding matrix element, do as follows:
  1. Cluster the event using some clustering algorithm
     - this gives us a corresponding "parton shower history"

Merging ME with PS



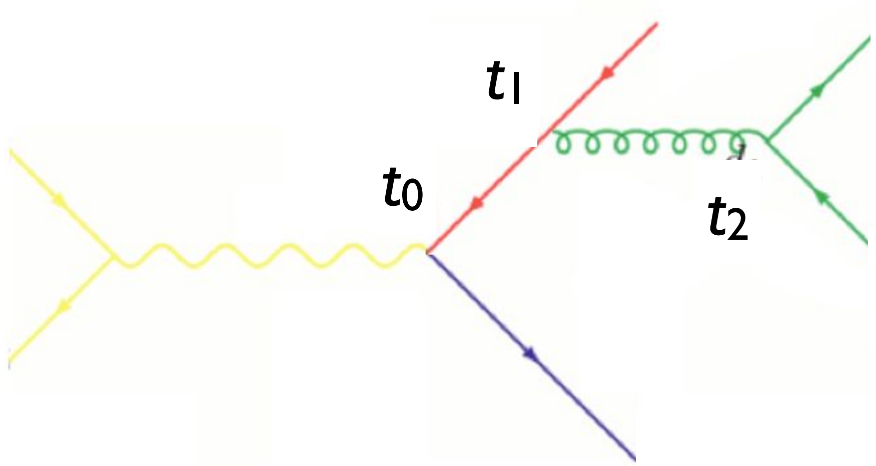
$$|\mathcal{M}|^2(\hat{s}, p_3, p_4, \ldots)$$

- To get an equivalent treatment of the corresponding matrix element, do as follows:

1. Cluster the event using some clustering algorithm
   - this gives us a corresponding "parton shower history"
2. Reweight α_s in each clustering vertex with the clustering scale

$$|\mathcal{M}|^2 \to |\mathcal{M}|^2 \frac{\alpha_s(t_1)}{\alpha_s(t_0)} \frac{\alpha_s(t_2)}{\alpha_s(t_0)}$$

Merging ME with PS

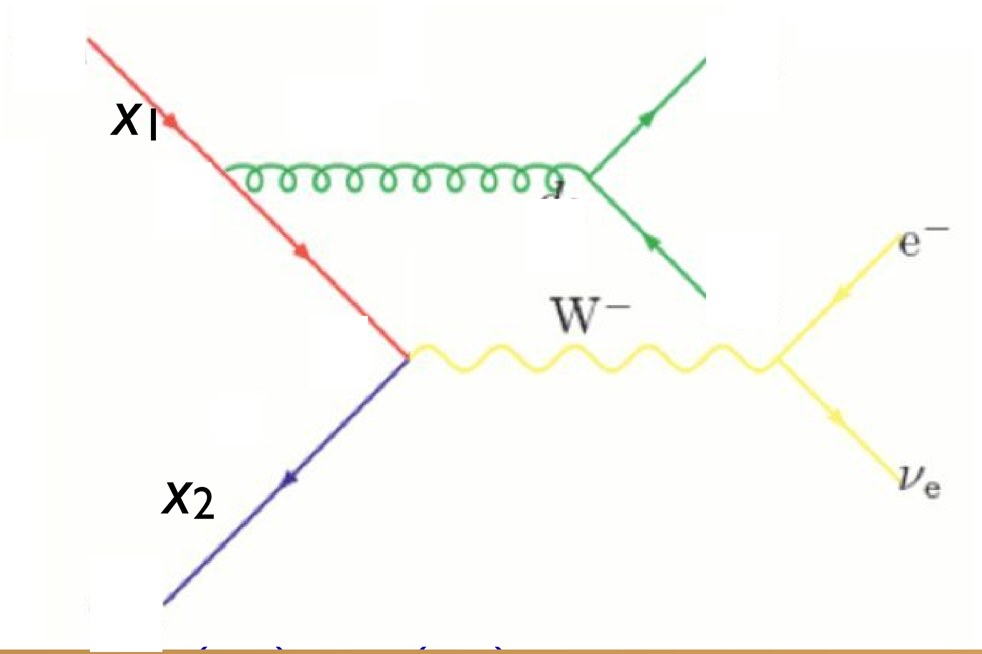


$$|\mathcal{M}|^2(\hat{s}, p_3, p_4, \ldots)$$

- To get an equivalent treatment of the corresponding matrix element, do as follows:

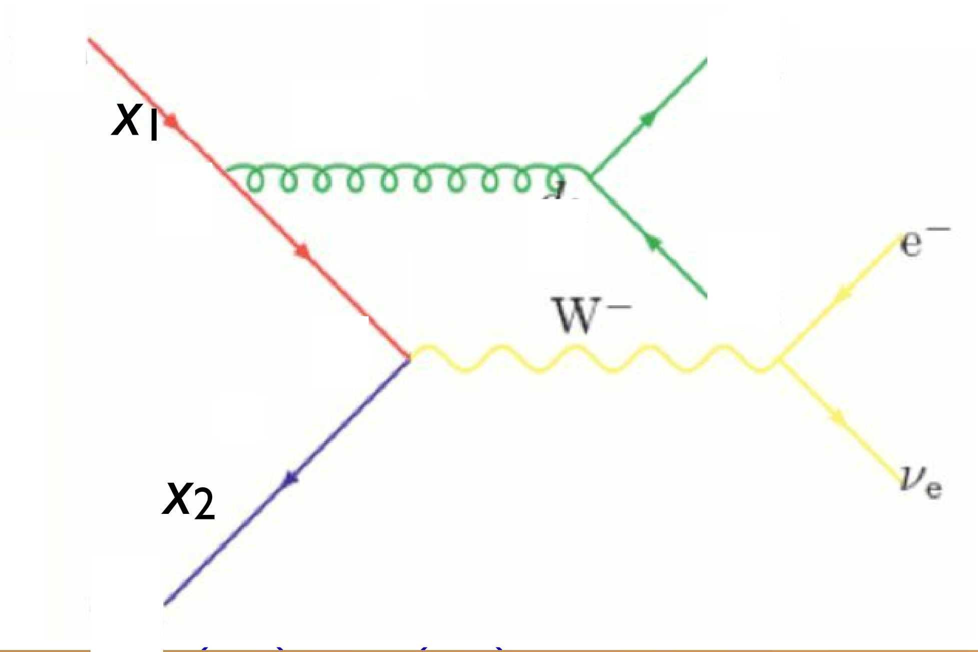
1. Cluster the event using some clustering algorithm
   - this gives us a corresponding "parton shower history"
2. Reweight α_s in each clustering vertex with the clustering scale
   $$|\mathcal{M}|^2 \to |\mathcal{M}|^2 \frac{\alpha_s(t_1)}{\alpha_s(t_0)} \frac{\alpha_s(t_2)}{\alpha_s(t_0)}$$
3. Use some algorithm to apply the equivalent Sudakov suppression $(\Delta_q(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q({\rm cut}, t_2))^2$

Matching for initial state radiation



Matching for initial state radiation

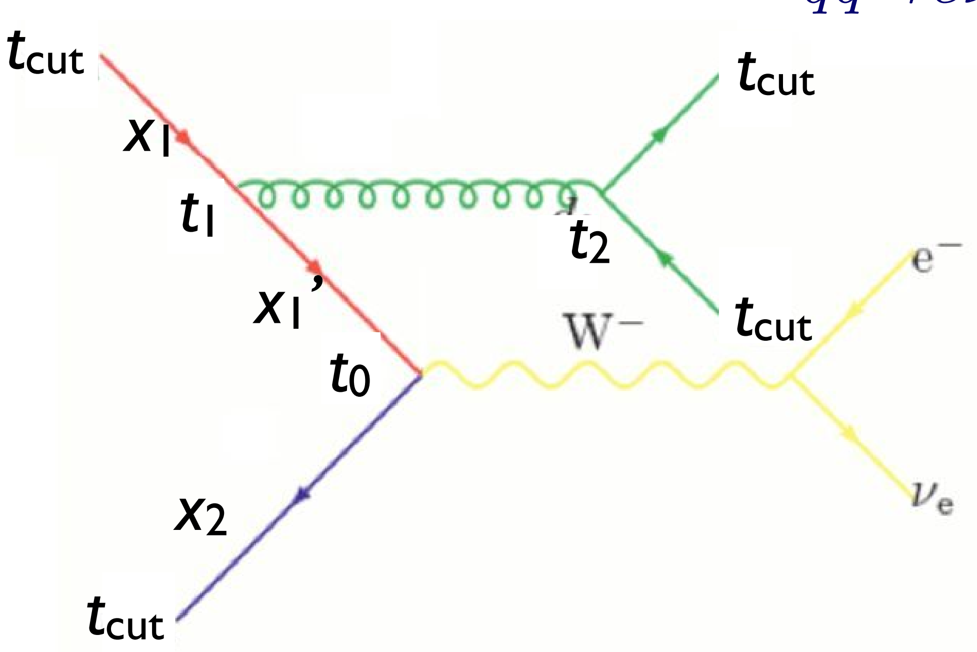
- We are of course not interested in e^+e^- but p-p(bar) - what happens for initial state radiation?



Matching for initial state radiation

- We are of course not interested in e+e- but p-p(bar) - what happens for initial state radiation?
- Let's do the same exercise as before:

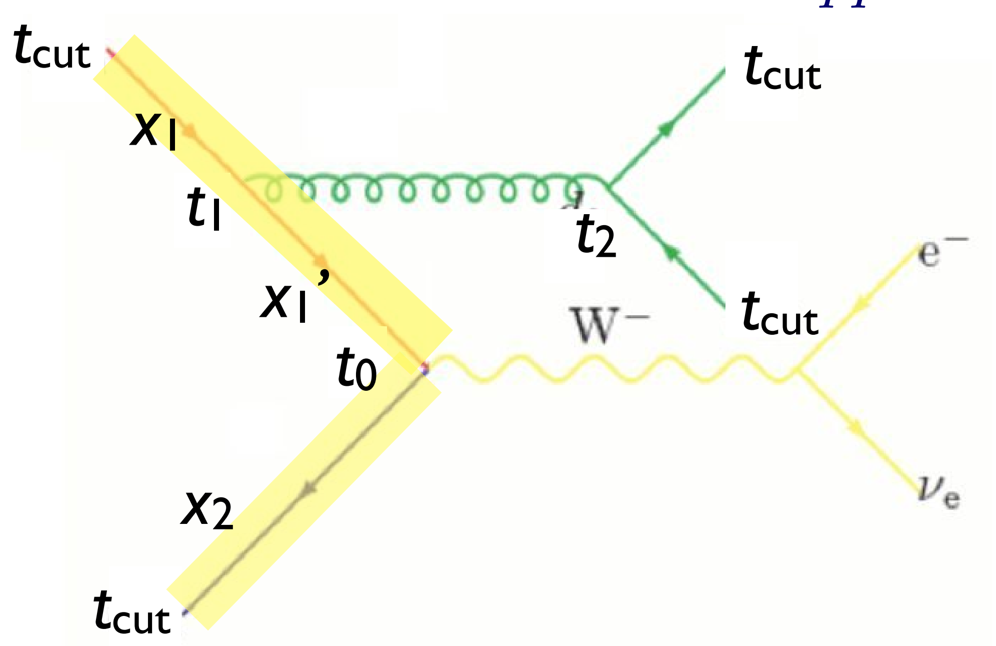
$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} \frac{P_{gq}(z)}{z} \frac{f_q(x_1, t_1)}{f_q(x_1', t_1)} \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$
$$\times \hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s}, \ldots) f_q(x_1', t_0) f_{\bar{q}}(x_2, t_0)$$



Matching for initial state radiation

- We are of course not interested in e^+e^- but p-p(bar) - what happens for initial state radiation?
- Let's do the same exercise as before:

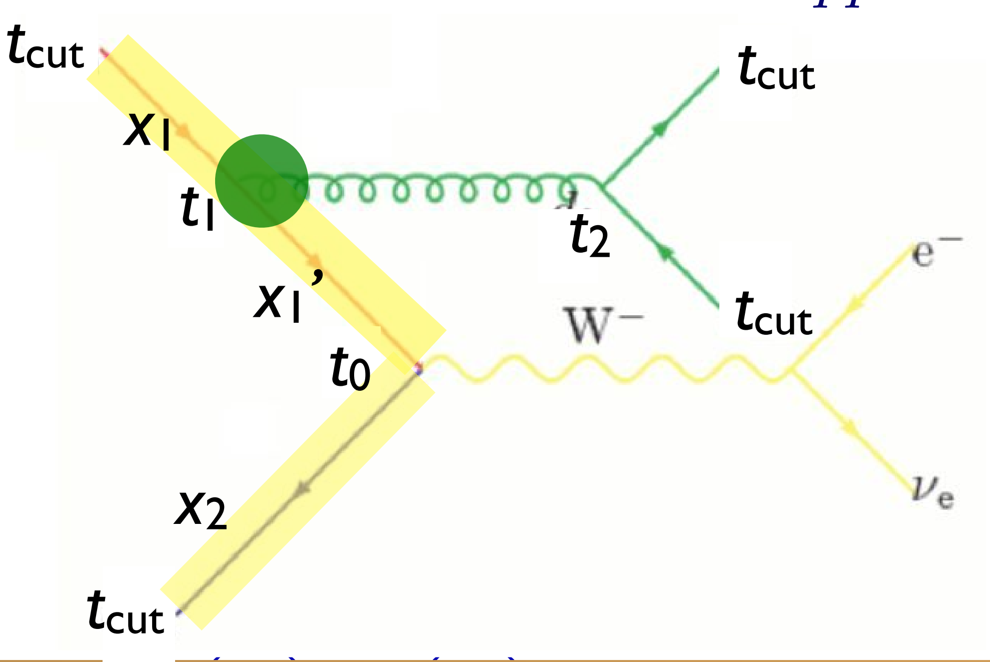
$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1, t_1)}{f_q(x_1', t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z') \times \hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s}, \ldots) f_q(x_1', t_0) f_{\bar{q}}(x_2, t_0)$$



Matching for initial state radiation

- We are of course not interested in e$^+$e$^-$ but p-p(bar) - what happens for initial state radiation?
- Let's do the same exercise as before:

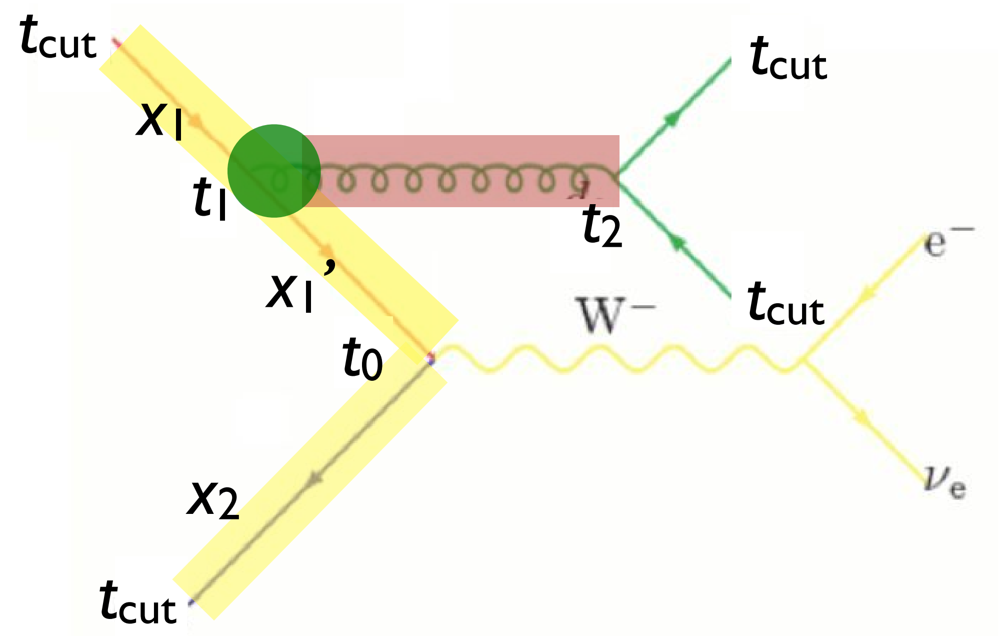
$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1, t_1)}{f_q(x_1', t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$
$$\times \hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s}, \ldots) f_q(x_1', t_0) f_{\bar{q}}(x_2, t_0)$$



Matching for initial state radiation

- We are of course not interested in e⁺e⁻ but p-p(bar) - what happens for initial state radiation?
- Let's do the same exercise as before:

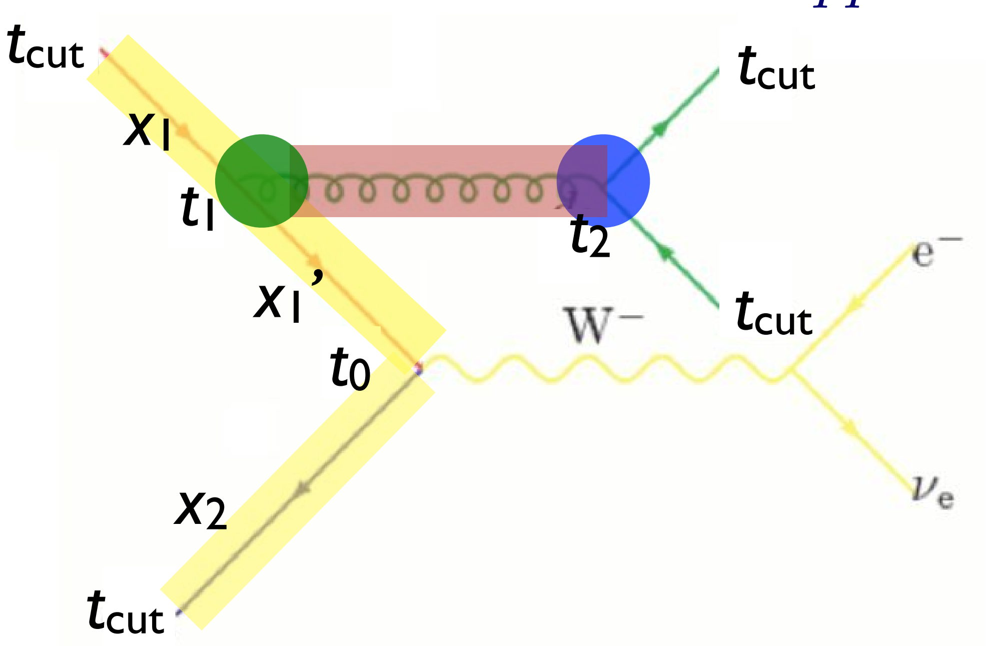
$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} \frac{P_{gq}(z)}{z} \frac{f_q(x_1, t_1)}{f_q(x'_1, t_1)} \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$
$$\times \hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s}, \ldots) f_q(x'_1, t_0) f_{\bar{q}}(x_2, t_0)$$



Matching for initial state radiation

- We are of course not interested in e+e- but p-p(bar) - what happens for initial state radiation?
- Let's do the same exercise as before:

$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1,t_1)}{f_q(x_1',t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$
$$\times\hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s},\ldots)f_q(x_1',t_0)f_{\bar{q}}(x_2,t_0)$$

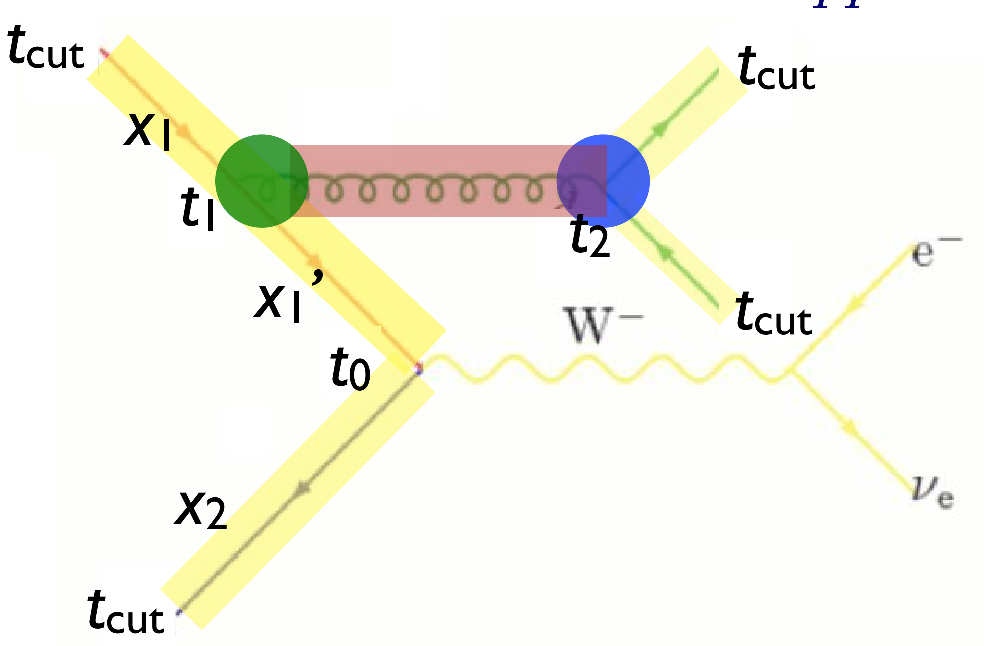


Matching for initial state radiation

- We are of course not interested in e^+e^- but p-p(bar) - what happens for initial state radiation?
- Let's do the same exercise as before:

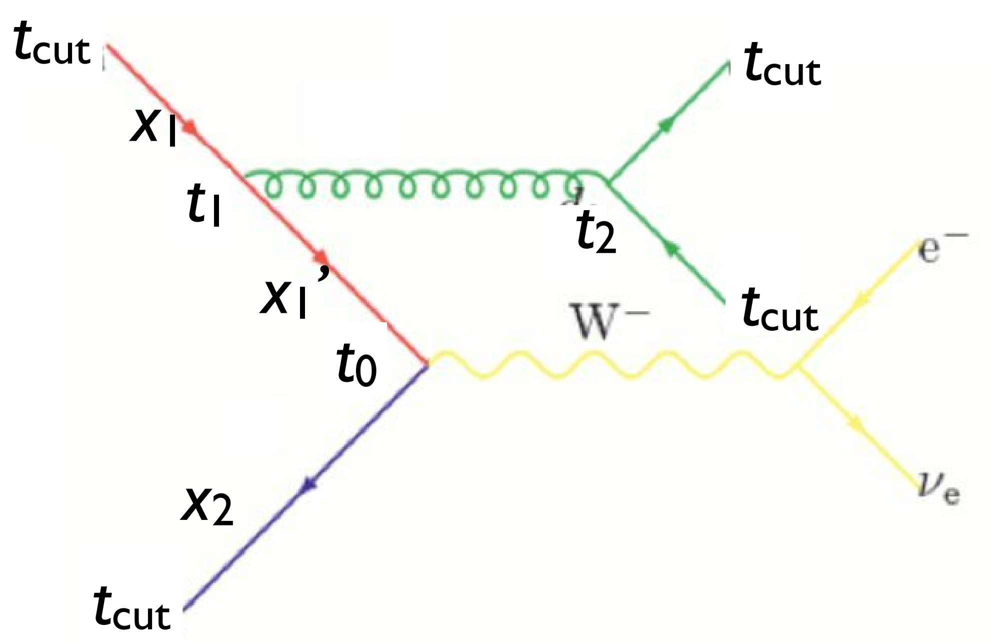
$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1, t_1)}{f_q(x_1', t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$

$$\times \hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s}, \ldots) f_q(x_1', t_0) f_{\bar{q}}(x_2, t_0)$$



Matching for initial state radiation

$$(\Delta_{Iq}(t_{\rm cut},t_0))^2\Delta_g(t_2,t_1)(\Delta_q(t_{\rm cut},t_2))^2\frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1,t_1)}{f_q(x_1',t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$
$$\times\hat\sigma_{q\bar q\to e\nu}(\hat s,\ldots)f_q(x_1',t_0)f_{\bar q}(x_2,t_0)$$

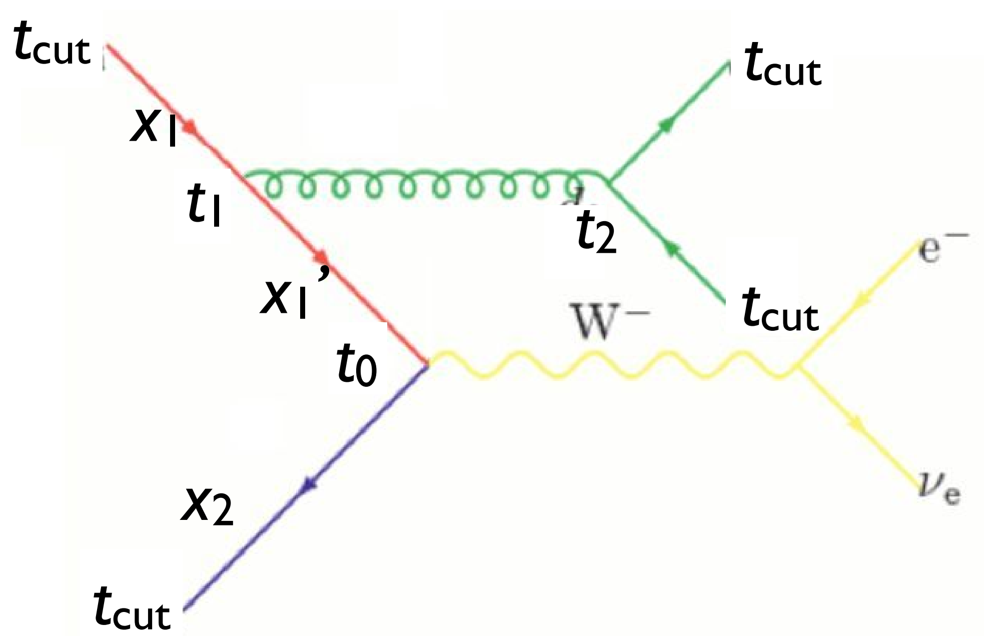


Matching for initial state radiation

$$(\Delta_{Iq}(t_{\rm cut},t_0))^2\Delta_g(t_2,t_1)(\Delta_q(t_{\rm cut},t_2))^2\frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1,t_1)}{f_q(x_1',t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$

$$\times\hat\sigma_{q\bar q\to e\nu}(\hat s,\ldots)f_q(x_1',t_0)f_{\bar q}(x_2,t_0)$$

ME with α_s evaluated at the scale of each splitting



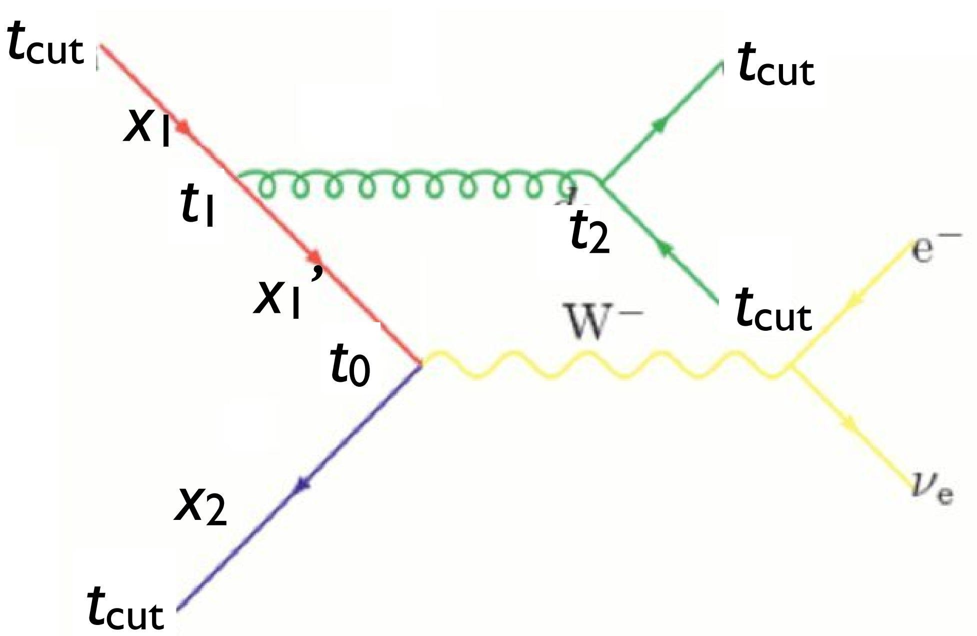
Matching for initial state radiation

$$(\Delta_{Iq}(t_{\rm cut},t_0))^2\Delta_g(t_2,t_1)(\Delta_q(t_{\rm cut},t_2))^2\frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1,t_1)}{f_q(x_1',t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$

$$\times\hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s},\ldots)f_q(x_1',t_0)f_{\bar{q}}(x_2,t_0)$$

ME with α_s evaluated at the scale of each splitting

PDF reweighting



Matching for initial state radiation

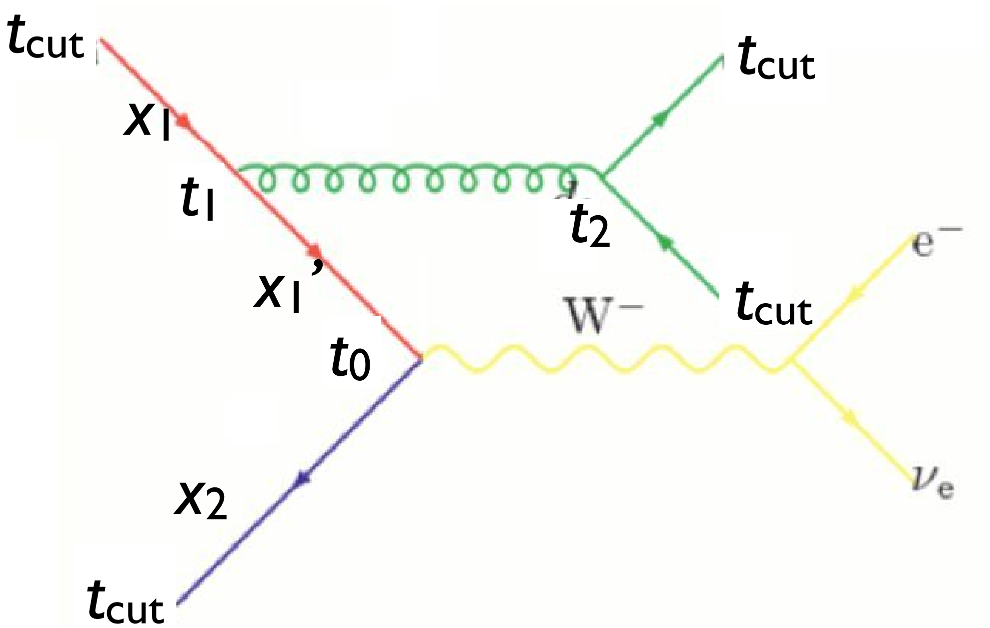
$$(\Delta_{Iq}(t_{\rm cut},t_0))^2\Delta_g(t_2,t_1)(\Delta_q(t_{\rm cut},t_2))^2\frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1,t_1)}{f_q(x_1',t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$

$$\times\hat\sigma_{q\bar q\to e\nu}(\hat s,\ldots)f_q(x_1',t_0)f_{\bar q}(x_2,t_0)$$

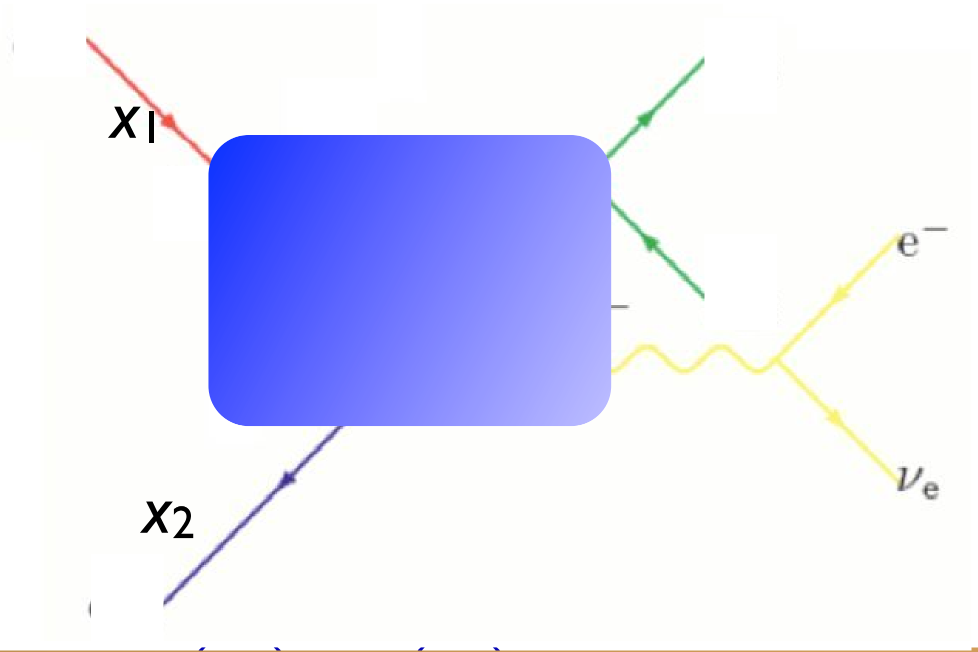
ME with α_s evaluated at the scale of each splitting

PDF reweighting

Sudakov suppression due to non-branching above scale $t_{\rm cut}$

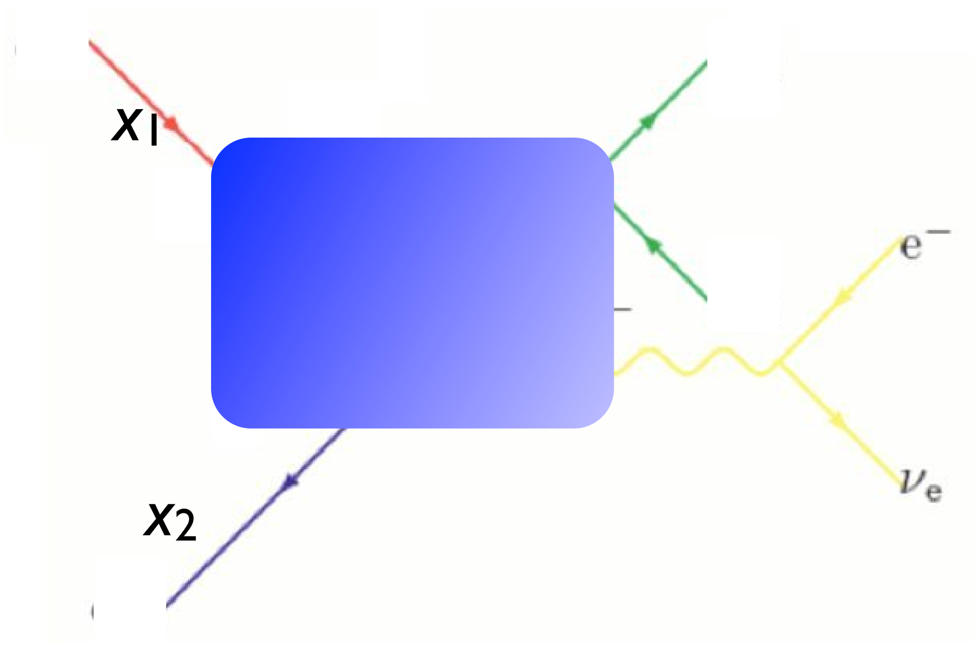


Matching for initial state radiation



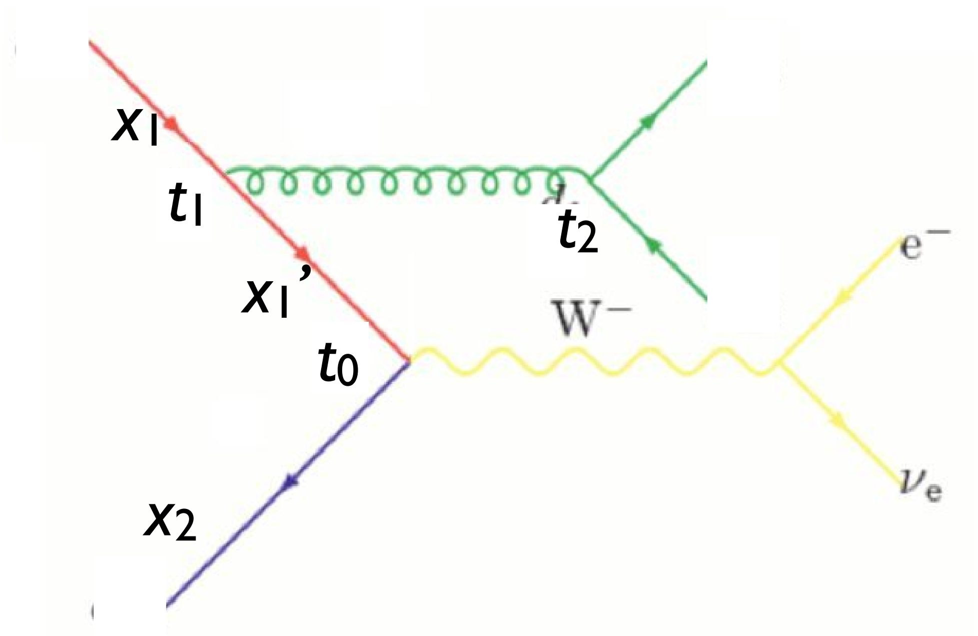
Matching for initial state radiation

- Again, use a clustering scheme to get a parton shower history



Matching for initial state radiation

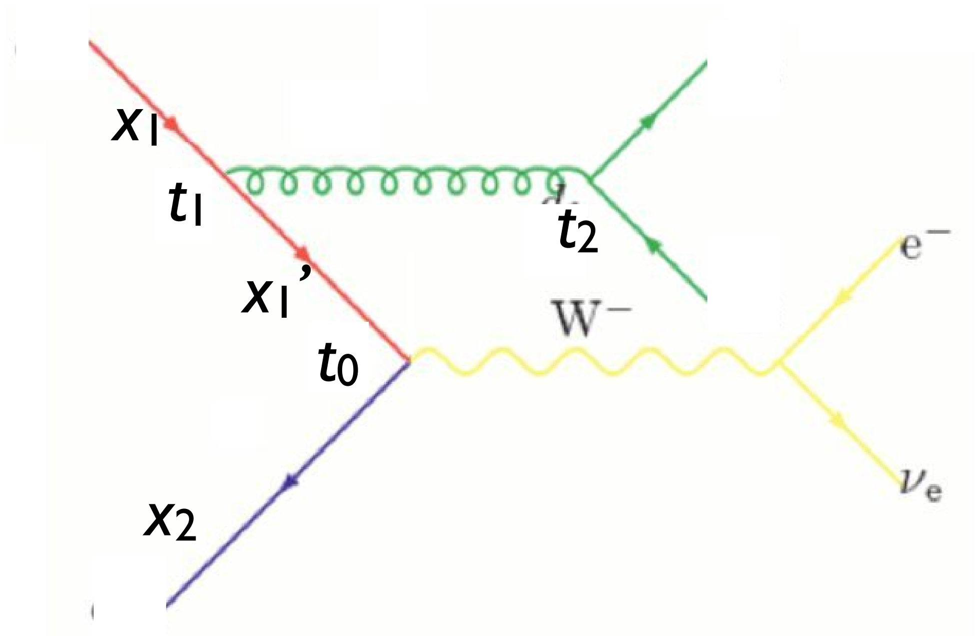
- Again, use a clustering scheme to get a parton shower history



Matching for initial state radiation

- Again, use a clustering scheme to get a parton shower history
- Now, reweight both due to α_s and PDF

$$|\mathcal{M}|^2 \to |\mathcal{M}|^2 \frac{\alpha_s(t_1)}{\alpha_s(t_0)} \frac{\alpha_s(t_2)}{\alpha_s(t_0)} \frac{f_q(x_1', t_0)}{f_q(x_1', t_1)}$$

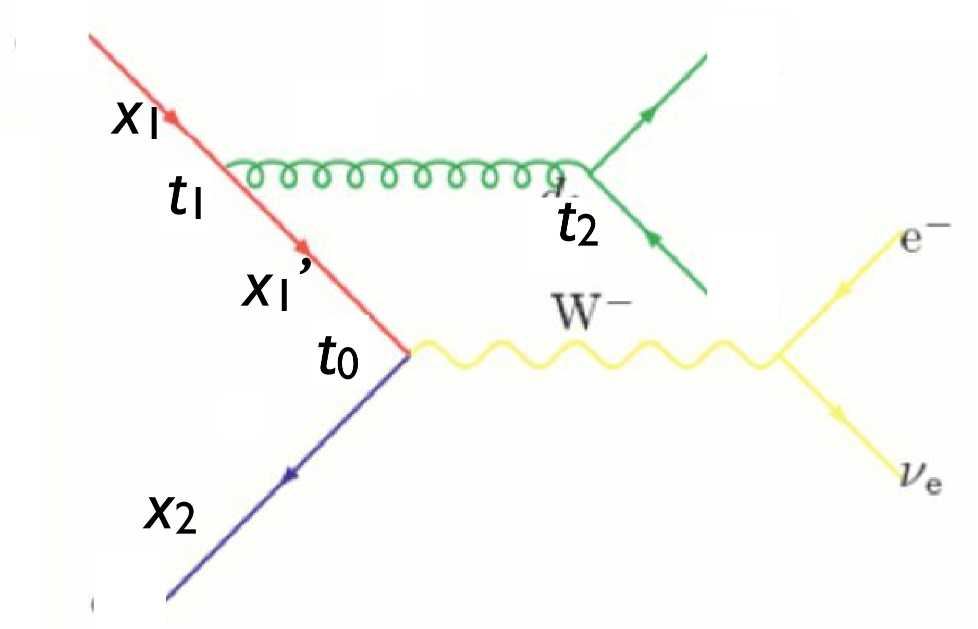


Matching for initial state radiation

- Again, use a clustering scheme to get a parton shower history
- Now, reweight both due to α_s and PDF

$$|\mathcal{M}|^2 \to |\mathcal{M}|^2 \frac{\alpha_s(t_1)}{\alpha_s(t_0)} \frac{\alpha_s(t_2)}{\alpha_s(t_0)} \frac{f_q(x_1', t_0)}{f_q(x_1', t_1)}$$

- Remember to use first clustering scale on each side for PDF scale: $\mathcal{P}_{\text{event}} = \hat{\sigma}(x_1, x_2, p_3, p_4, \dots) f_q(x_1, t_1) f_{\bar{q}}(x_2, t_0)$



K_T clustering schemes

The default clustering scheme used (in MG/Sherpa/AlpGen) to determine the parton shower history is the Durham k_T scheme. For e^+e^-:

$$k_{Tij}^2 = 2\min(E_i^2, E_j^2)(1-\cos\theta_{ij})$$

and for hadron collisions, the minimum of:

$$k_{Ti\mathrm{beam}}^2 = m_i^2 + p_{Ti}^2 = (E_i + p_{zi})(E_i - p_{zi})$$

and

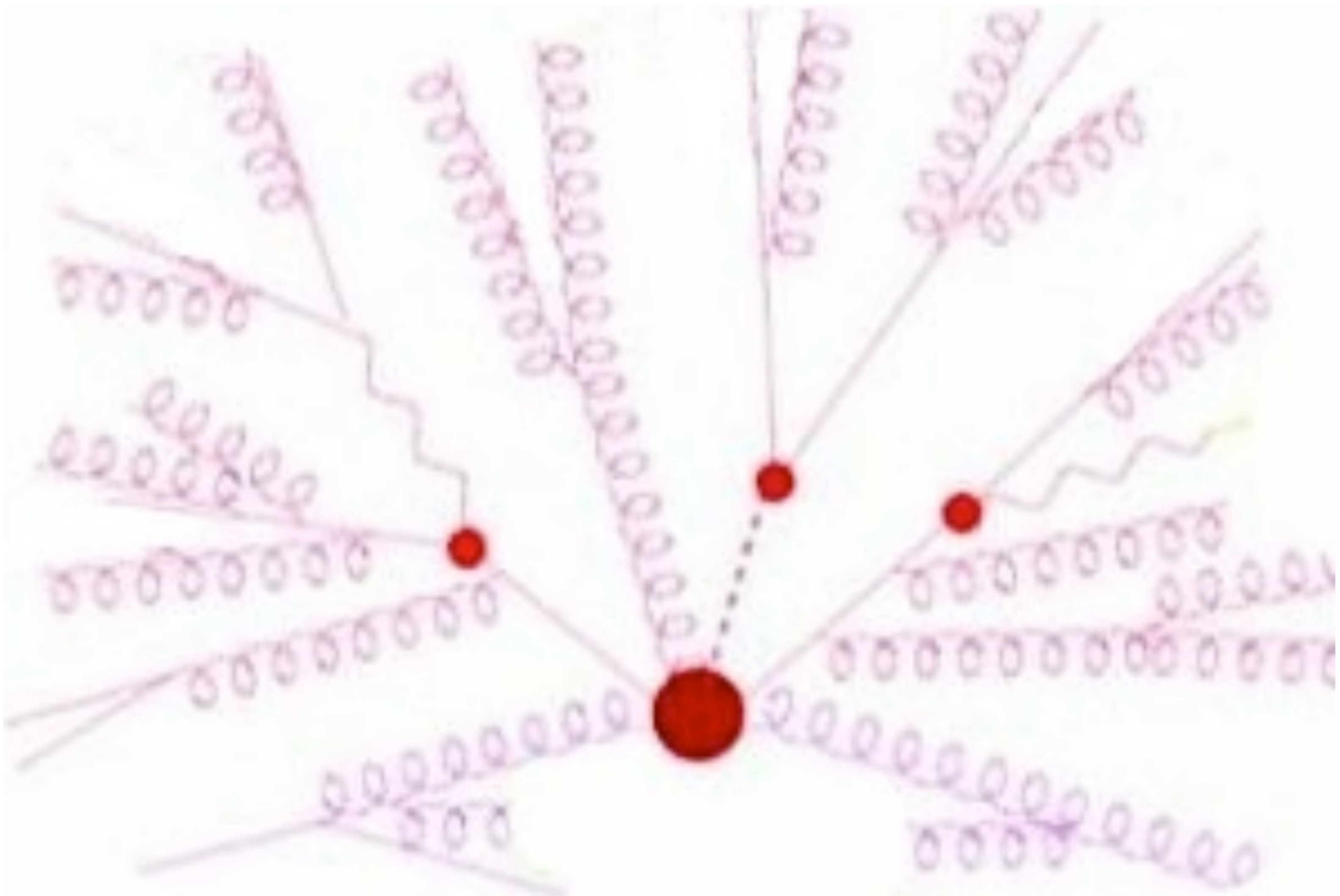
$$k_{Tij}^2 = \max(m_i^2, m_2^2) + \min(p_{Ti}^2, p_{Tj}^2)R_{ij}$$

with

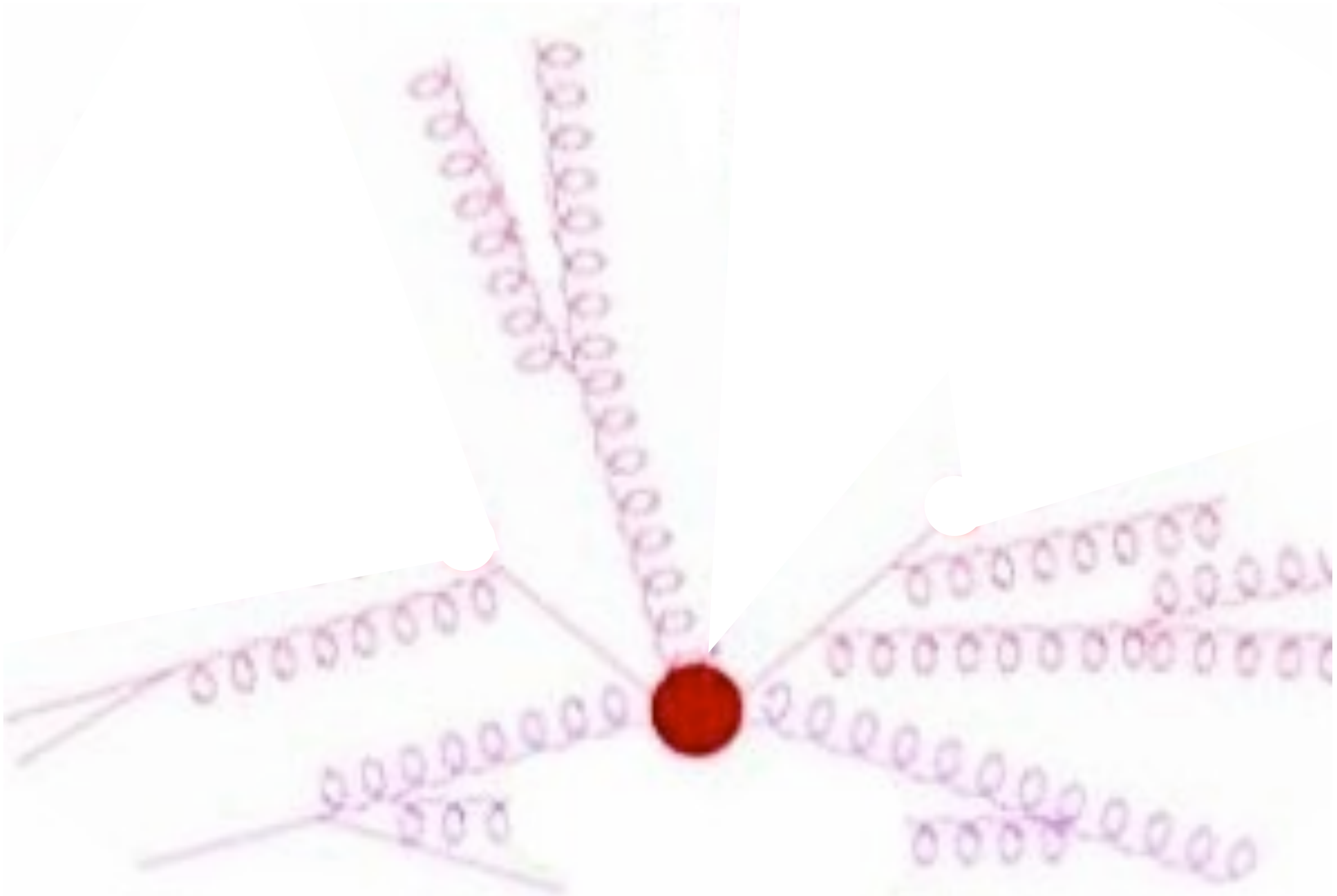
$$R_{ij} = 2[\cosh(y_i - y_j) - \cos(\phi_i - \phi_j)] \simeq (\Delta y)^2 + (\Delta\phi)^2$$

Find the smallest k_{Tij} (or $k_{Ti\mathrm{beam}}$), combine partons *i* and *j* (or *i* and the beam), and continue until you reach a 2 → 2 (or 2 → 1) scattering.

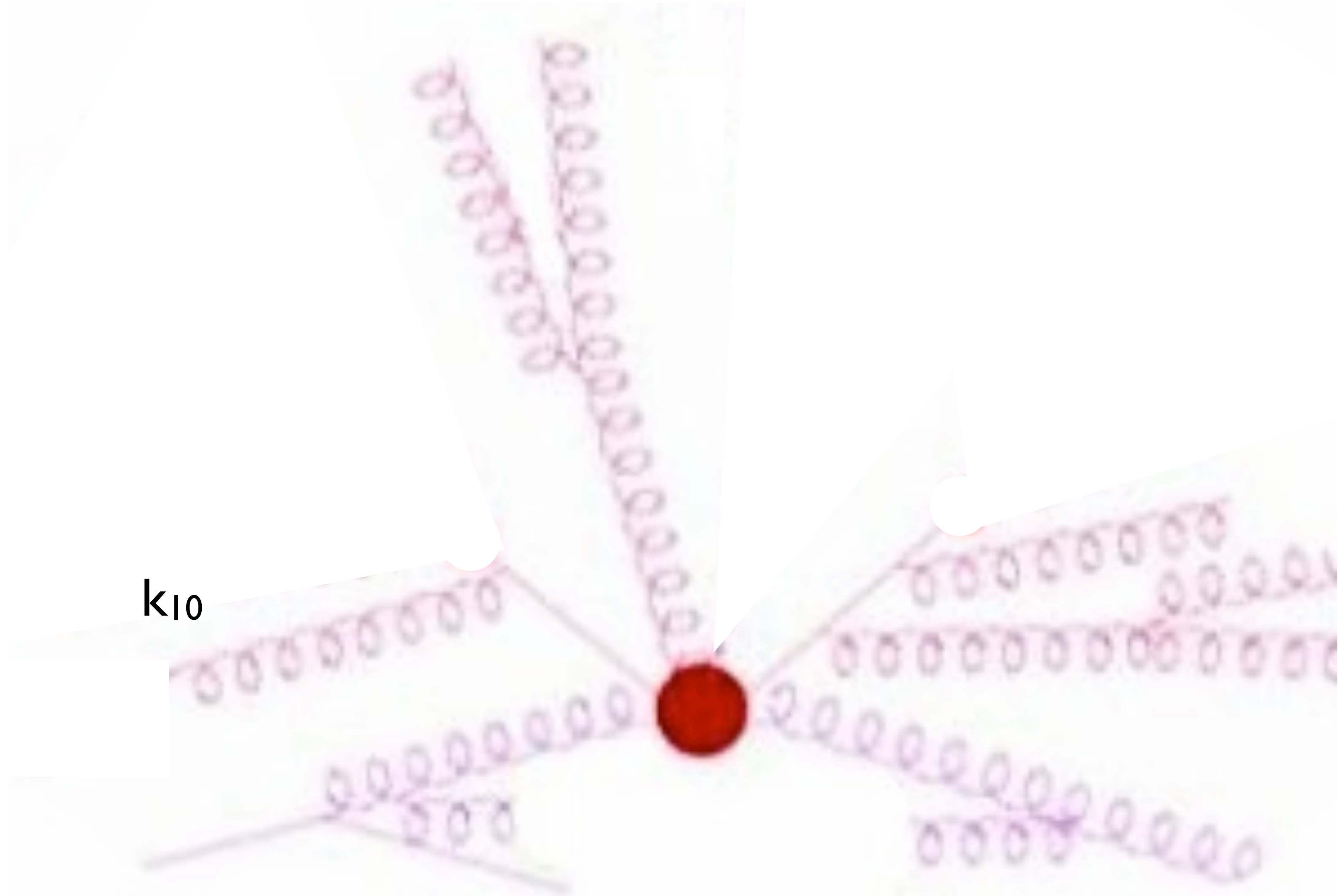
Clustering example



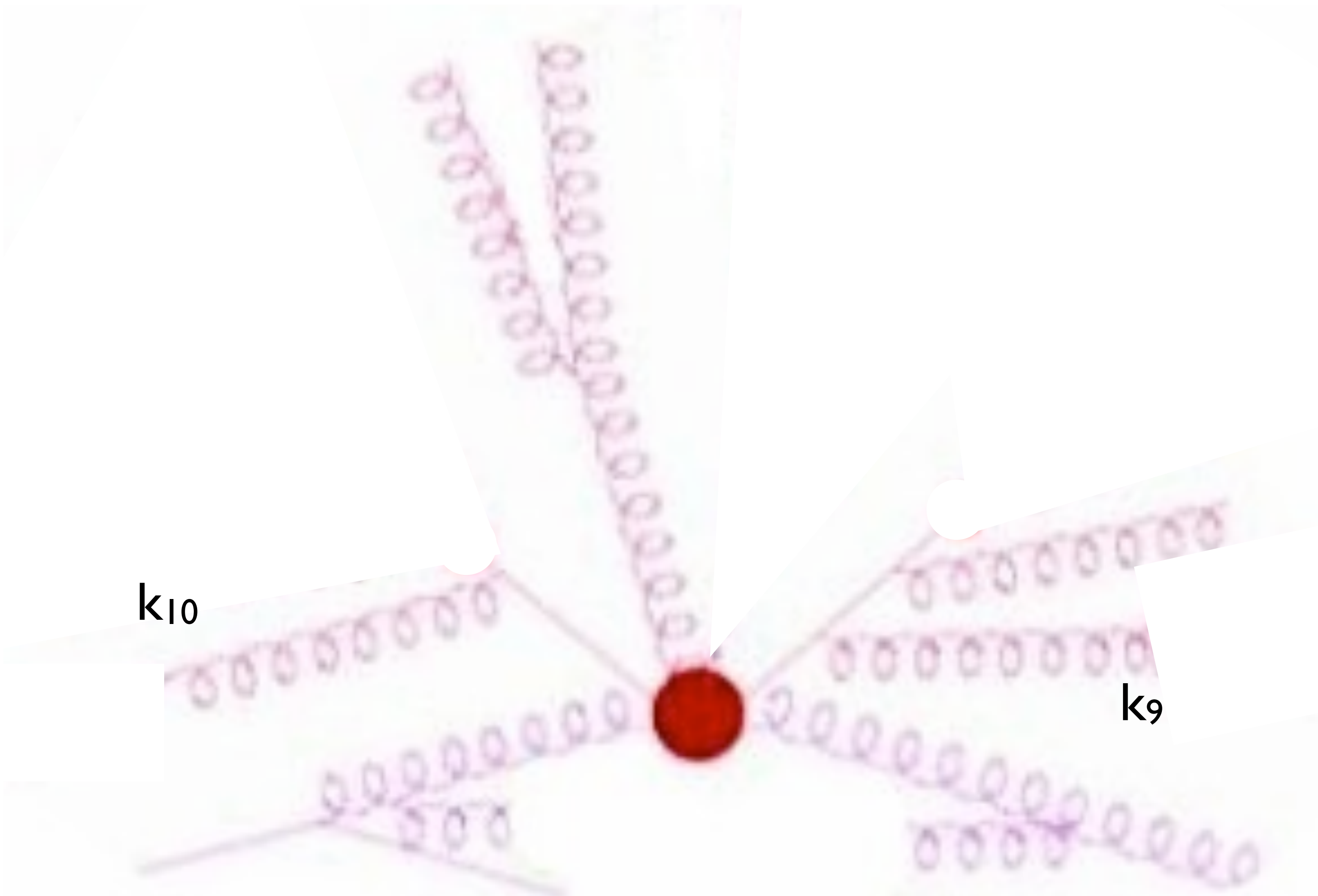
Clustering example



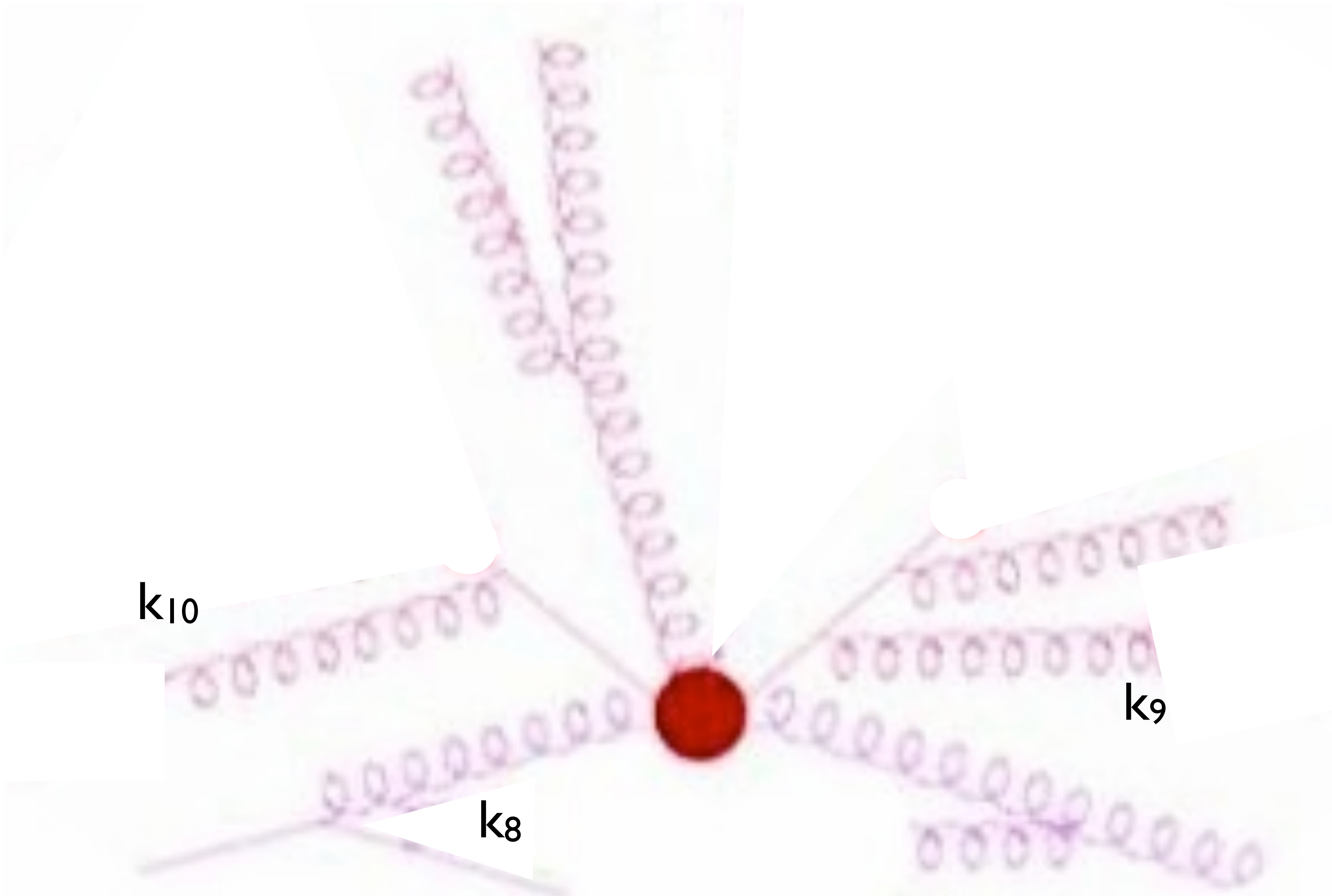
Clustering example



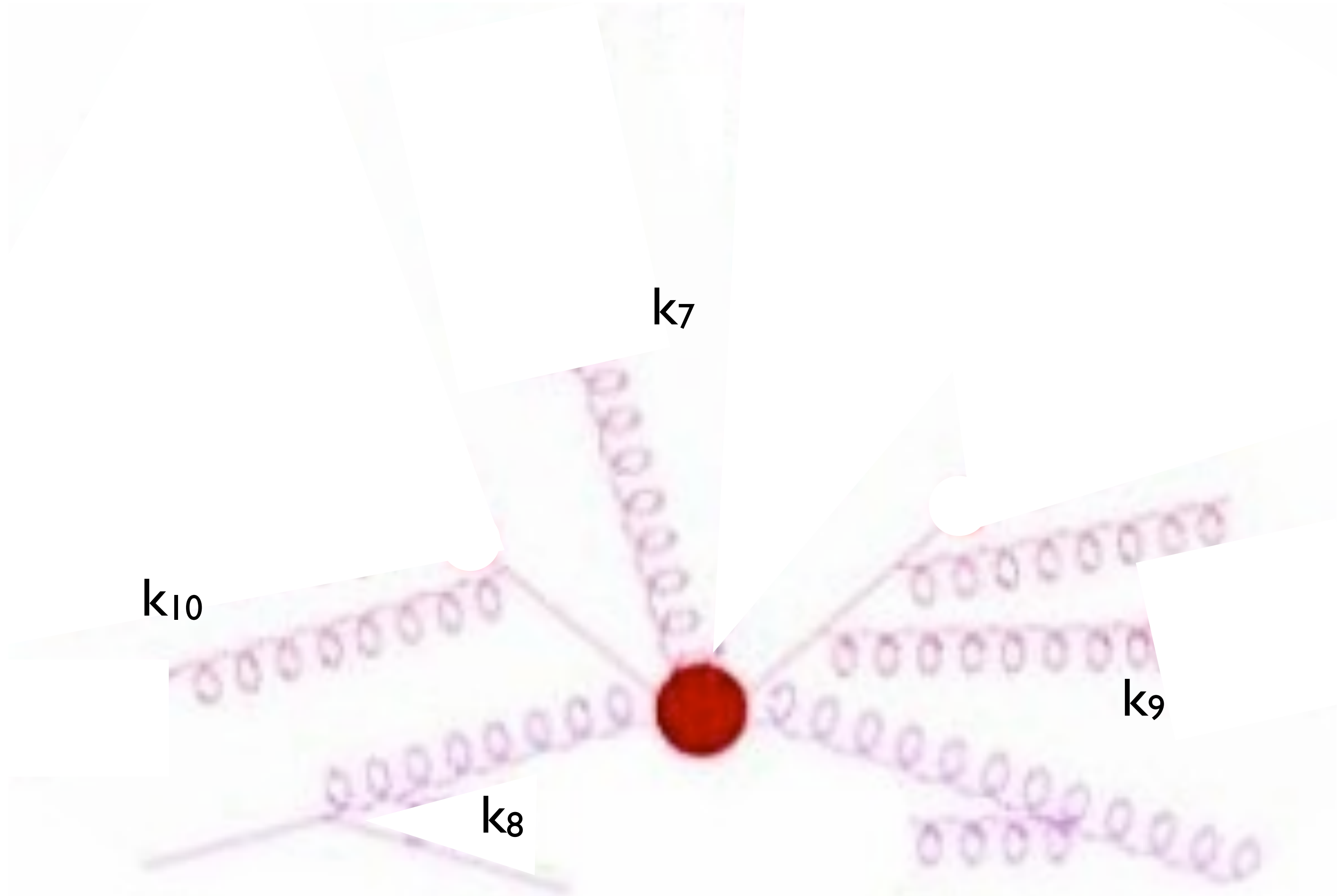
Clustering example



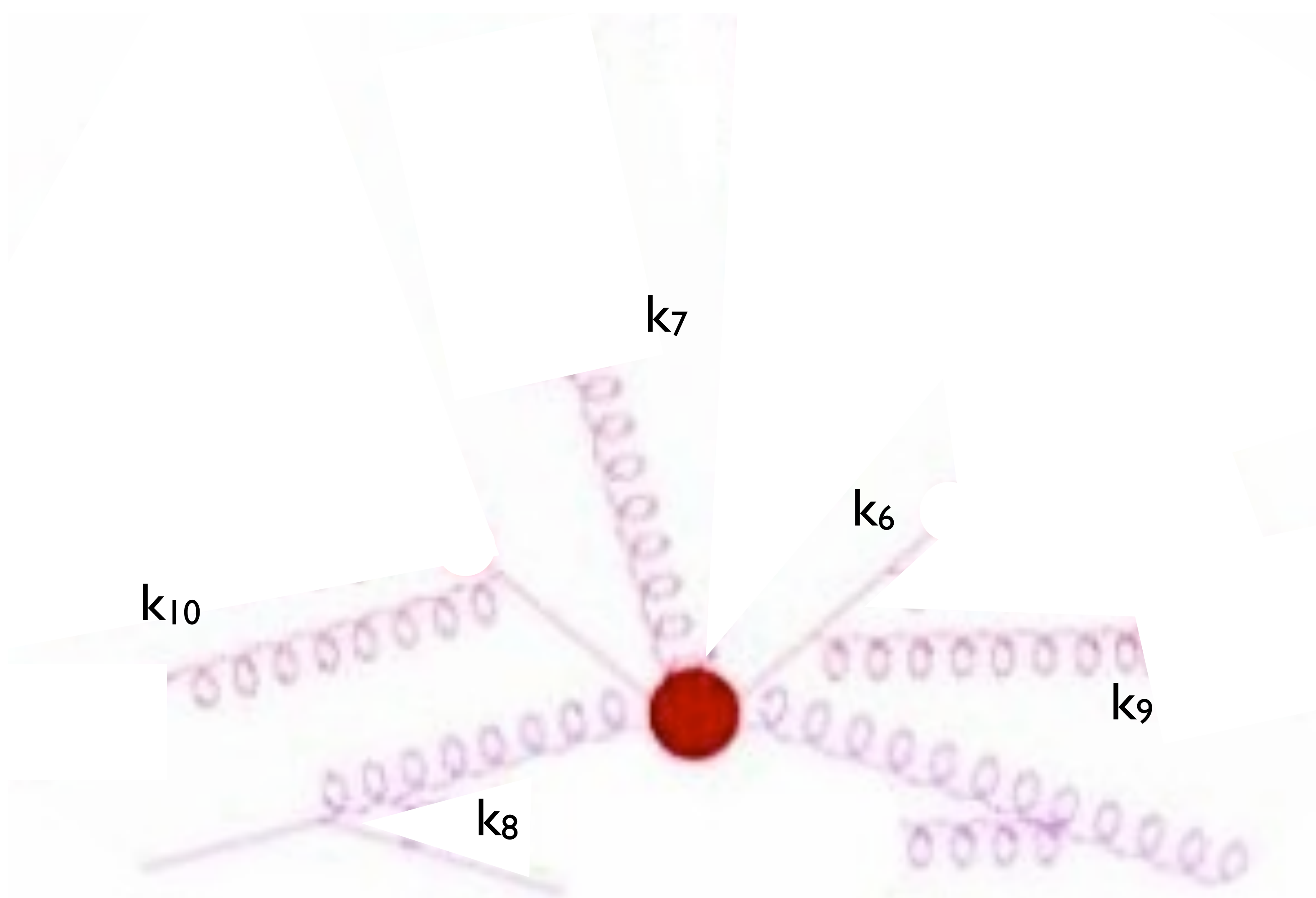
Clustering example



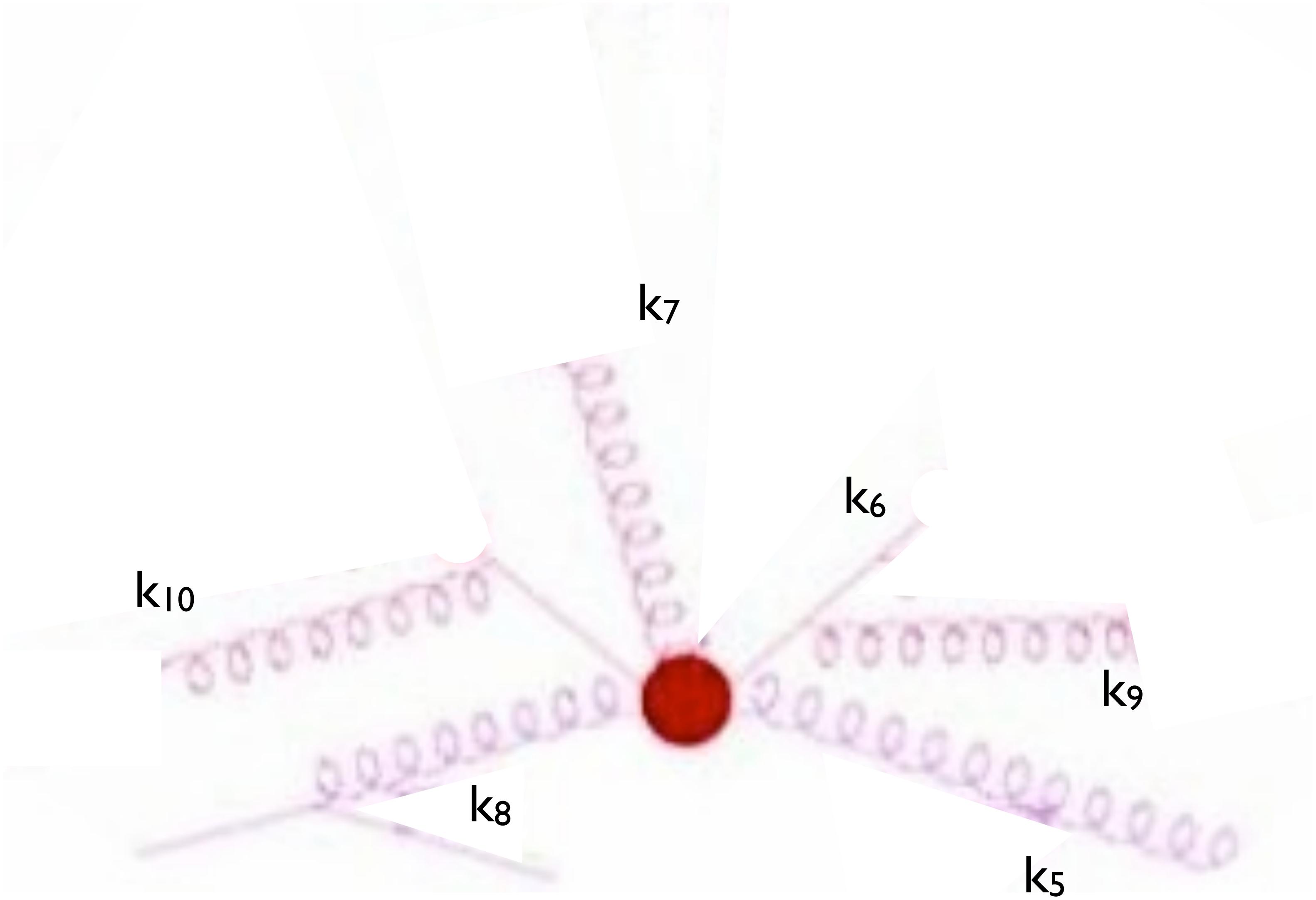
Clustering example



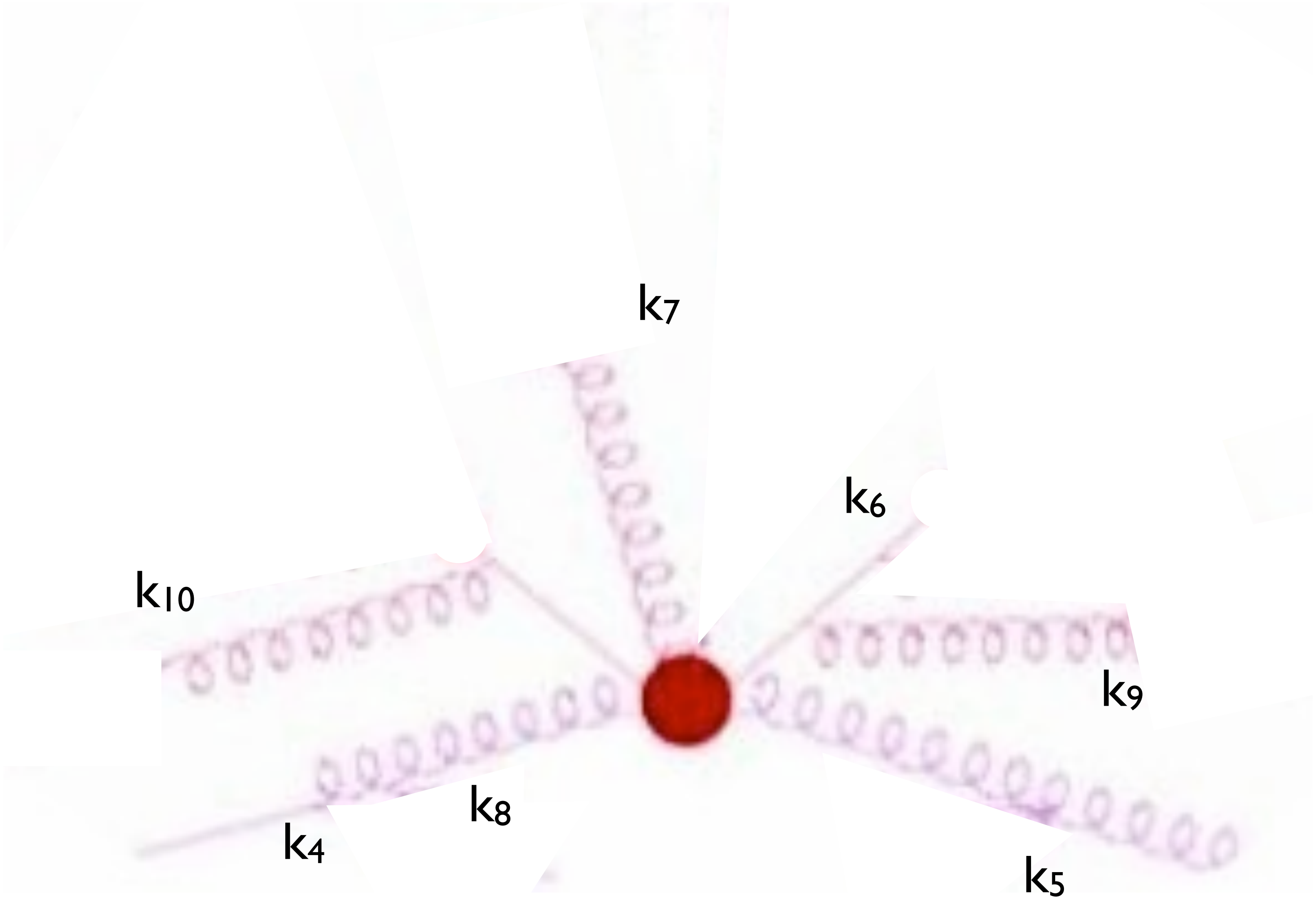
Clustering example



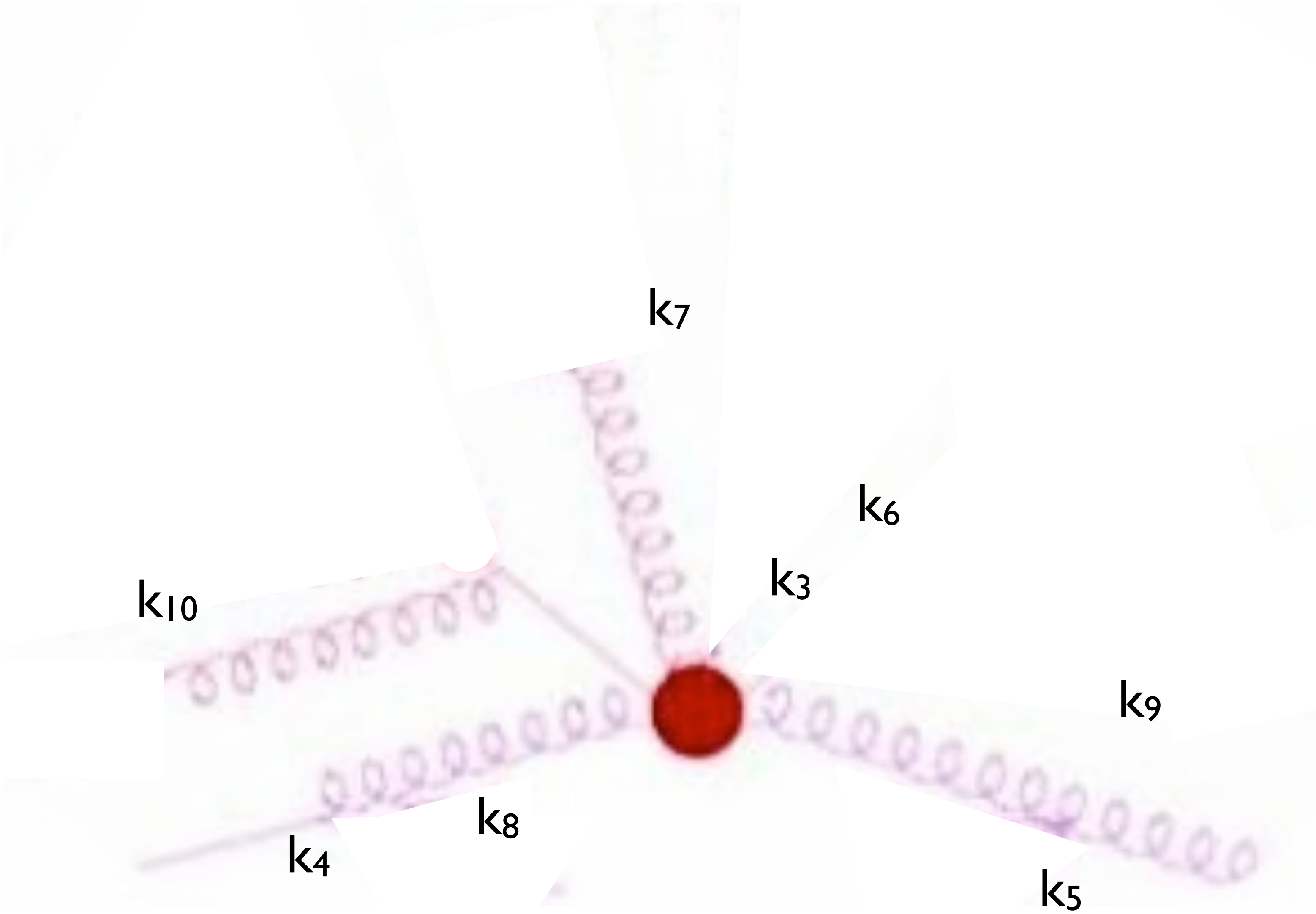
Clustering example



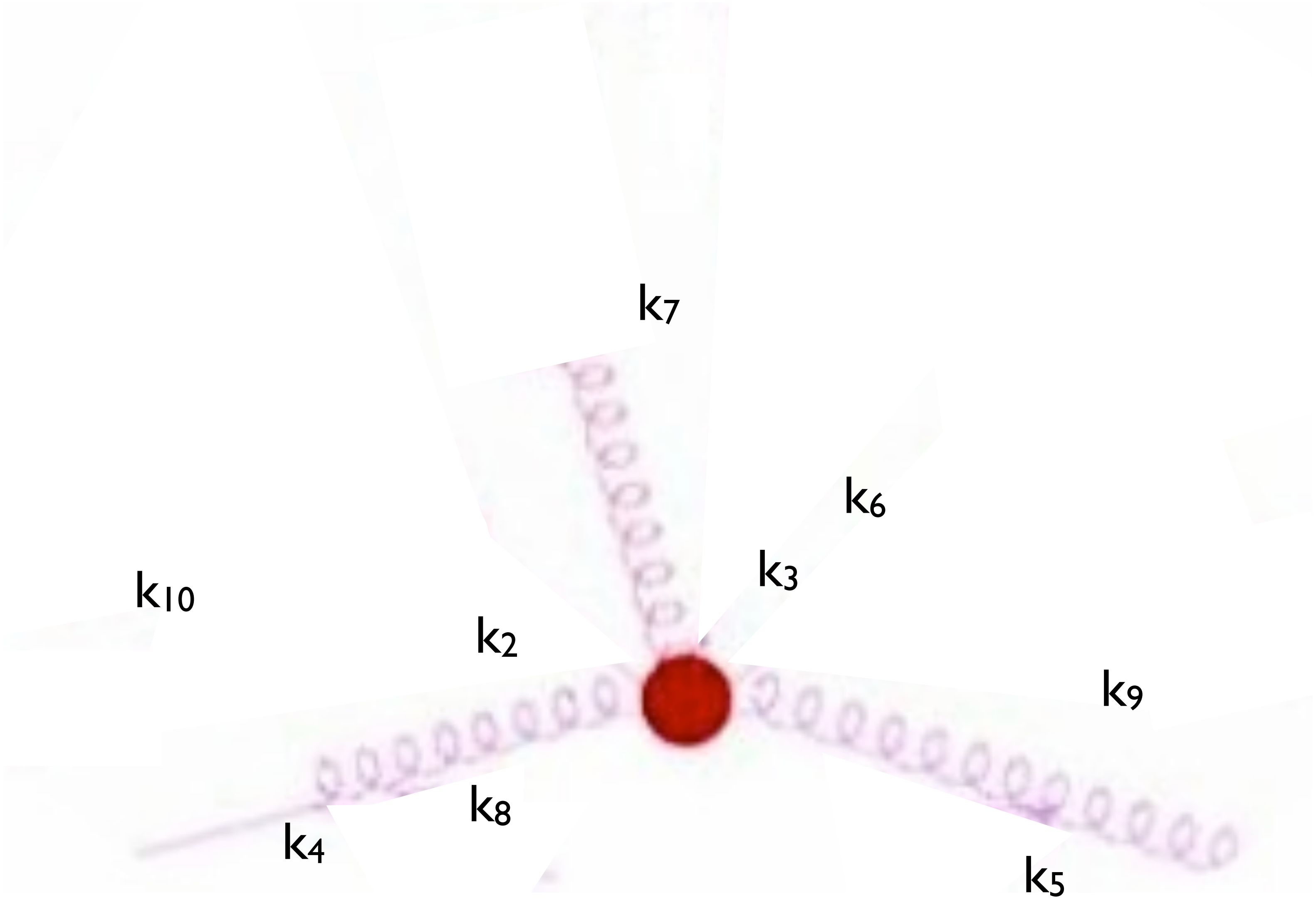
Clustering example



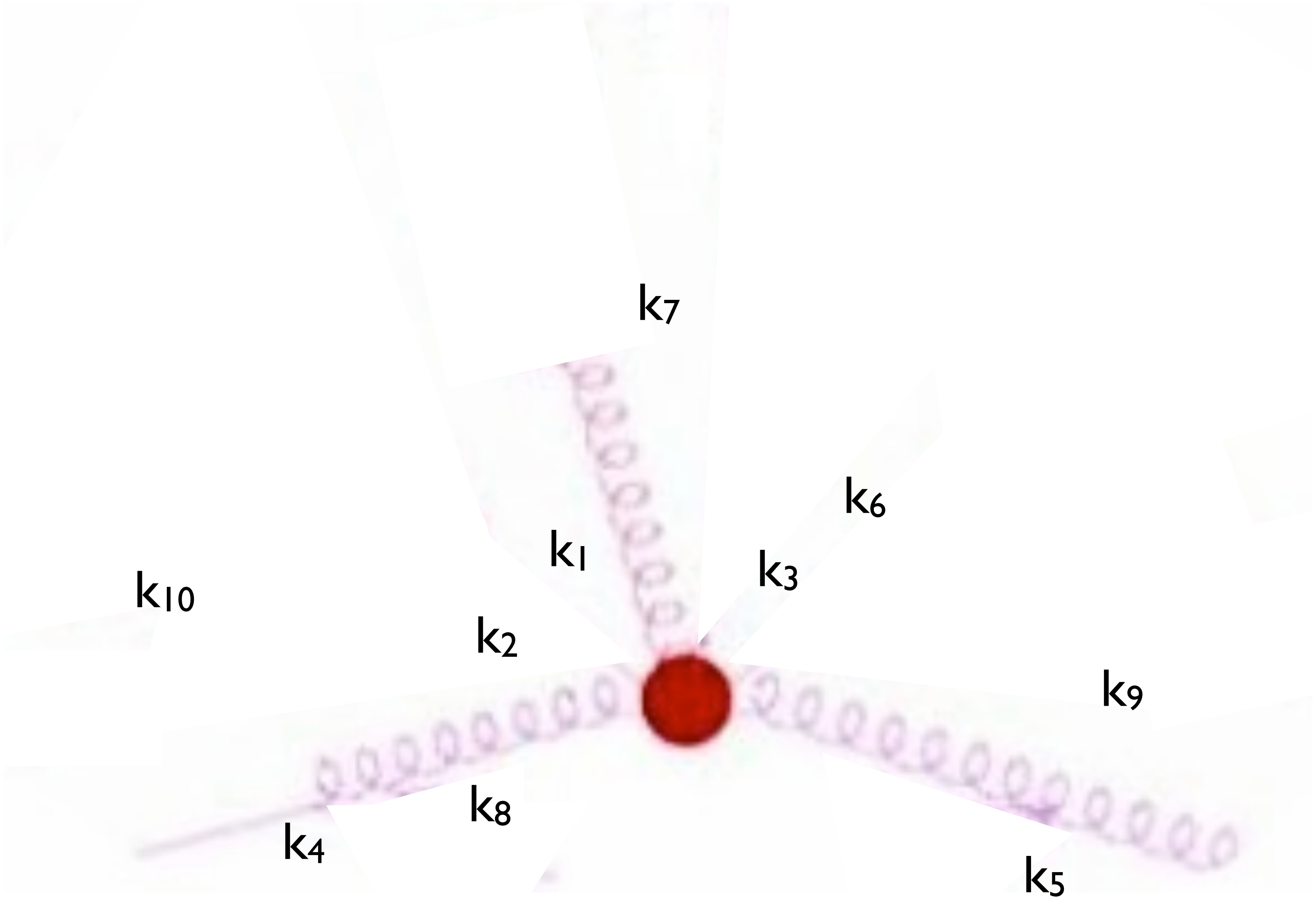
Clustering example



Clustering example



Clustering example



Matching schemes

- We still haven't specified how to apply the Sudakov reweighting to the matrix element
- Three general schemes available in the literature:
  - ➡ CKKW scheme [Catani,Krauss,Kuhn,Webber 2001; Krauss 2002]
  - ➡ Lönnblad scheme (or CKKW-L) [Lönnblad 2002]
  - ➡ MLM scheme [Mangano *unpublished* 2002; Mangano et al. 2007]

CKKW matching

[Catani, Krauss, Kuhn, Webber 2001]
[Krauss 2002]

CKKW matching

[Catani, Krauss, Kuhn, Webber 2001]
[Krauss 2002]

- Apply the required Sudakov suppression

$$(\Delta_{Iq}(t_{\rm cut},t_0))^2\Delta_g(t_2,t_1)(\Delta_q(t_{\rm cut},t_2))^2$$

analytically, using the best available (NLL) Sudakovs.

CKKW matching

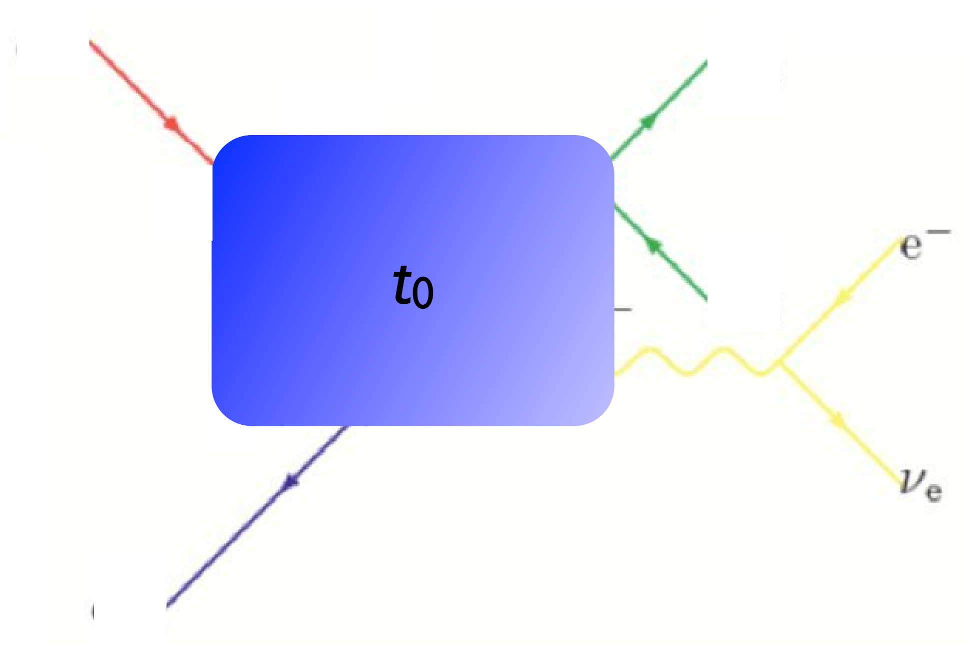
[Catani, Krauss, Kuhn, Webber 2001]
[Krauss 2002]

- Apply the required Sudakov suppression

$$(\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2$$

analytically, using the best available (NLL) Sudakovs.

- Perform "truncated showering": Run the parton shower starting at t_0, but forbid any showers above the cutoff scale $t_{\rm cut}$.



CKKW matching

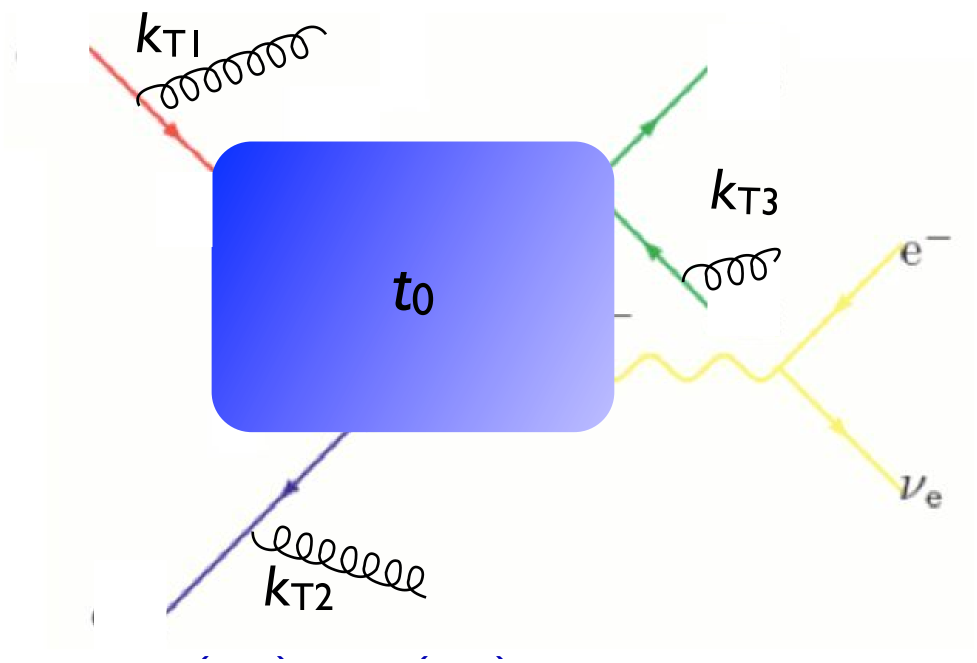
[Catani, Krauss, Kuhn, Webber 2001]
[Krauss 2002]

- Apply the required Sudakov suppression

$$(\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2$$

analytically, using the best available (NLL) Sudakovs.

- Perform "truncated showering": Run the parton shower starting at t_0, but forbid any showers above the cutoff scale $t_{\rm cut}$.



CKKW matching

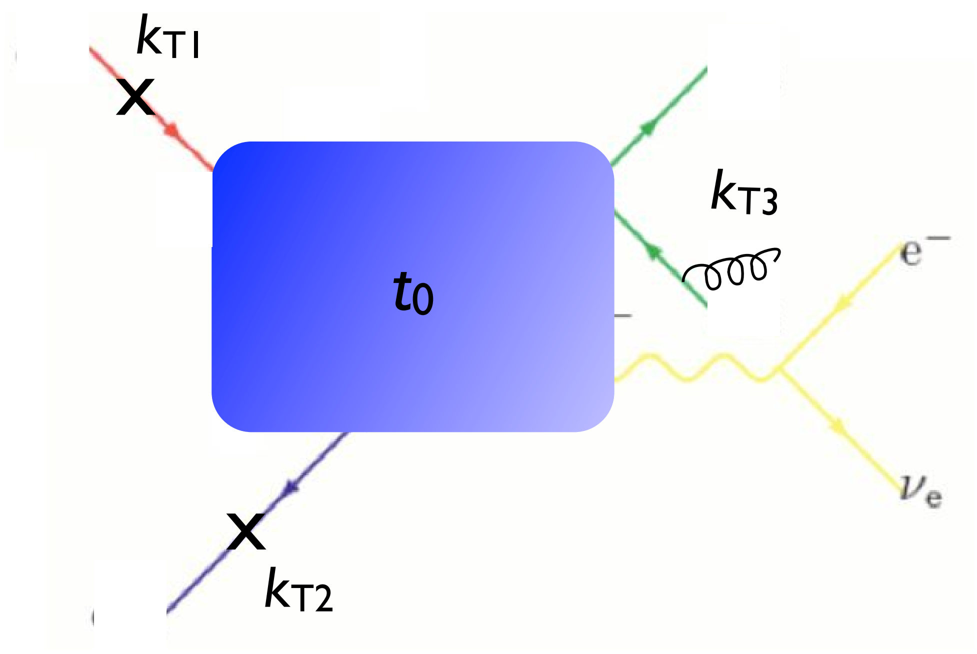
[Catani, Krauss, Kuhn, Webber 2001]
[Krauss 2002]

- Apply the required Sudakov suppression

$$(\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2$$

analytically, using the best available (NLL) Sudakovs.

- Perform "truncated showering": Run the parton shower starting at t_0, but forbid any showers above the cutoff scale $t_{\rm cut}$.



CKKW matching

[Catani, Krauss, Kuhn, Webber 2001]
[Krauss 2002]

- Apply the required Sudakov suppression

$$(\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2$$

analytically, using the best available (NLL) Sudakovs.

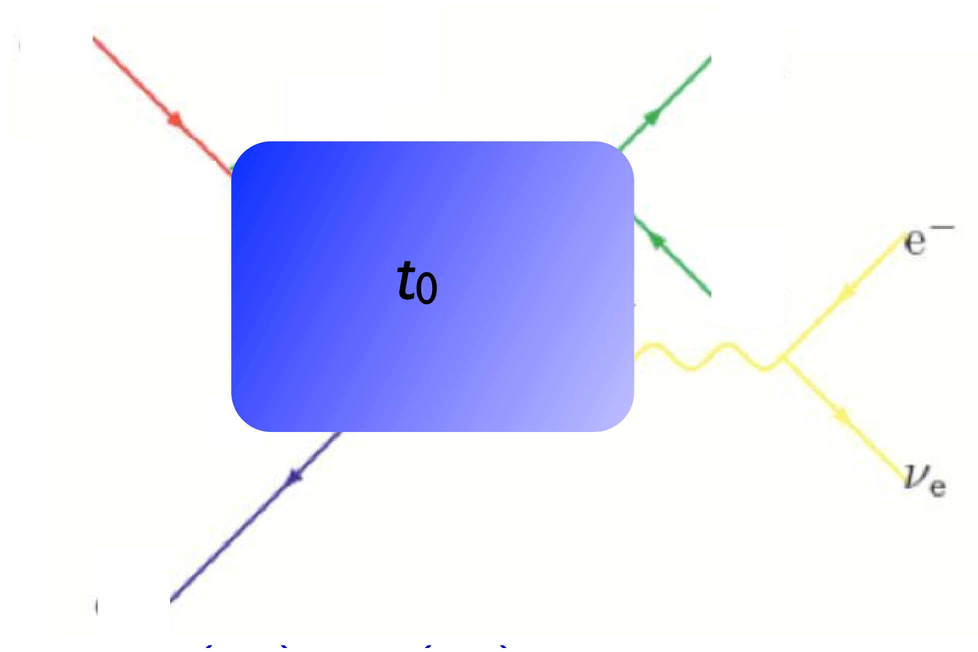
- Perform "truncated showering": Run the parton shower starting at t_0, but forbid any showers above the cutoff scale $t_{\rm cut}$.

✓ Best theoretical treatment of matrix element

- Requires dedicated PS implementation
- Mismatch between analytical Sudakov and (non-NLL) shower
- Implemented in Sherpa (v. 1.1)

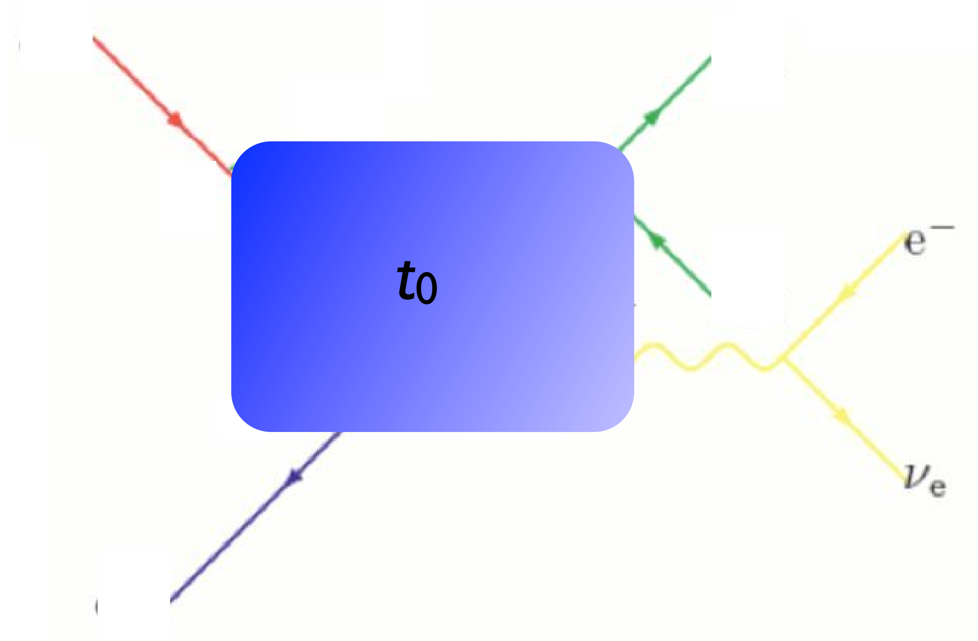
CKKW-L matching

[Lönnblad 2002]
[Hoeche et al. 2009]



CKKW-L matching

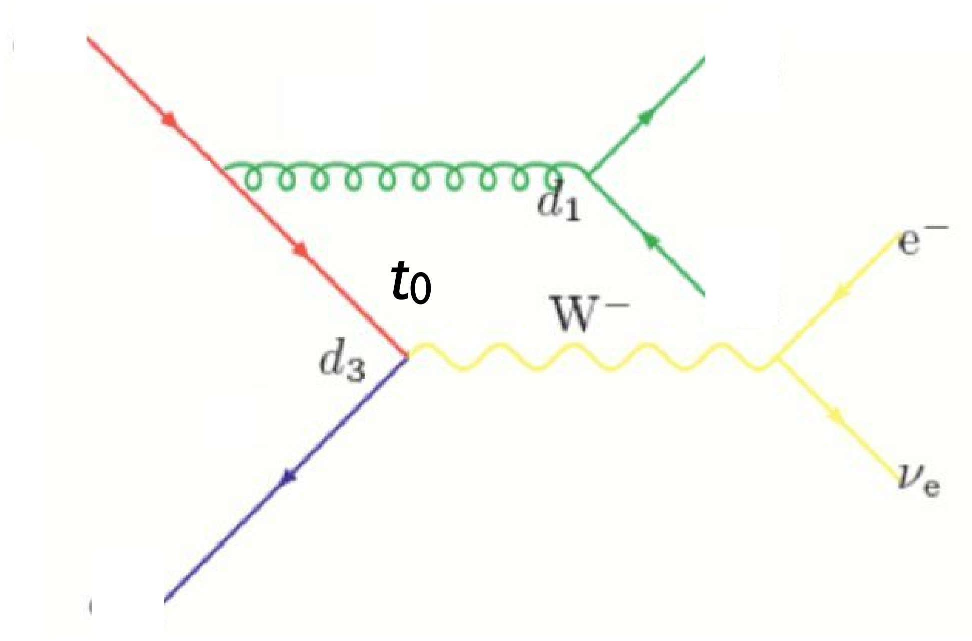
[Lönnblad 2002]
[Hoeche et al. 2009]



- Cluster back to "parton shower history"

CKKW-L matching

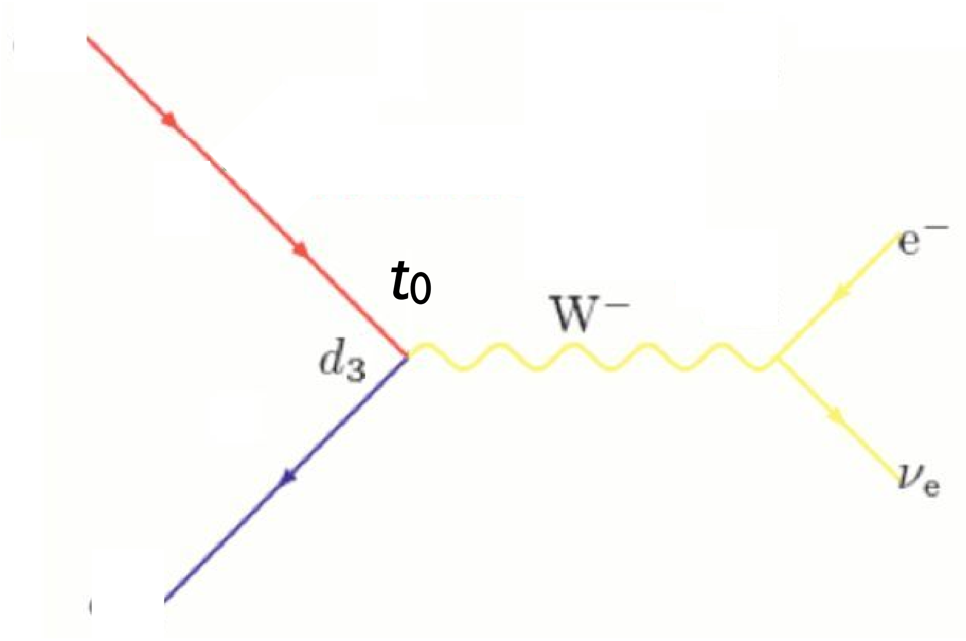
[Lönnblad 2002]
[Hoeche et al. 2009]



- Cluster back to "parton shower history"

CKKW-L matching

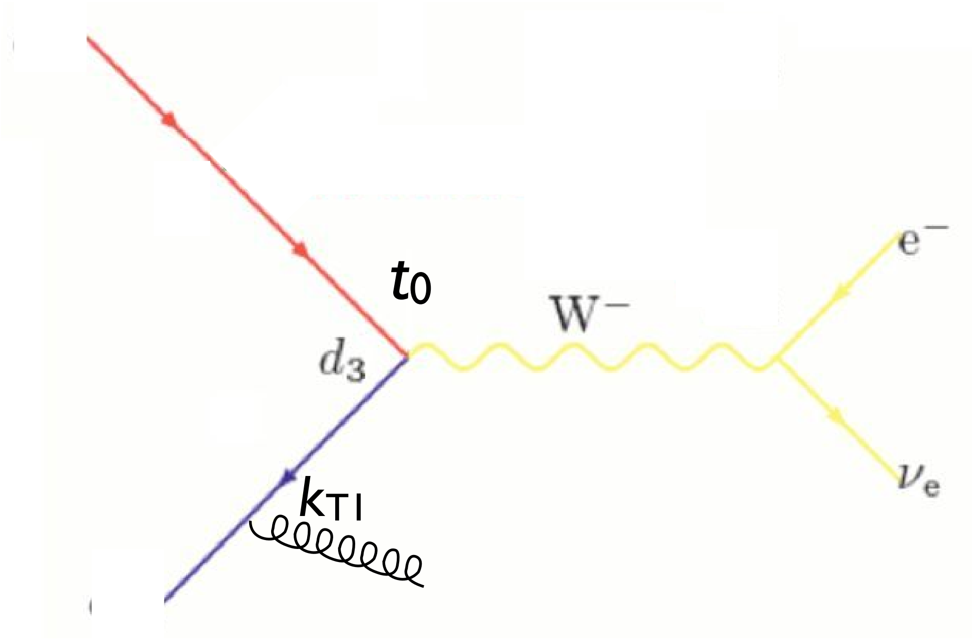
[Lönnblad 2002]
[Hoeche et al. 2009]



- Cluster back to "parton shower history"
- Perform showering step-by-step for each step in the parton shower history, starting from the clustering scale for that step

CKKW-L matching

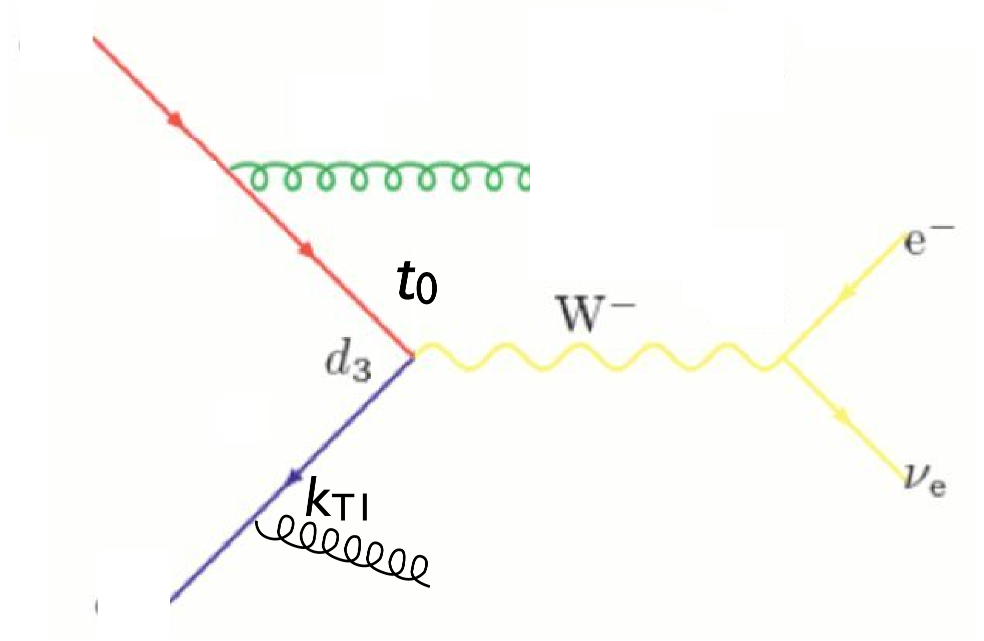
[Lönnblad 2002]
[Hoeche et al. 2009]



- Cluster back to "parton shower history"
- Perform showering step-by-step for each step in the parton shower history, starting from the clustering scale for that step

CKKW-L matching

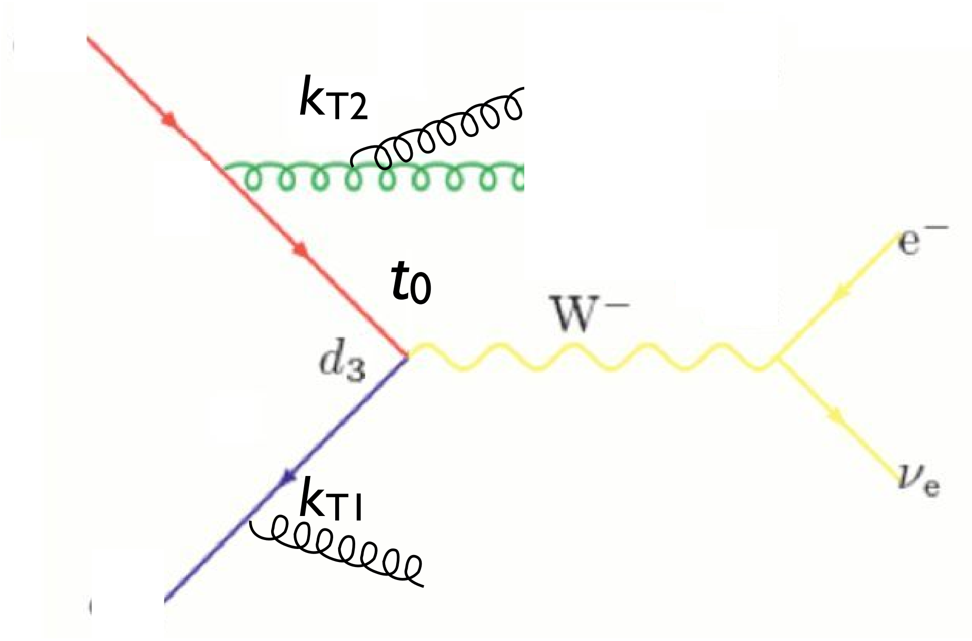
[Lönnblad 2002]
[Hoeche et al. 2009]



- Cluster back to "parton shower history"
- Perform showering step-by-step for each step in the parton shower history, starting from the clustering scale for that step

CKKW-L matching

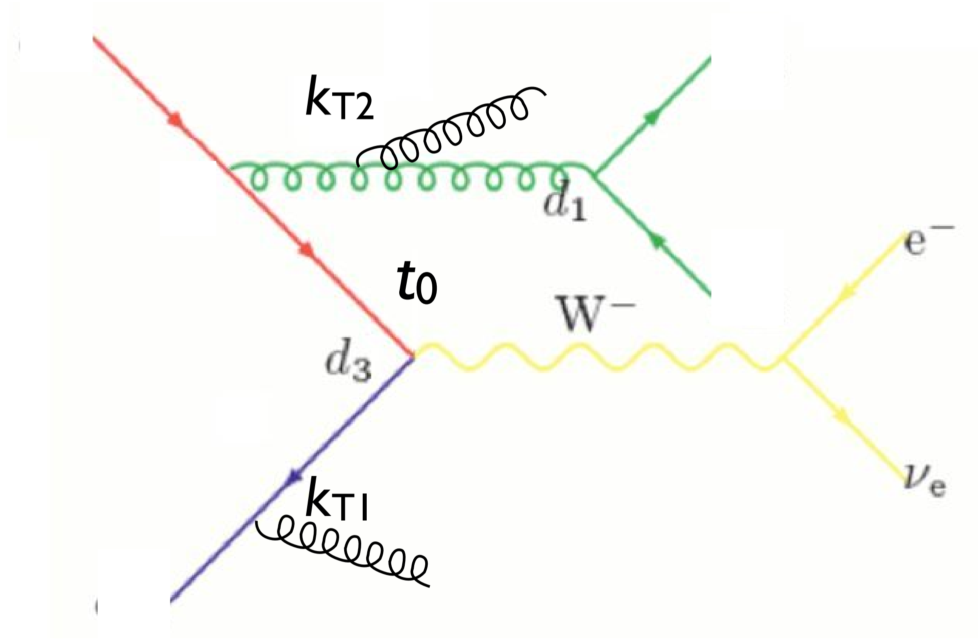
[Lönnblad 2002]
[Hoeche et al. 2009]



- Cluster back to "parton shower history"
- Perform showering step-by-step for each step in the parton shower history, starting from the clustering scale for that step

CKKW-L matching

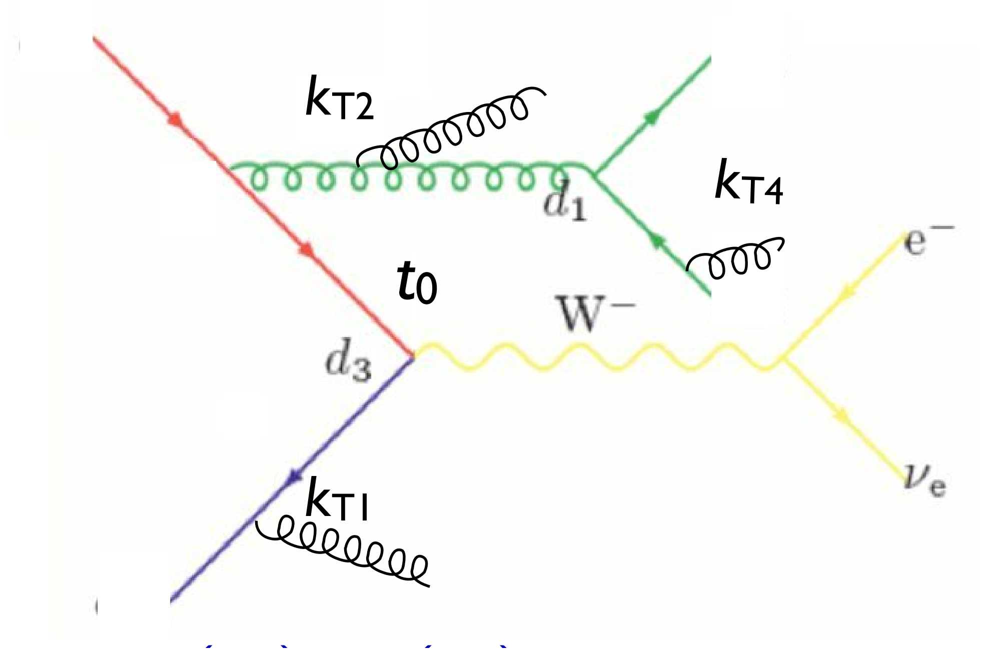
[Lönnblad 2002]
[Hoeche et al. 2009]



- Cluster back to "parton shower history"
- Perform showering step-by-step for each step in the parton shower history, starting from the clustering scale for that step

CKKW-L matching

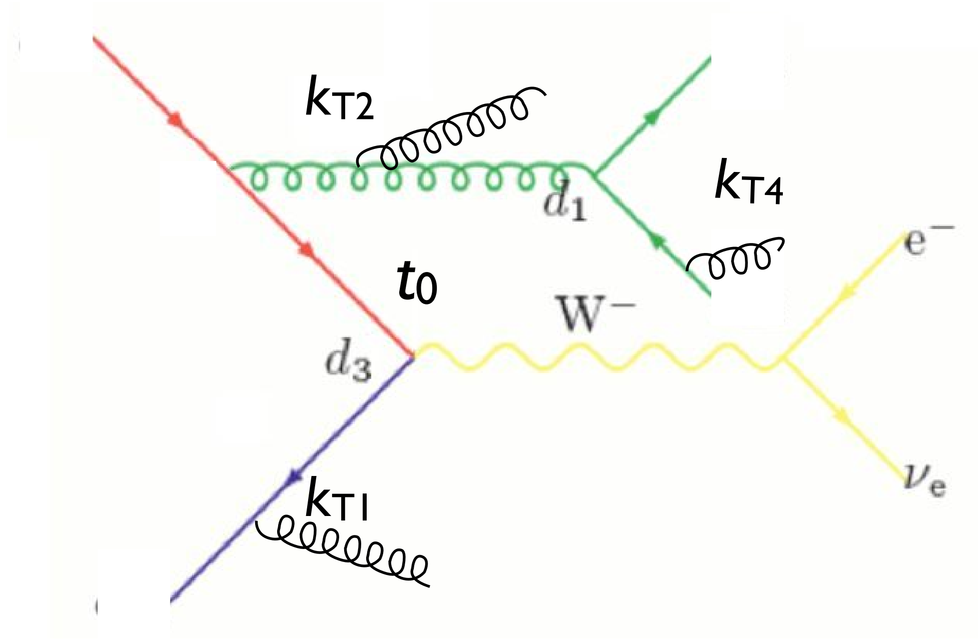
[Lönnblad 2002]
[Hoeche et al. 2009]



- Cluster back to "parton shower history"
- Perform showering step-by-step for each step in the parton shower history, starting from the clustering scale for that step

CKKW-L matching

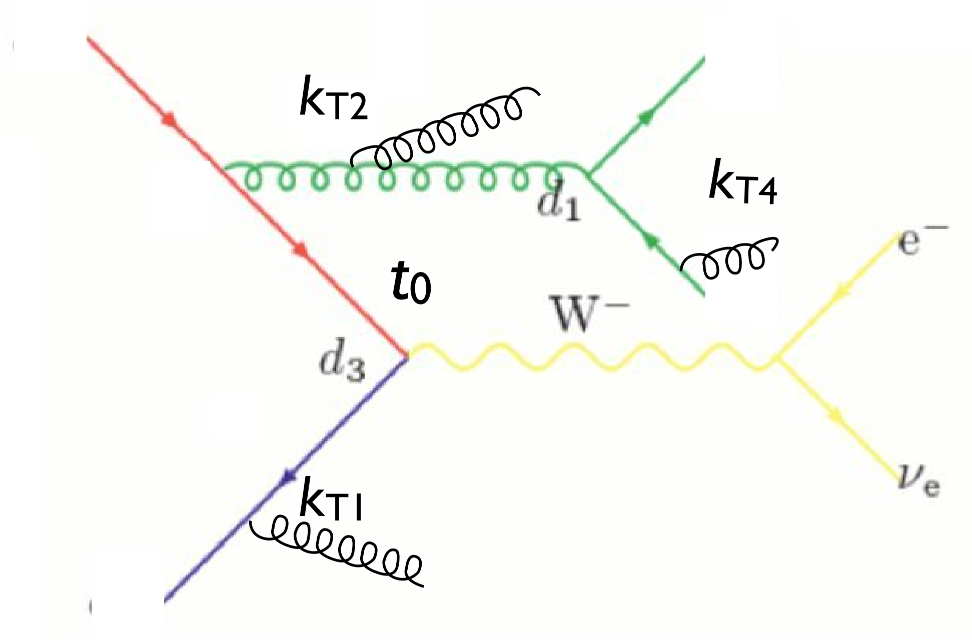
[Lönnblad 2002]
[Hoeche et al. 2009]



- Cluster back to "parton shower history"
- Perform showering step-by-step for each step in the parton shower history, starting from the clustering scale for that step
- Veto the event if any shower is harder than the clustering scale for the next step (or t_{cut}, if last step)

CKKW-L matching

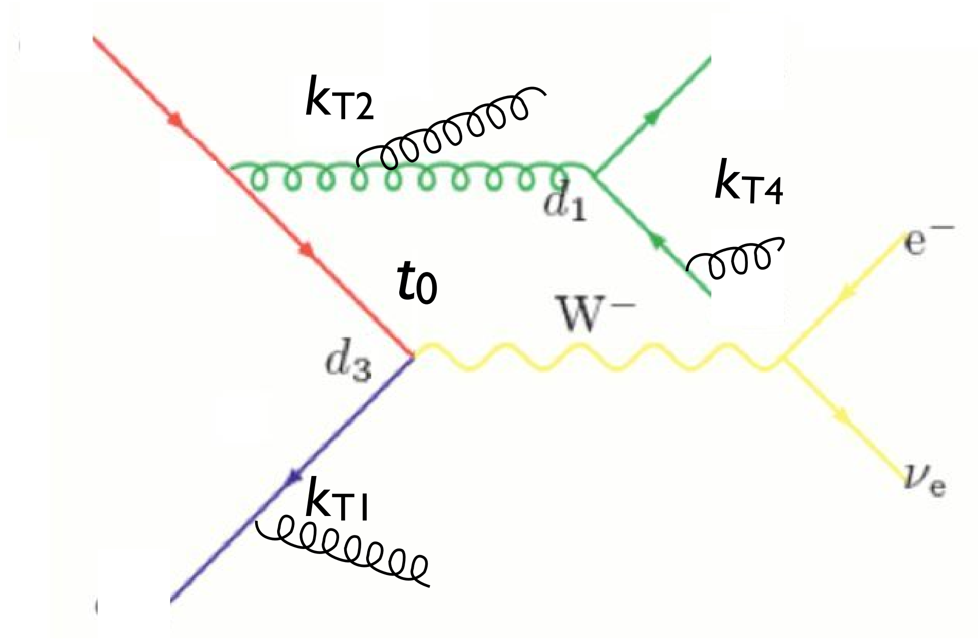
[Lönnblad 2002]
[Hoeche et al. 2009]



- Cluster back to "parton shower history"
- Perform showering step-by-step for each step in the parton shower history, starting from the clustering scale for that step
- Veto the event if any shower is harder than the clustering scale for the next step (or t_{cut}, if last step)
- Keep any shower emissions that are softer than the clustering scale for the next step

CKKW-L matching

[Lönnblad 2002]
[Hoeche et al. 2009]



- Cluster back to "parton shower history"
- Perform showering step-by-step for each step in the parton shower history, starting from the clustering scale for that step

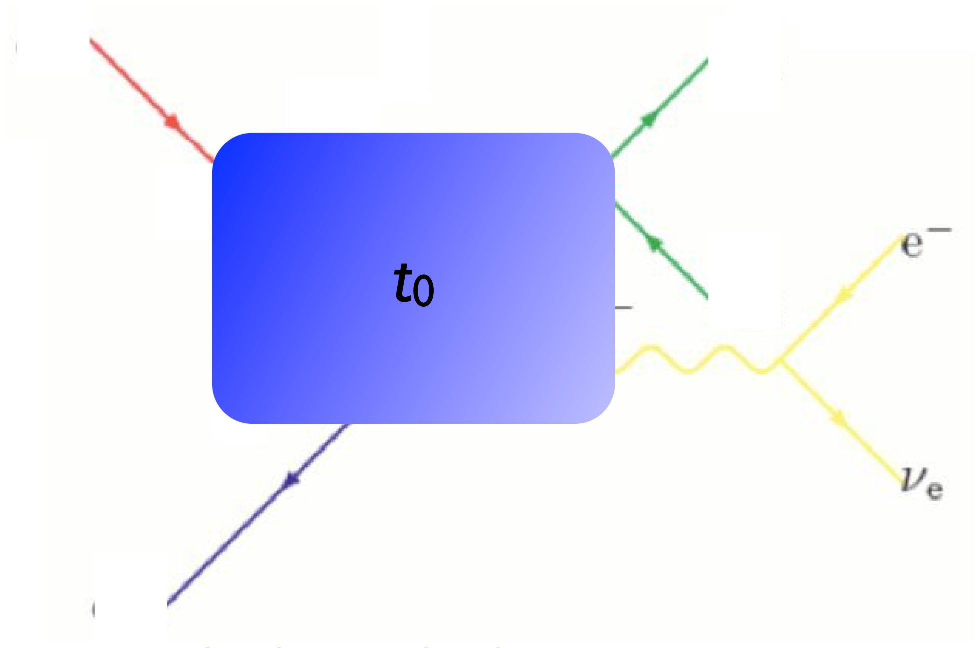
✓ Automatic agreement between Sudakov and shower

- Requires dedicated PS implementation
  ➡ Need multiple implementations to compare between showers

- Implemented in Ariadne, Sherpa (v. 1.2), and Pythia 8

MLM matching

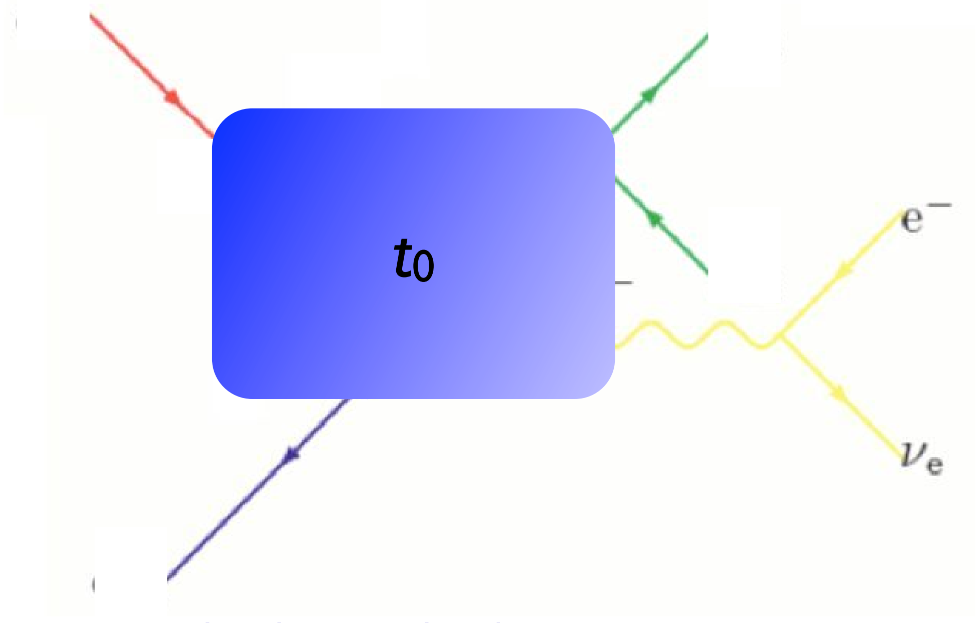
[M.L. Mangano, ~2002, 2007]
[J.A. et al 2007, 2008]



MLM matching

[M.L. Mangano, ~2002, 2007]
[J.A. et al 2007, 2008]

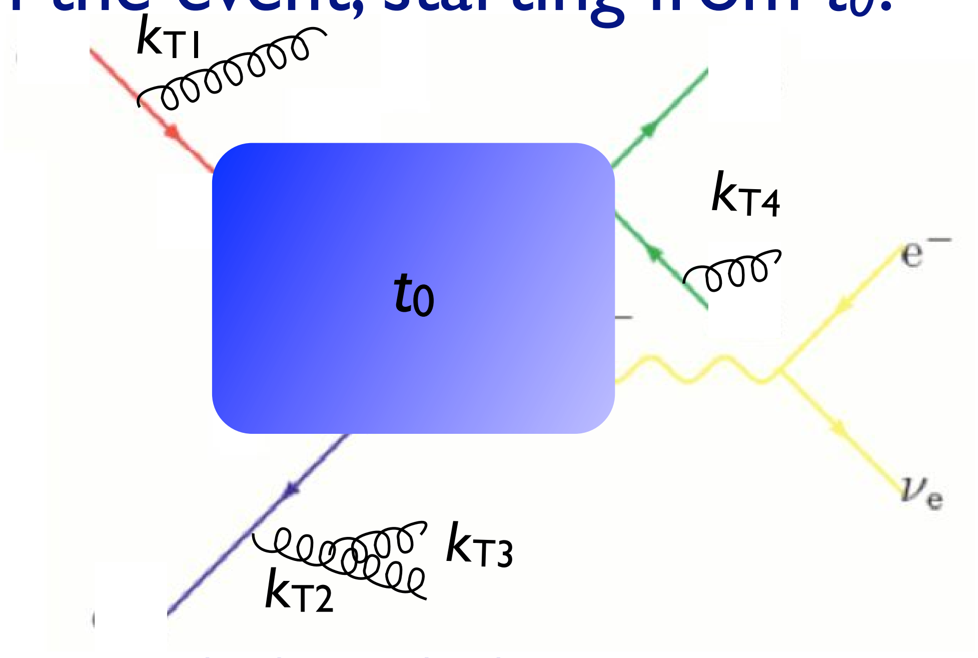
- The simplest way to do the Sudakov suppression is to run the shower on the event, starting from t_0!



MLM matching

[M.L. Mangano, ~2002, 2007]
[J.A. et al 2007, 2008]

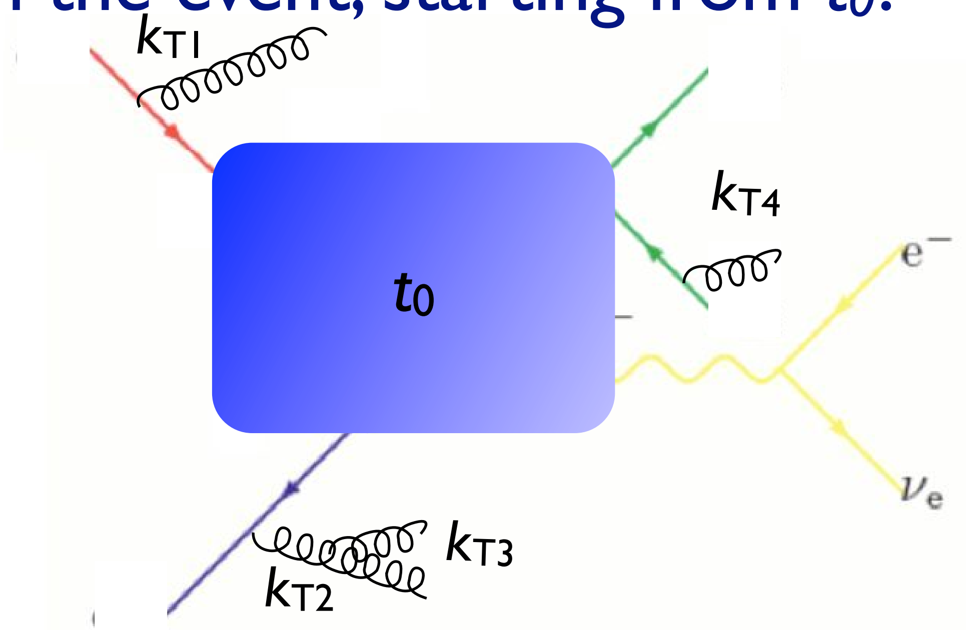
- The simplest way to do the Sudakov suppression is to run the shower on the event, starting from t_0!



MLM matching

[M.L. Mangano, ~2002, 2007]
[J.A. et al 2007, 2008]

- The simplest way to do the Sudakov suppression is to run the shower on the event, starting from t_0!

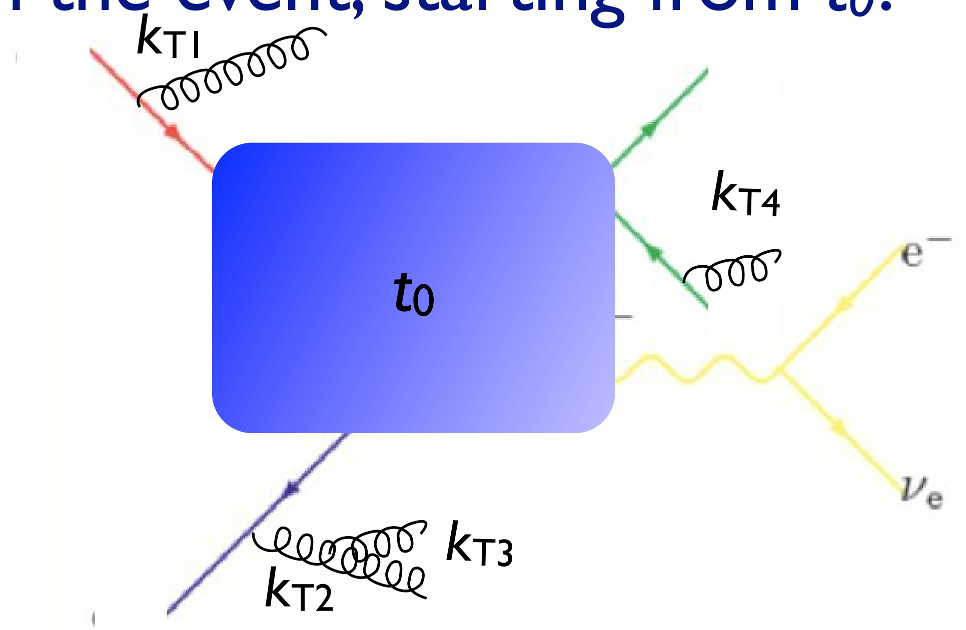


- Perform jet clustering after PS - if hardest jet $k_{T1} > t_{cut}$ or there are jets not matched to partons, reject the event

MLM matching

[M.L. Mangano, ~2002, 2007]
[J.A. et al 2007, 2008]

- The simplest way to do the Sudakov suppression is to run the shower on the event, starting from t_0!



- Perform jet clustering after PS - if hardest jet $k_{T1} > t_{cut}$ or there are jets not matched to partons, reject the event
- The resulting Sudakov suppression from the procedure is

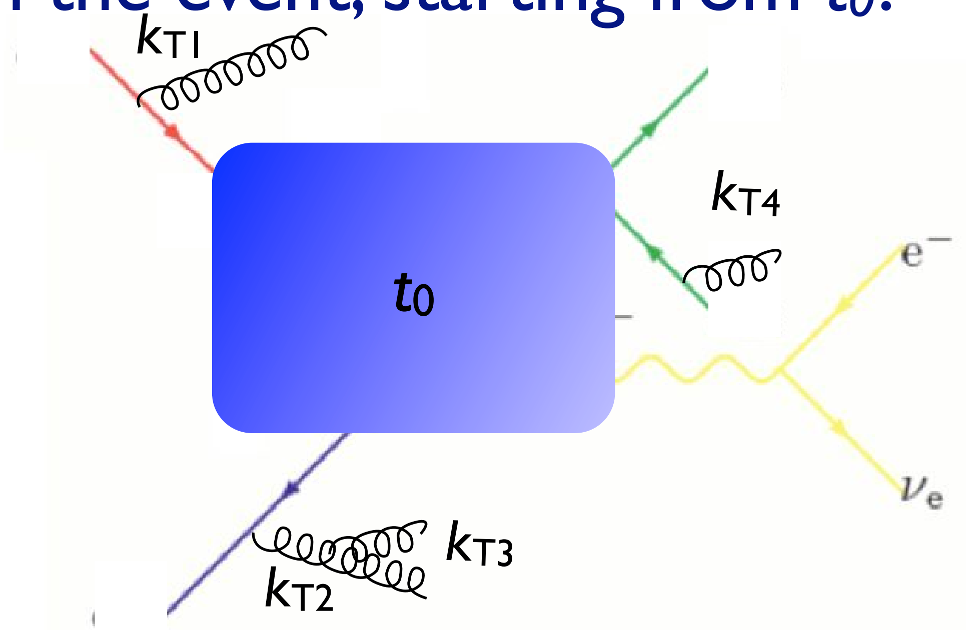
$$(\Delta_{Iq}(t_{\text{cut}}, t_0))^2 (\Delta_q(t_{\text{cut}}, t_0))^2$$

which turns out to be a good enough approximation of the correct expression $(\Delta_{Iq}(t_{\text{cut}}, t_0))^2 \Delta_g(t_2, t_1) (\Delta_q(t_{\text{cut}}, t_2))^2$

MLM matching

[M.L. Mangano, ~2002, 2007]
[J.A. et al 2007, 2008]

- The simplest way to do the Sudakov suppression is to run the shower on the event, starting from t_0!



- Perform jet clustering after PS - if hardest jet $k_{T1} > t_{cut}$ or

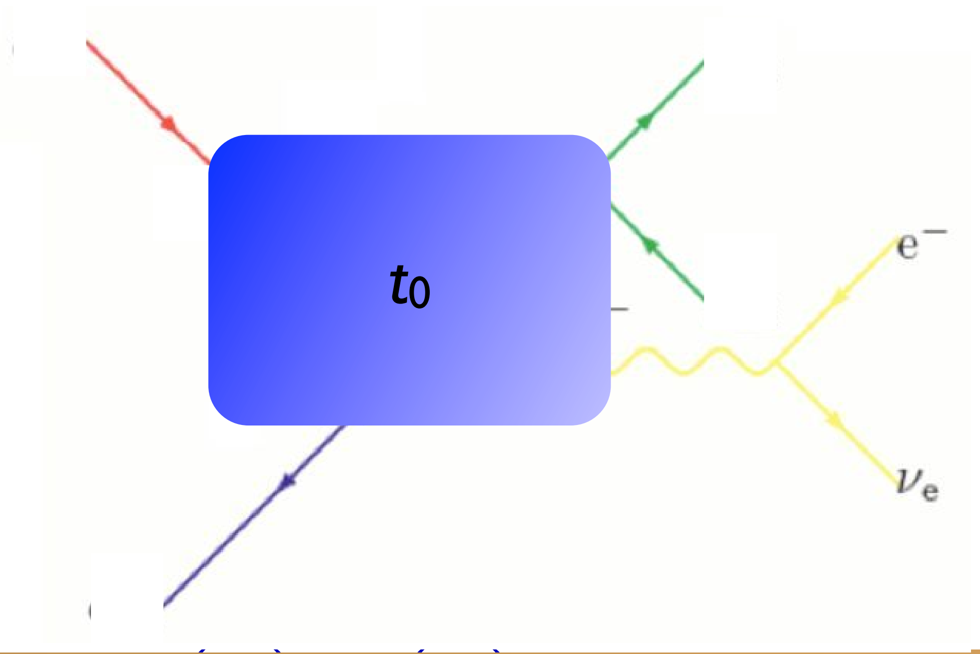
✓ Simplest available scheme

✓ Allows matching with any shower, without modification

➡ Sudakov suppression not exact, minor mismatch with shower

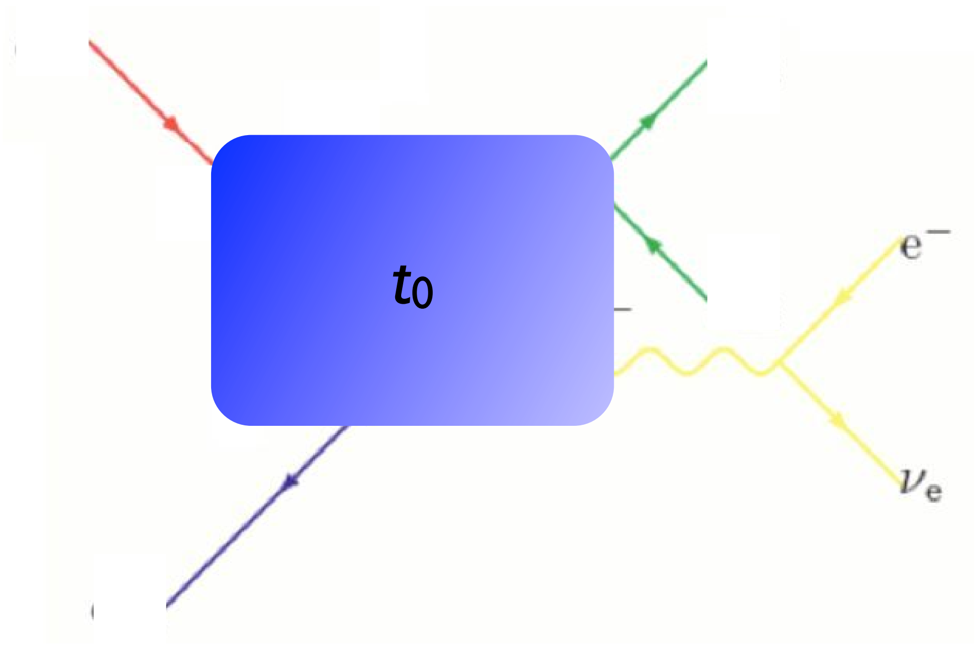
- Implemented in AlpGen, HELAC, MadGraph

MLM matching for initial state radiation



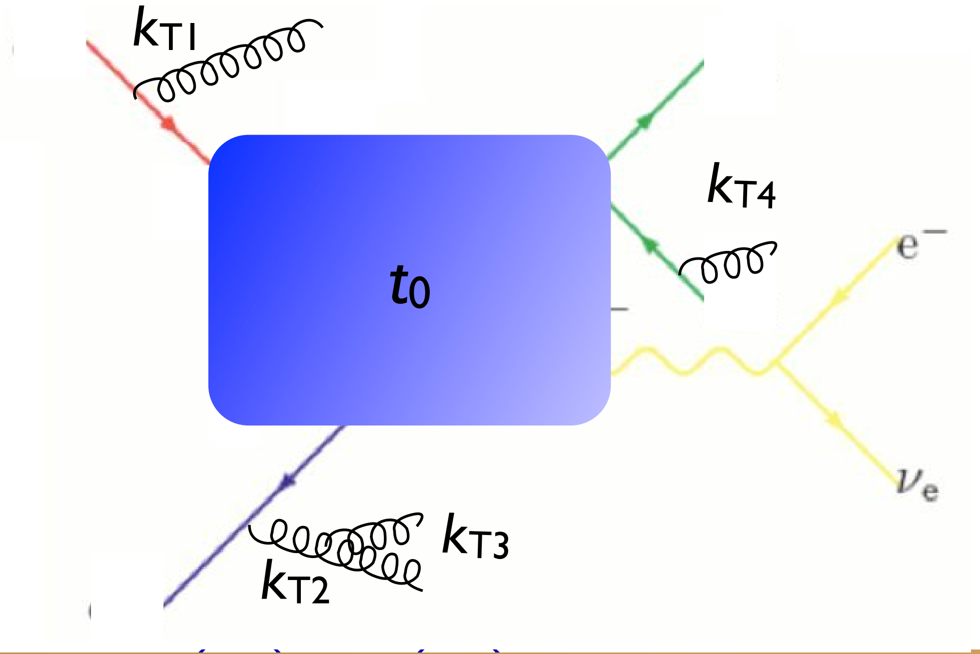
MLM matching for initial state radiation

- For MLM matching, we run the shower and then veto events if the hardest shower emission scale $k_{T1} > t_{cut}$



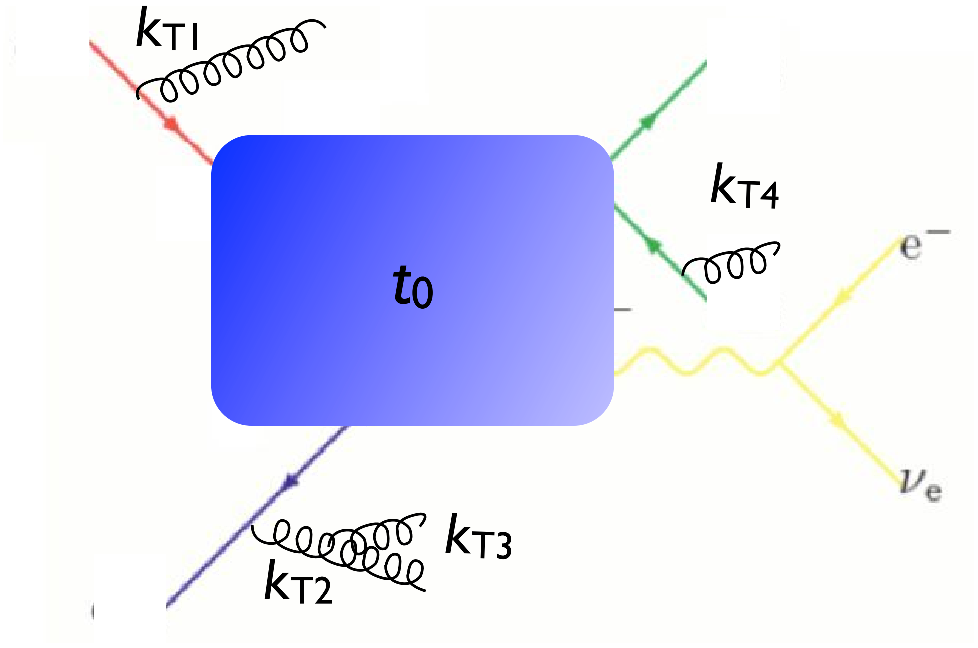
MLM matching for initial state radiation

- For MLM matching, we run the shower and then veto events if the hardest shower emission scale $k_{T1} > t_{cut}$



MLM matching for initial state radiation

- For MLM matching, we run the shower and then veto events if the hardest shower emission scale $k_{T1} > t_{cut}$
- The resulting Sudakov suppression from the procedure is
$$(\Delta_{Iq}(t_{\rm cut},t_0))^2(\Delta_q(t_{\rm cut},t_0))^2$$
which is a good enough approximation of the correct expression $(\Delta_{Iq}(t_{\rm cut},t_0))^2\Delta_g(t_2,t_1)(\Delta_q(t_{\rm cut},t_2))^2$ (since the main suppression is from Δ_{Iq})



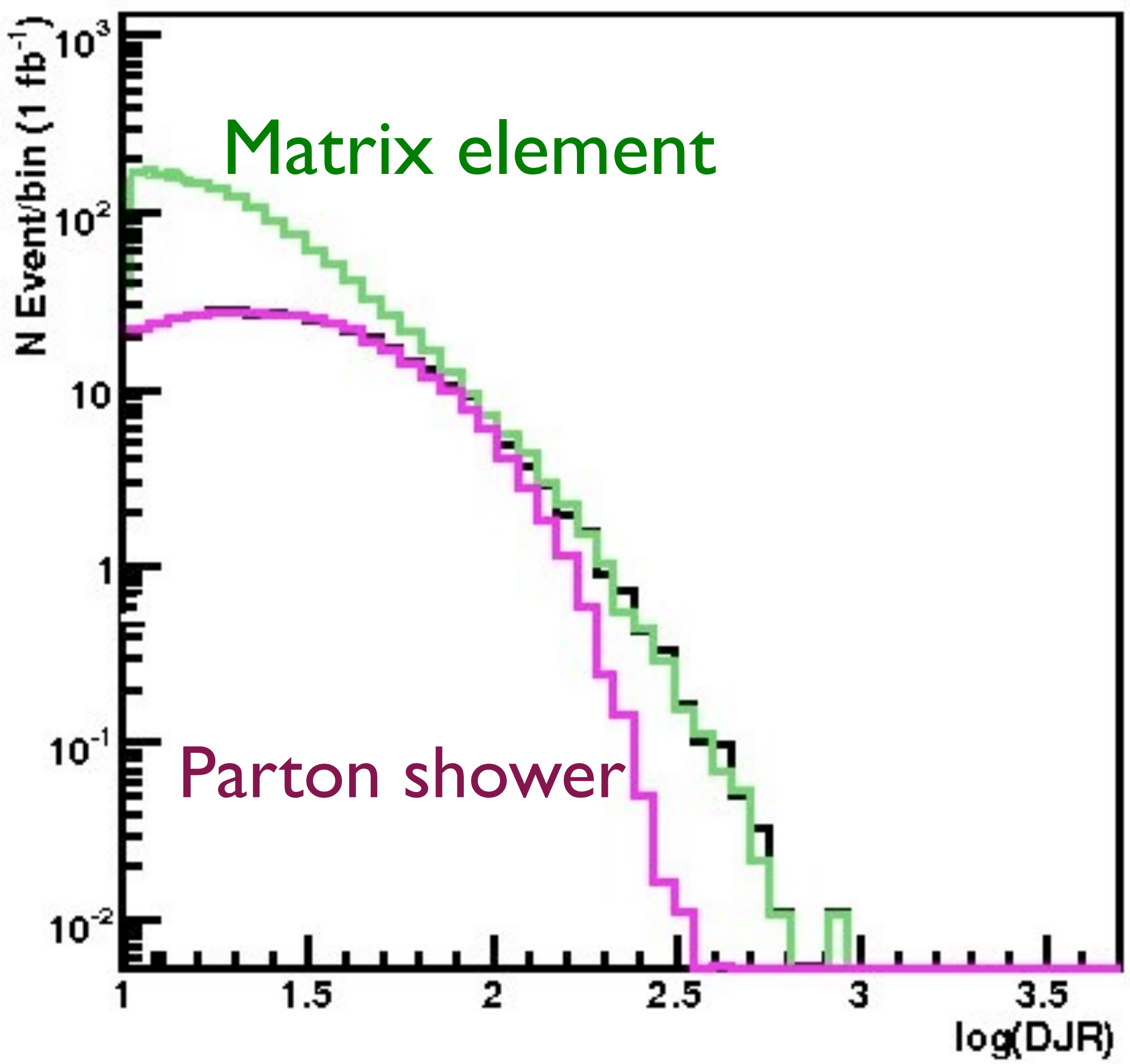
Highest multiplicity sample

- In the previous, assumed we can simulate all parton multiplicities by the ME
- In practice, we can only do limited number of final-state partons with matrix element (up to 4-5 or so)
- For the highest jet multiplicity that we generate with the matrix element, we need to allow additional jets above the matching scale t_{cut}, since we will otherwise not get a jet-inclusive description – but still can't allow PS radiation harder than the ME partons

➡ Need to replace t_{cut} by the clustering scale for the softest ME parton for the highest multiplicity

Back to the “matching goal”

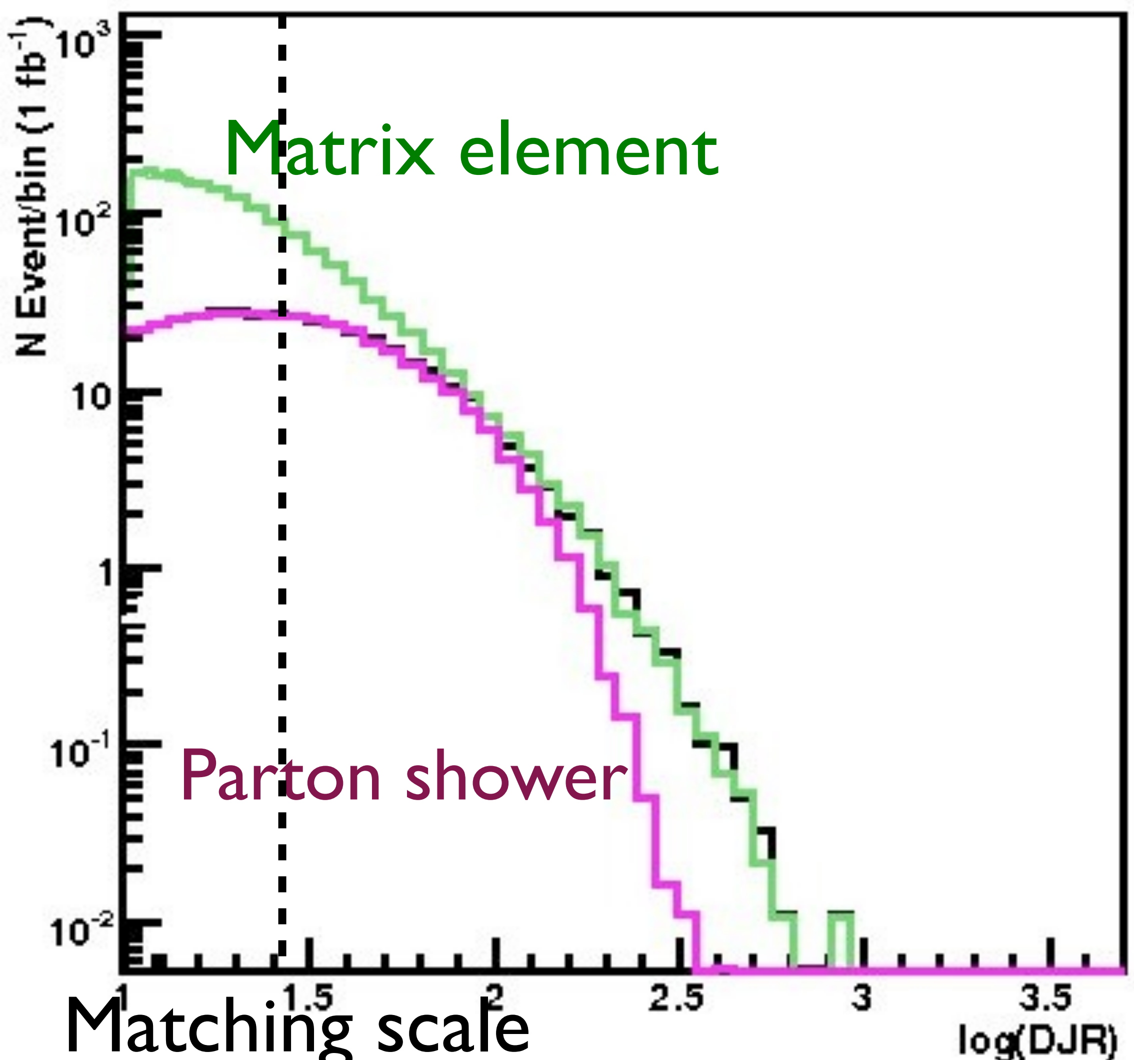
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Back to the "matching goal"

- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions

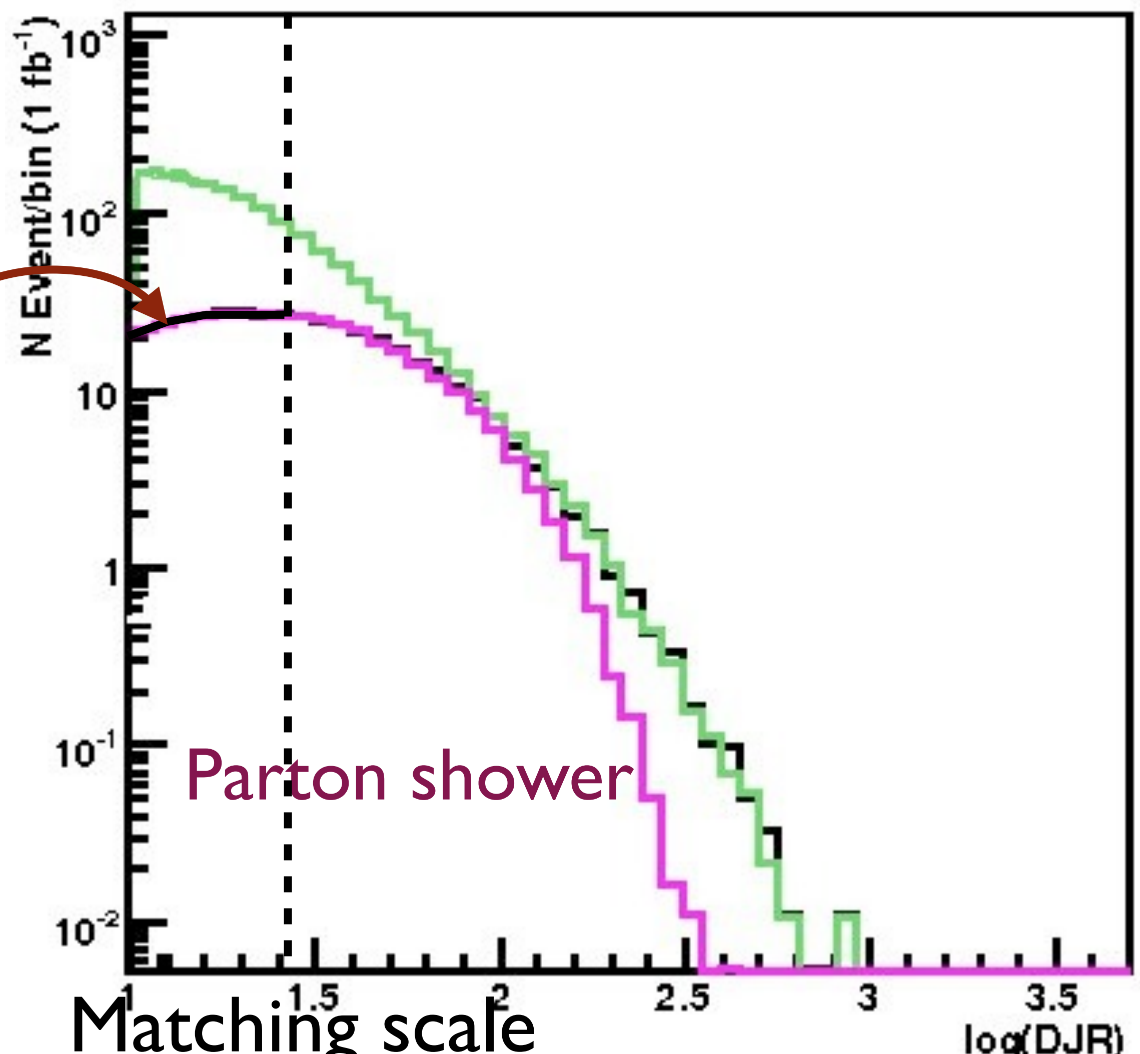


2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Back to the "matching goal"

- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions

Emissions from PS

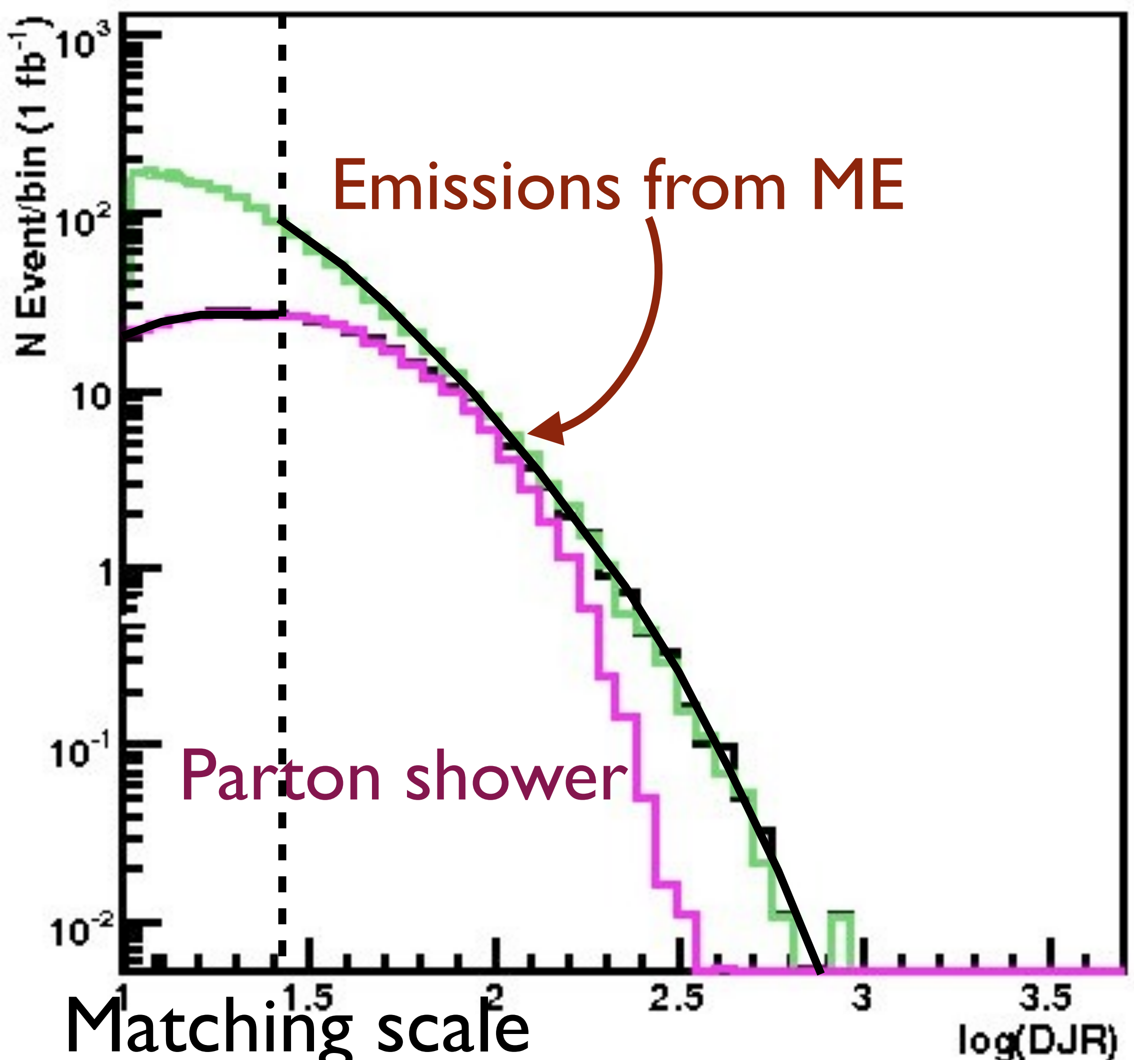


Matching scale

2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Back to the "matching goal"

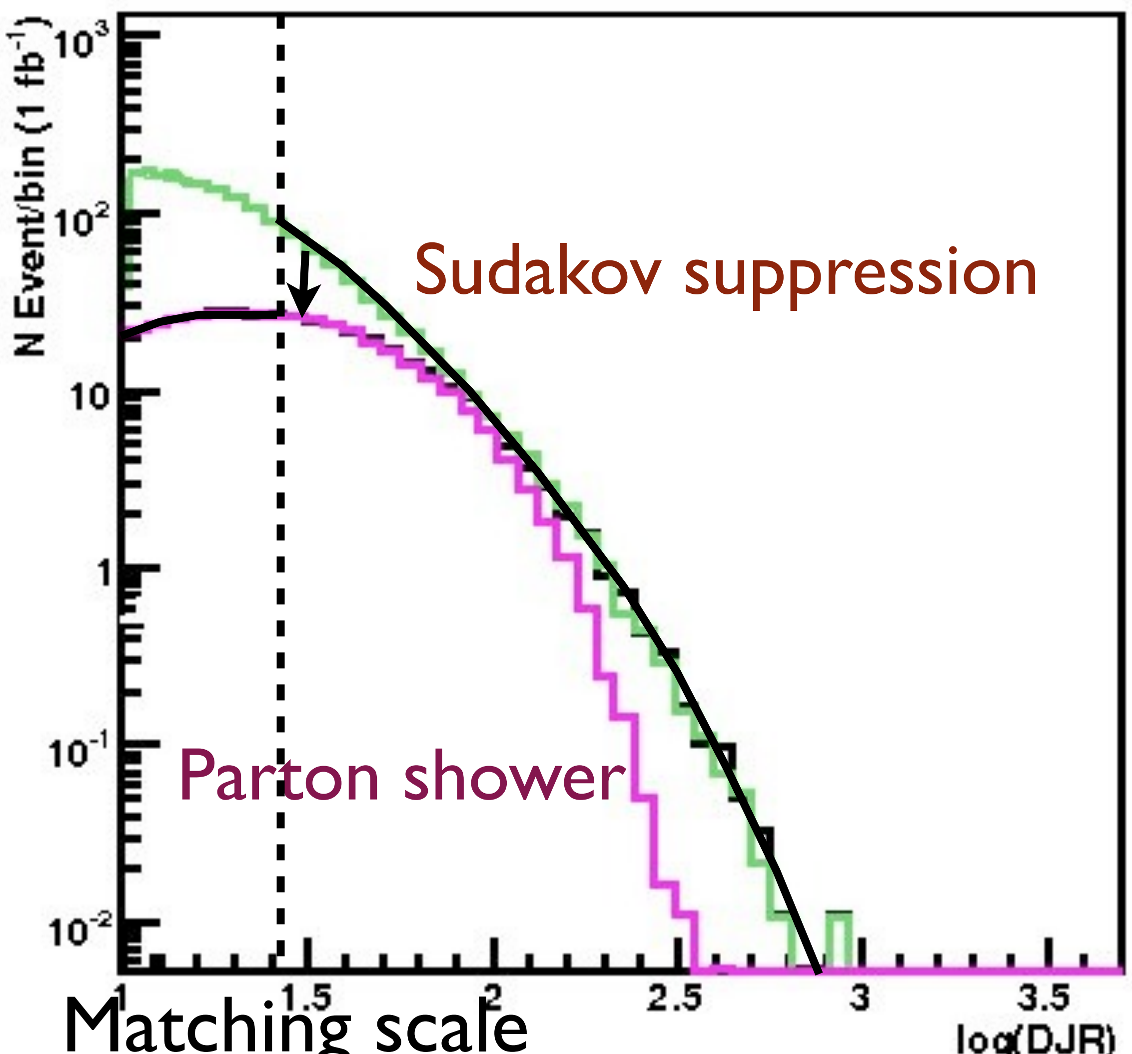
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Back to the “matching goal”

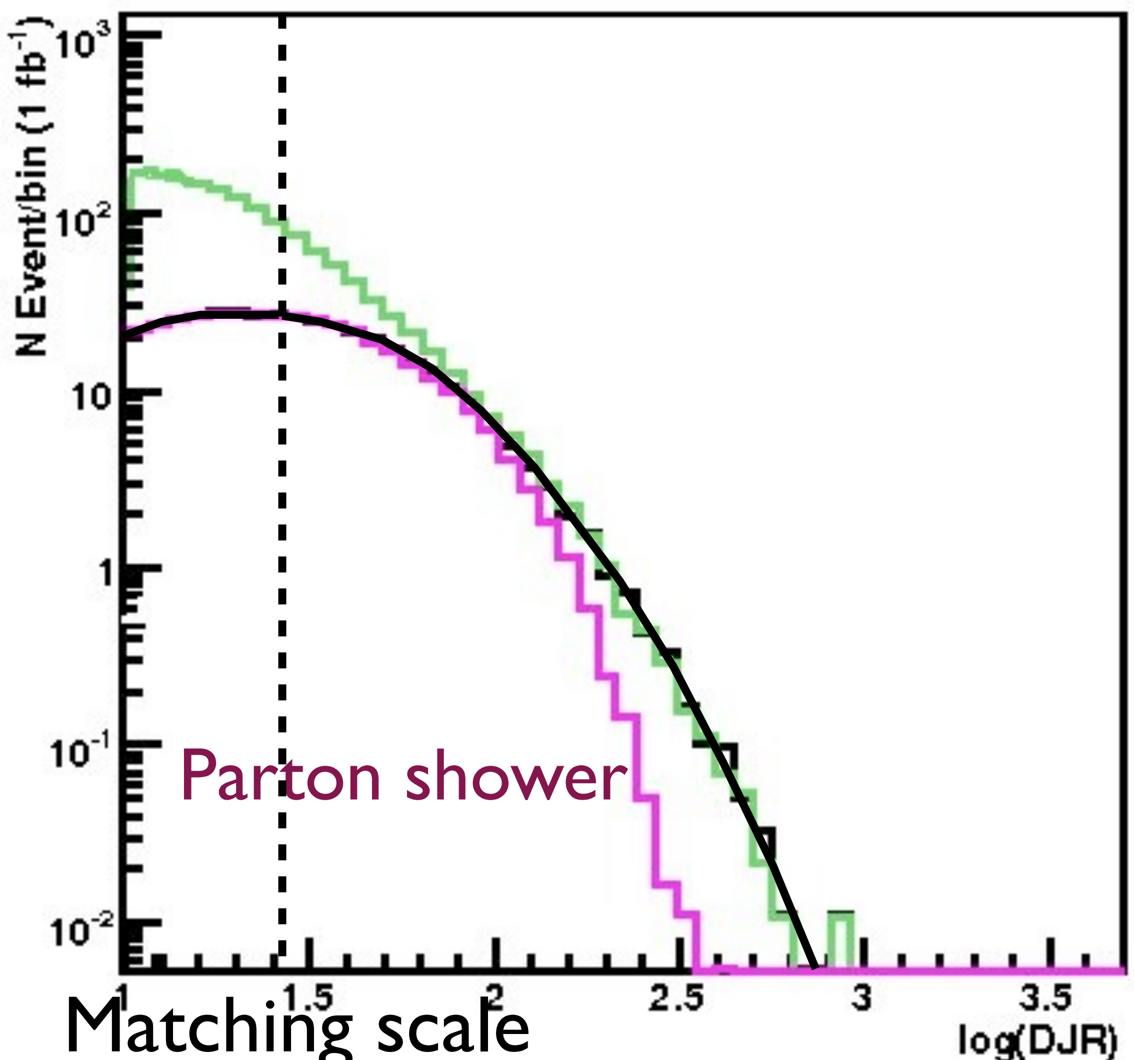
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Back to the "matching goal"

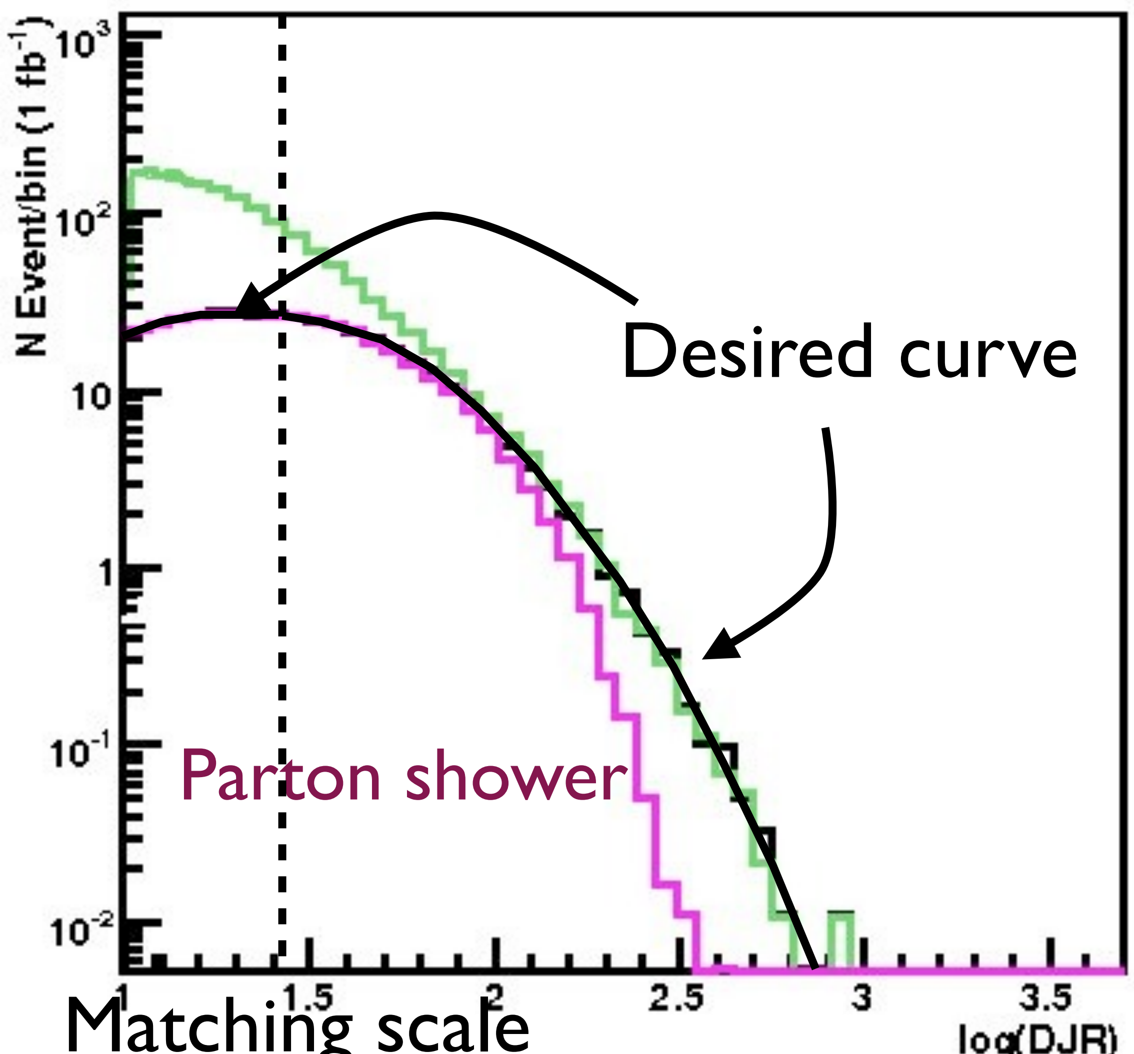
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Back to the "matching goal"

- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions

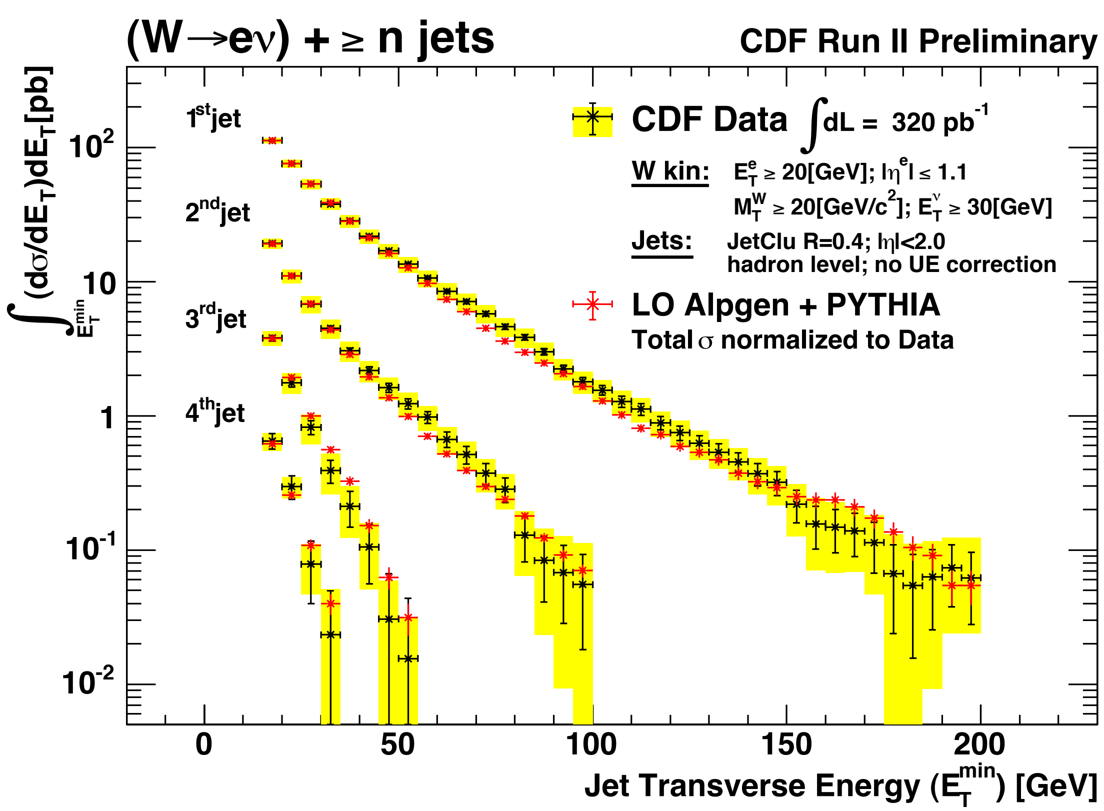


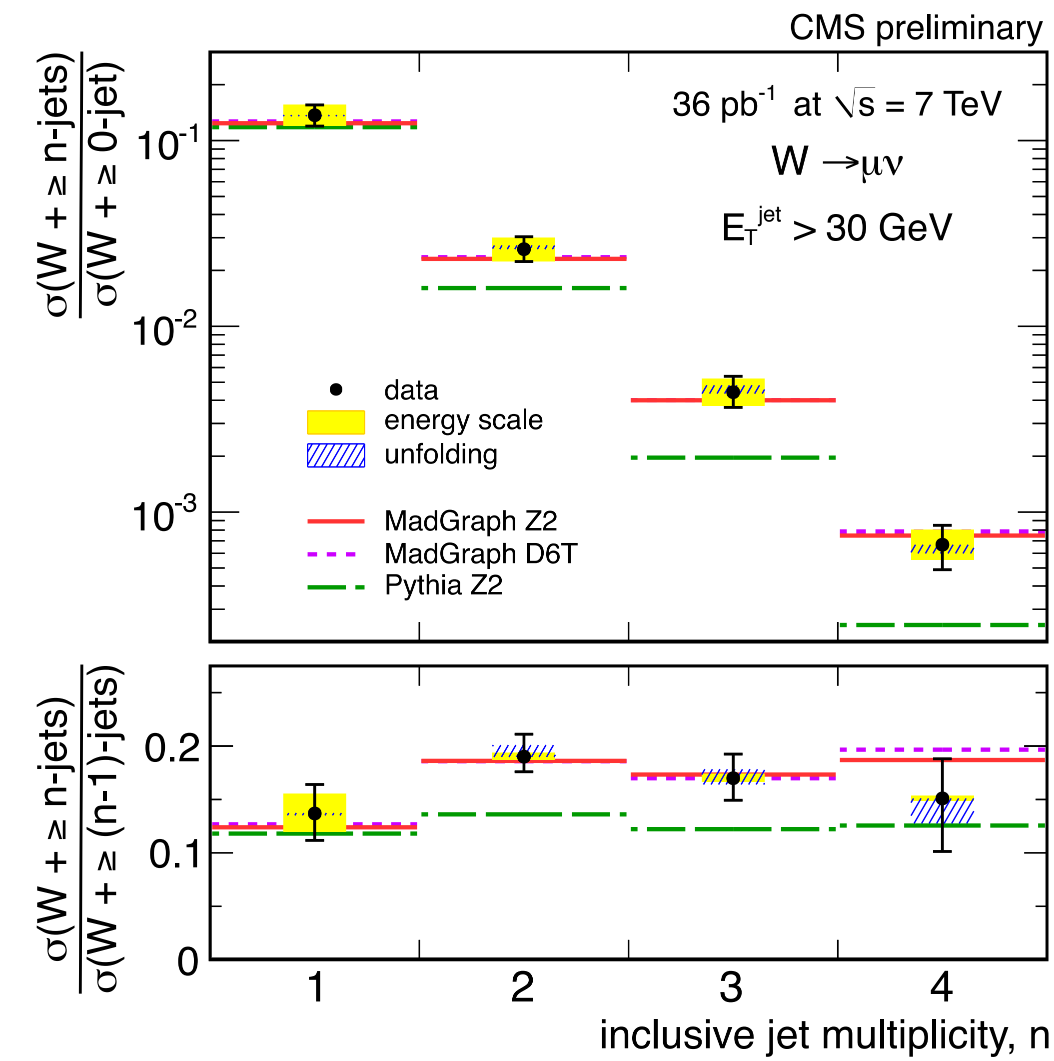
2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Summary of Matching Procedure

1. Generate ME events (with different parton multiplicities) using parton-level cuts ($p_T^{ME}/\Delta R$ or k_T^{ME})
2. Cluster each event and reweight α_s and PDFs based on the scales in the clustering vertices
3. Apply Sudakov factors to account for the required non-radiation above clustering cutoff scale and generate parton shower emissions below clustering cutoff:
    a. (CKKW) Analytical Sudakovs + truncated showers
    b. (CKKW-L) Sudakovs from truncated showers
    c. (MLM) Sudakovs from reclustered shower emissions

Comparing to experiment: W+jets





- Very good agreement at Tevatron (left) and LHC (right)
- Matched samples obtained via different matching schemes (MLM and CKKW) consistent within the expected uncertaintes.
- Pure parton shower (Pythia) doesn't describe the data beyond 1st jet.

MLM matching schemes in MadGraph

[J.A. et al (2007, 2008)]
[J.A. et al (2011)]

MLM matching schemes in MadGraph

[J.A. et al (2007, 2008)]
[J.A. et al (2011)]

In MadGraph, there are 3 different MLM-type matching schemes differing in how to divide ME vs. PS regions:

MLM matching schemes in MadGraph

[J.A. et al (2007, 2008)]
[J.A. et al (2011)]

In MadGraph, there are 3 different MLM-type matching schemes differing in how to divide ME vs. PS regions:

a. Cone jet MLM scheme:
   - Use cuts in p_T (p_T^{ME})and ΔR between partons in ME
   - Cluster events after parton shower using a cone jet algorithm with the same ΔR and $p_T^{match} > p_T^{ME}$
   - Keep event if all jets are matched to ME partons (i.e., all ME partons are within ΔR of a jet)

MLM matching schemes in MadGraph

[J.A. et al (2007, 2008)]
[J.A. et al (2011)]

In MadGraph, there are 3 different MLM-type matching schemes differing in how to divide ME vs. PS regions:

a. Cone jet MLM scheme:
   - Use cuts in p_T (p_T^{ME})and ΔR between partons in ME
   - Cluster events after parton shower using a cone jet algorithm with the same ΔR and $p_T^{match} > p_T^{ME}$
   - Keep event if all jets are matched to ME partons (i.e., all ME partons are within ΔR of a jet)
b. k_T-jet MLM scheme:
   - Use cut in the Durham k_T in ME
   - Cluster events after parton shower using the same k_T clustering algorithm into k_T jets with $k_T^{match} > k_T^{ME}$
   - Keep event if all jets are matched to ME partons (i.e., all partons are within k_T^{match} to a jet)

MLM matching schemes in MadGraph

c. Shower-k_T scheme:
- Use cut in the Durham k_T in ME
- After parton shower, get information from the PS generator about the k_T^{PS} of the hardest shower emission
- Keep event if $k_T^{PS} < k_T^{match}$

How to do matching in MadGraph+Pythia

Example: Simulation of pp→W with 0, 1, 2 jets (comfortable on a laptop)

```
mg5> generate p p > w+, w+ > l+ vl @0
mg5> add process p p > w+ j, w+ > l+ vl @1
mg5> add process p p > w+ j j, w+ > l+ vl @2
mg5> output
```

```
In run_card.dat:
…
  1 = ickkw
…
  0 = ptj
…
 15 = xqcut
```

- `1 = ickkw` ← Matching on
- `0 = ptj` ← No cone matching
- `15 = xqcut` ← k_T matching scale

Matching automatically done when run through MadEvent and Pythia!

How to do matching in MadGraph+Pythia

- By default, k_T-MLM matching is run if xqcut > 0, with the matching scale QCUT = max(xqcut*1.4, xqcut+10)
- For shower-kT, by default QCUT = xqcut
- If you want to change the Pythia setting for matching scale or switch to shower-k_T matching:

```
In pythia_card.dat:
…
! This sets the matching scale, needs to be > xqcut
QCUT = 30
! This switches from kT-MLM to shower-kT matching
! Note that MSTP(81)>=20 needed (pT-ordered shower)
SHOWERKT = T
```

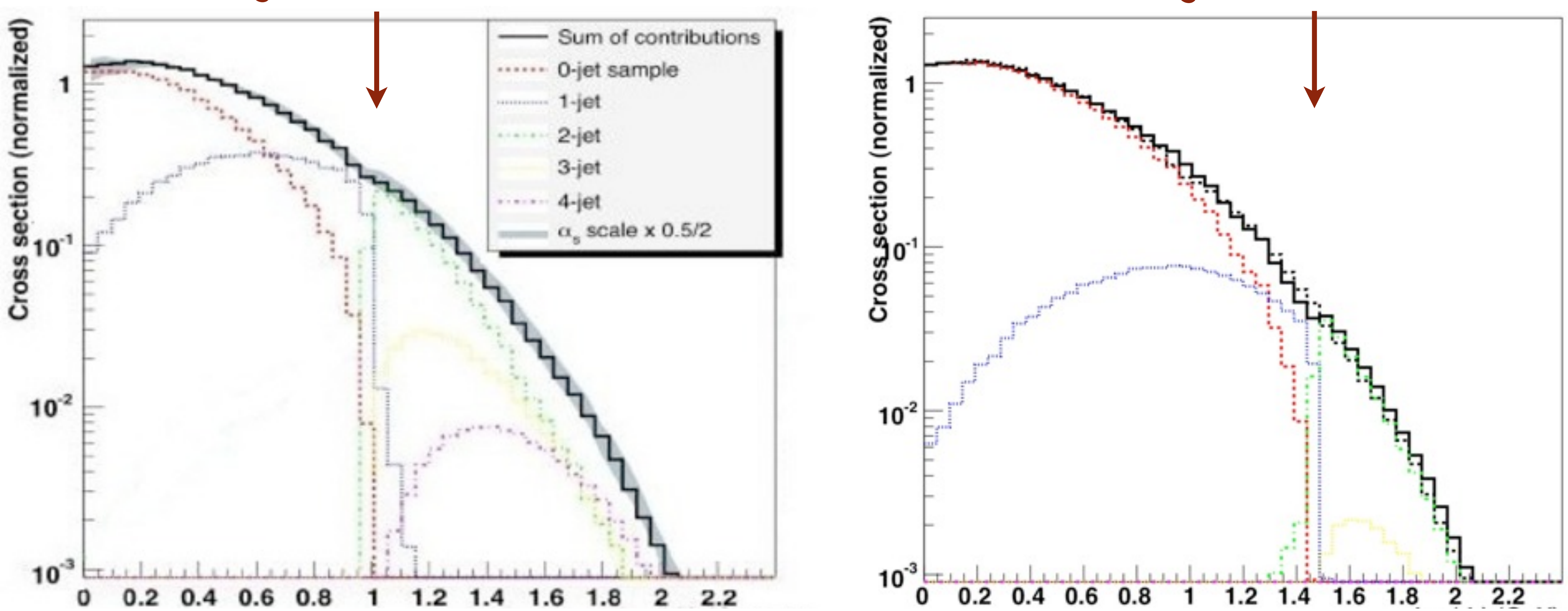
How to do validate the matching

- The matching scale (QCUT) should typically be chosen around 1/6-1/2 x hard scale (so xqcut correspondingly lower)
- The matched cross section (for X+0,1,... jets) should be close to the unmatched cross section for the 0-jet sample (found on the process HTML page)
- The differential jet rate plots should be smooth
- When QCUT is varied (within the region of validity), the matched cross section or differential jet rates should not vary significantly

Matching validation

W+jets production at the Tevatron for MadGraph+Pythia
(k_T-jet MLM scheme, q^2-ordered Pythia showers)

Q^{match} = 10 GeV Q^{match} = 30 GeV



log(Differential jet rate for 1 → 2 radiated jets ~ pT(2nd jet))

Jet distributions smooth, and stable when we vary the matching scale!

Summary

Summary

- Despite the apparent enormous complexity of simulation of complete collider events, nature has kindly allowed us to factorize the simulation into separate steps

Summary

- Despite the apparent enormous complexity of simulation of complete collider events, nature has kindly allowed us to factorize the simulation into separate steps
- The Monte Carlo method allows us to step-by-step simulate hard scattering, parton shower, particle decays, hadronization, and underlying event

Summary

- Despite the apparent enormous complexity of simulation of complete collider events, nature has kindly allowed us to factorize the simulation into separate steps
- The Monte Carlo method allows us to step-by-step simulate hard scattering, parton shower, particle decays, hadronization, and underlying event
- Jet matching between matrix elements and parton showers gives crucial improvement of simulation of background as well as signal processes

Summary

- Despite the apparent enormous complexity of simulation of complete collider events, nature has kindly allowed us to factorize the simulation into separate steps
- The Monte Carlo method allows us to step-by-step simulate hard scattering, parton shower, particle decays, hadronization, and underlying event
- Jet matching between matrix elements and parton showers gives crucial improvement of simulation of background as well as signal processes
- Running matching with MadGraph + Pythia is very easy, but the results should always be checked for consistency