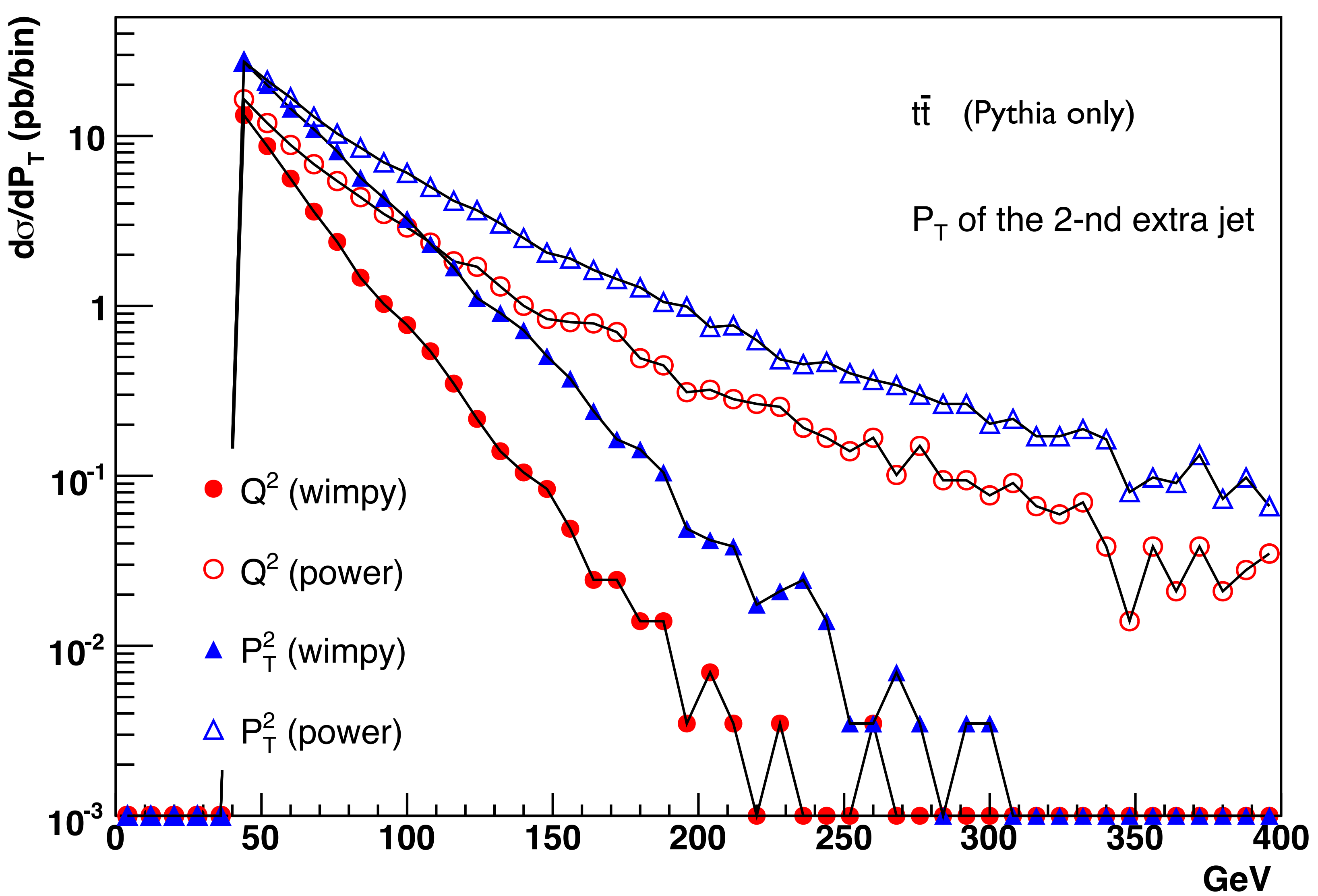
Matching/Merging

Olivier Mattelaer
IPPP/Durham

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)



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

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!

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

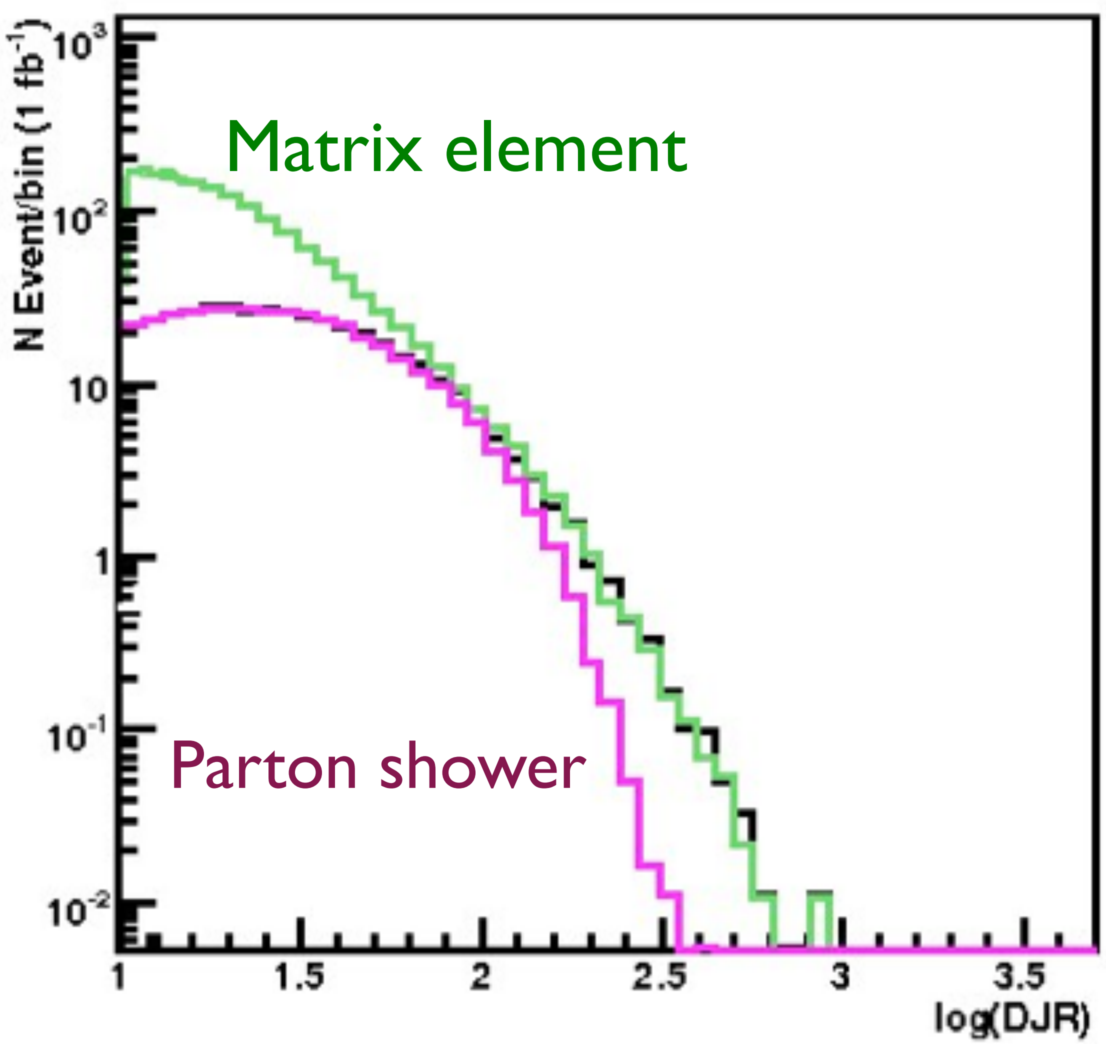
Shower MC

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

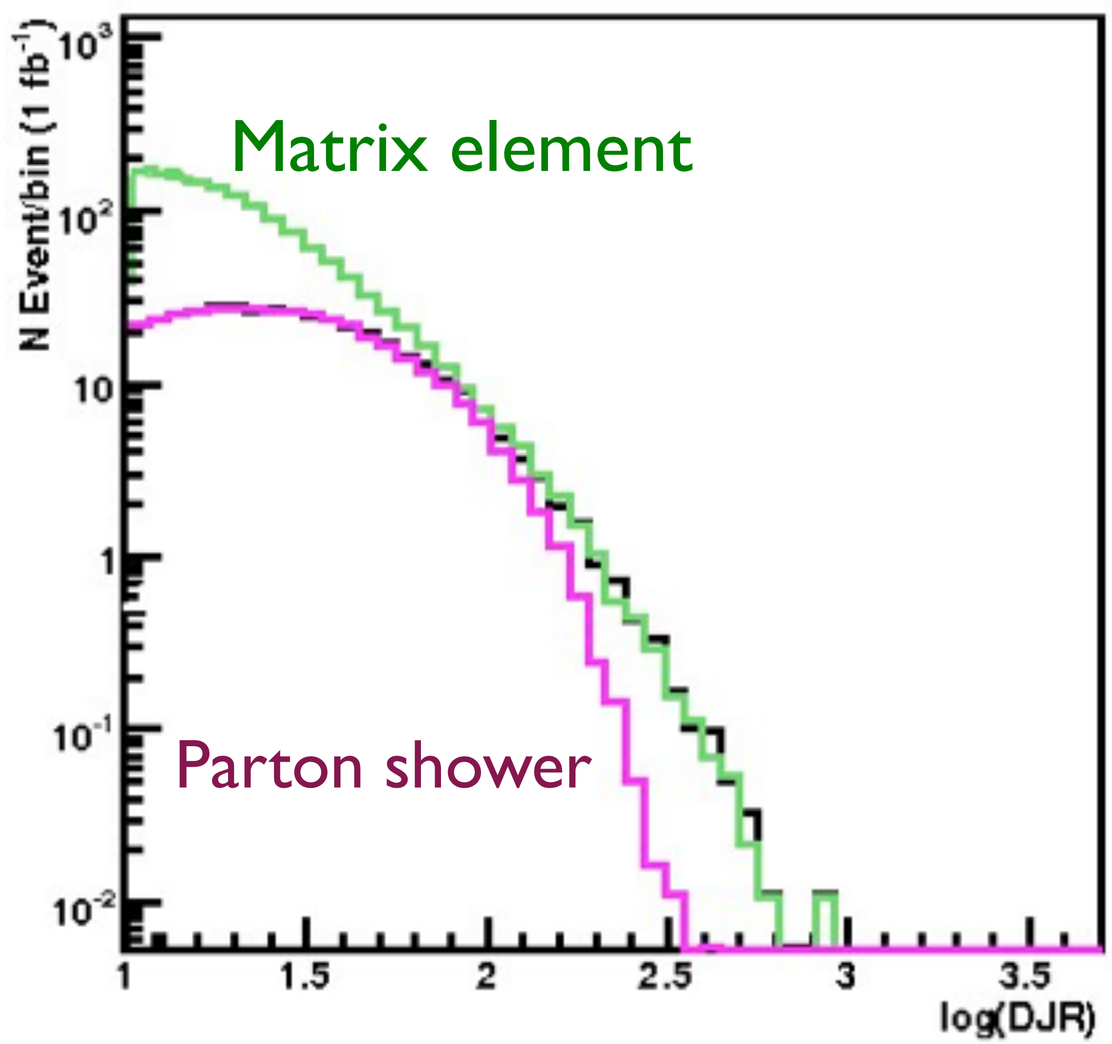
Goal for ME-PS merging/matching



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

Goal for ME-PS merging/matching

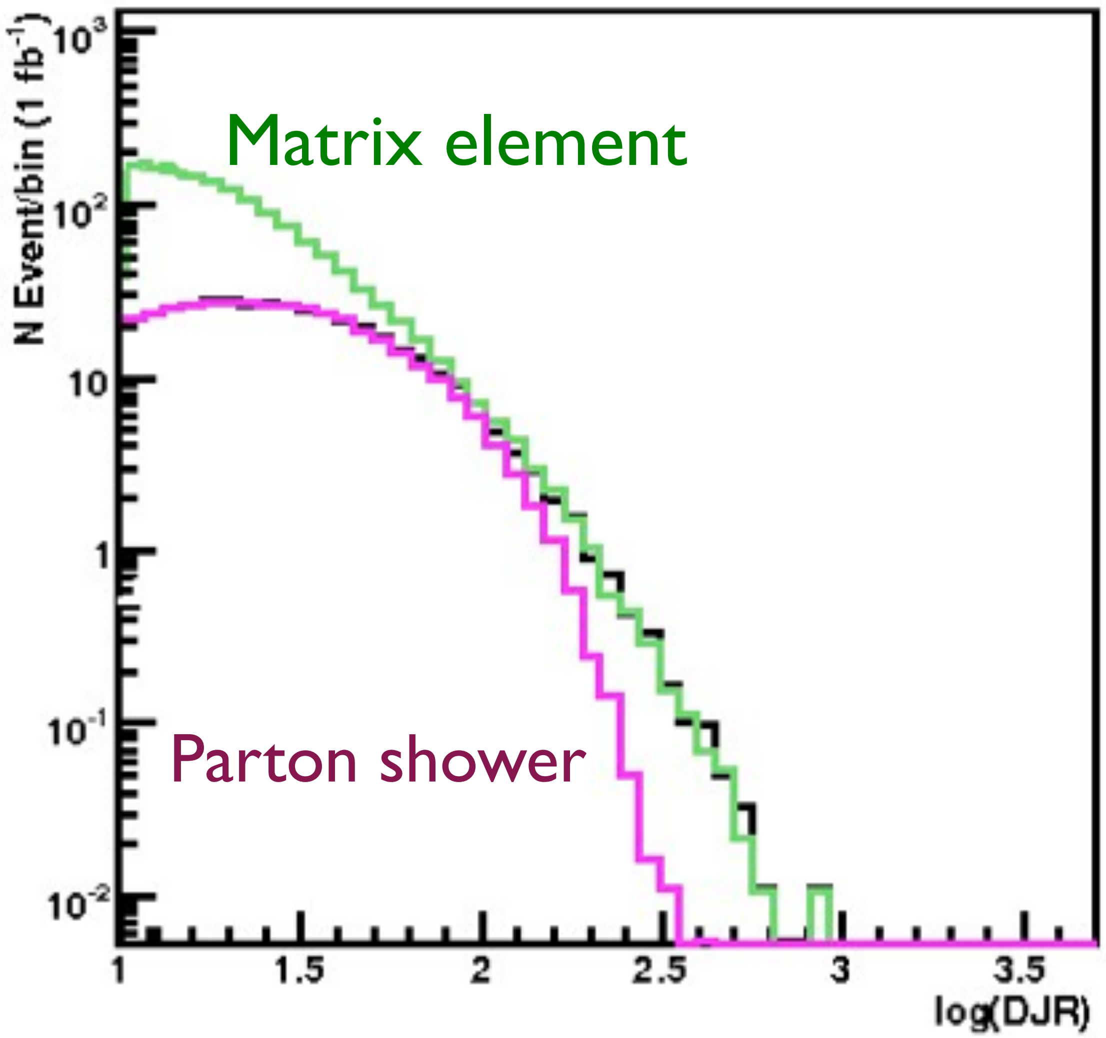
- Regularization of matrix element divergence



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

Goal for ME-PS merging/matching

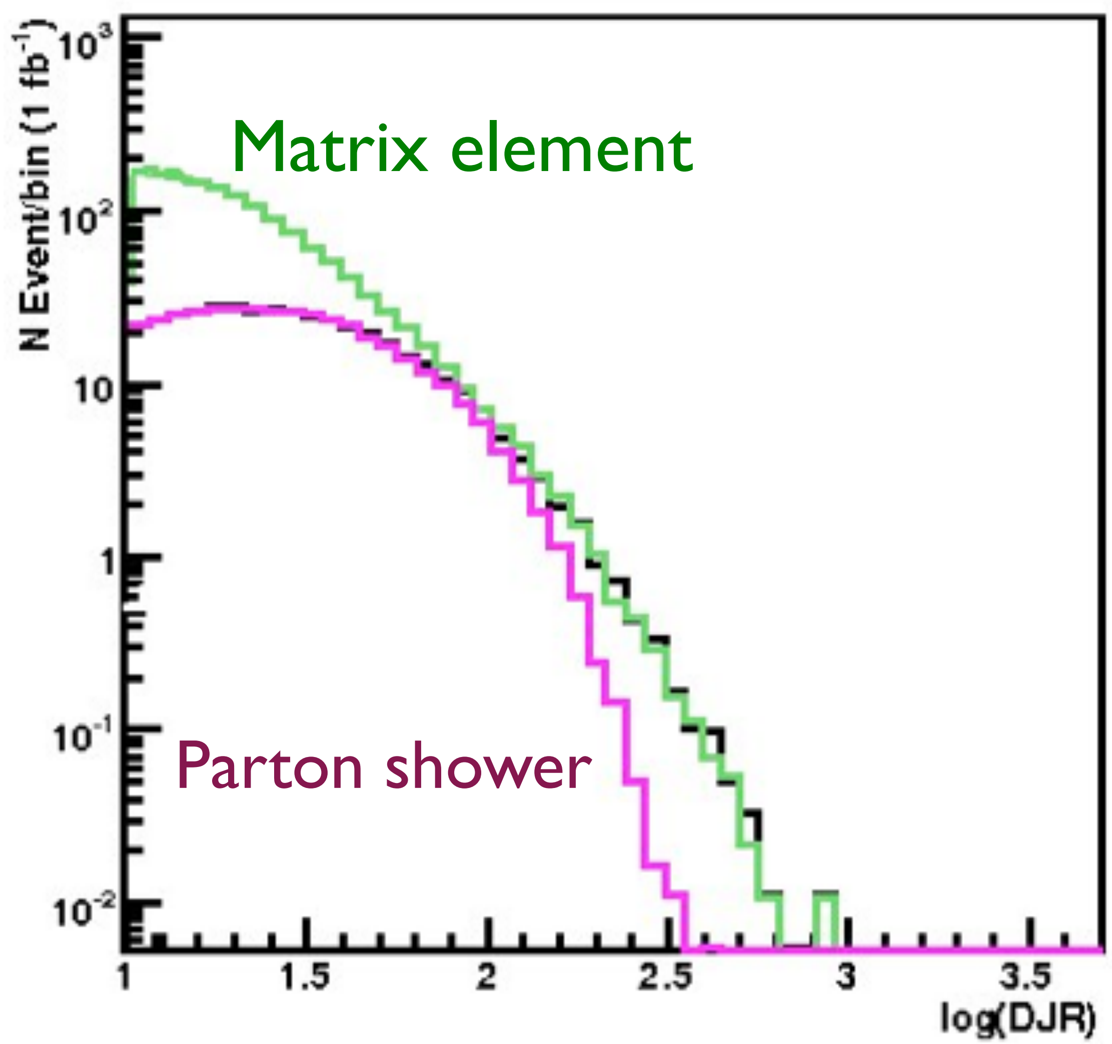
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta



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

Goal for ME-PS merging/matching

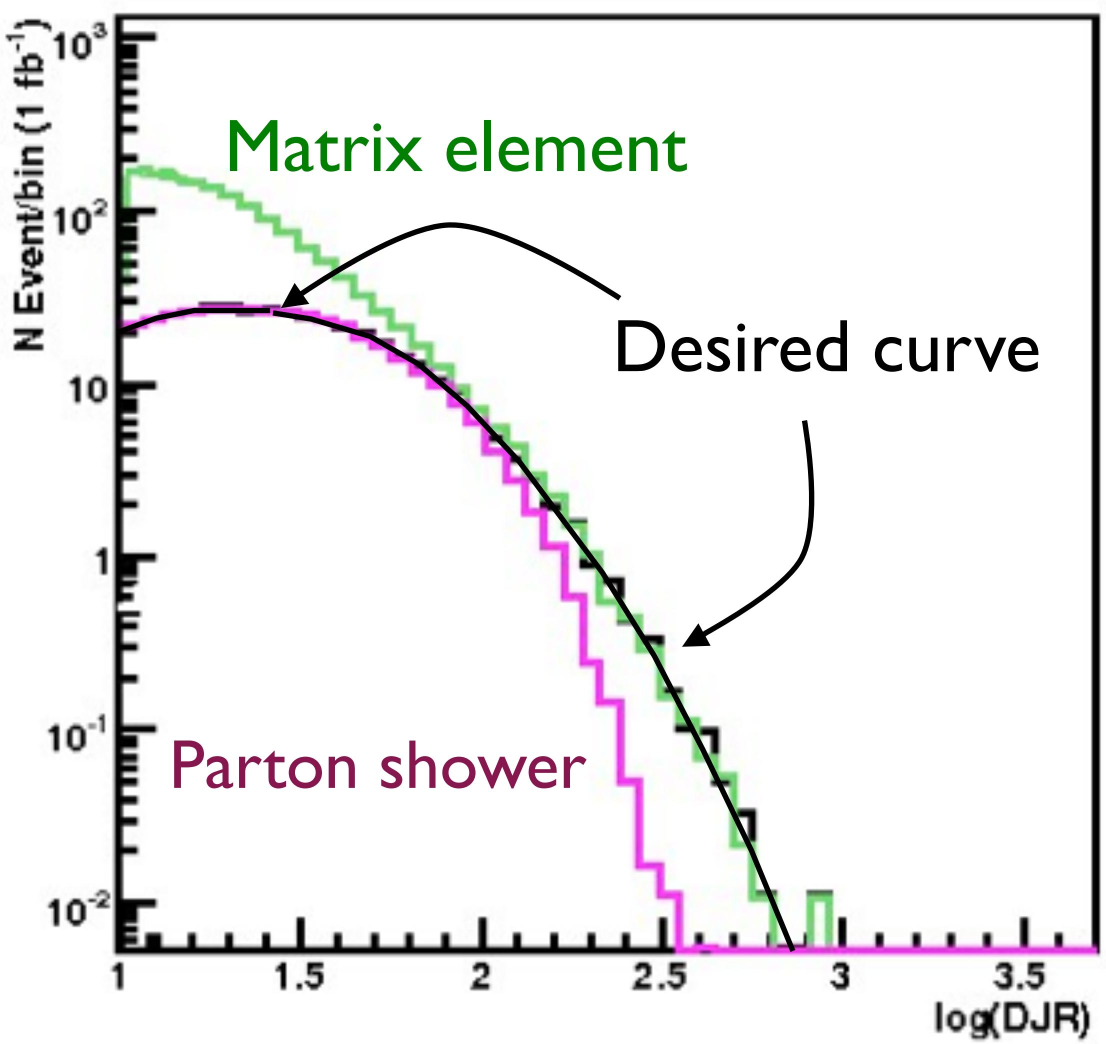
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



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

Goal for ME-PS merging/matching

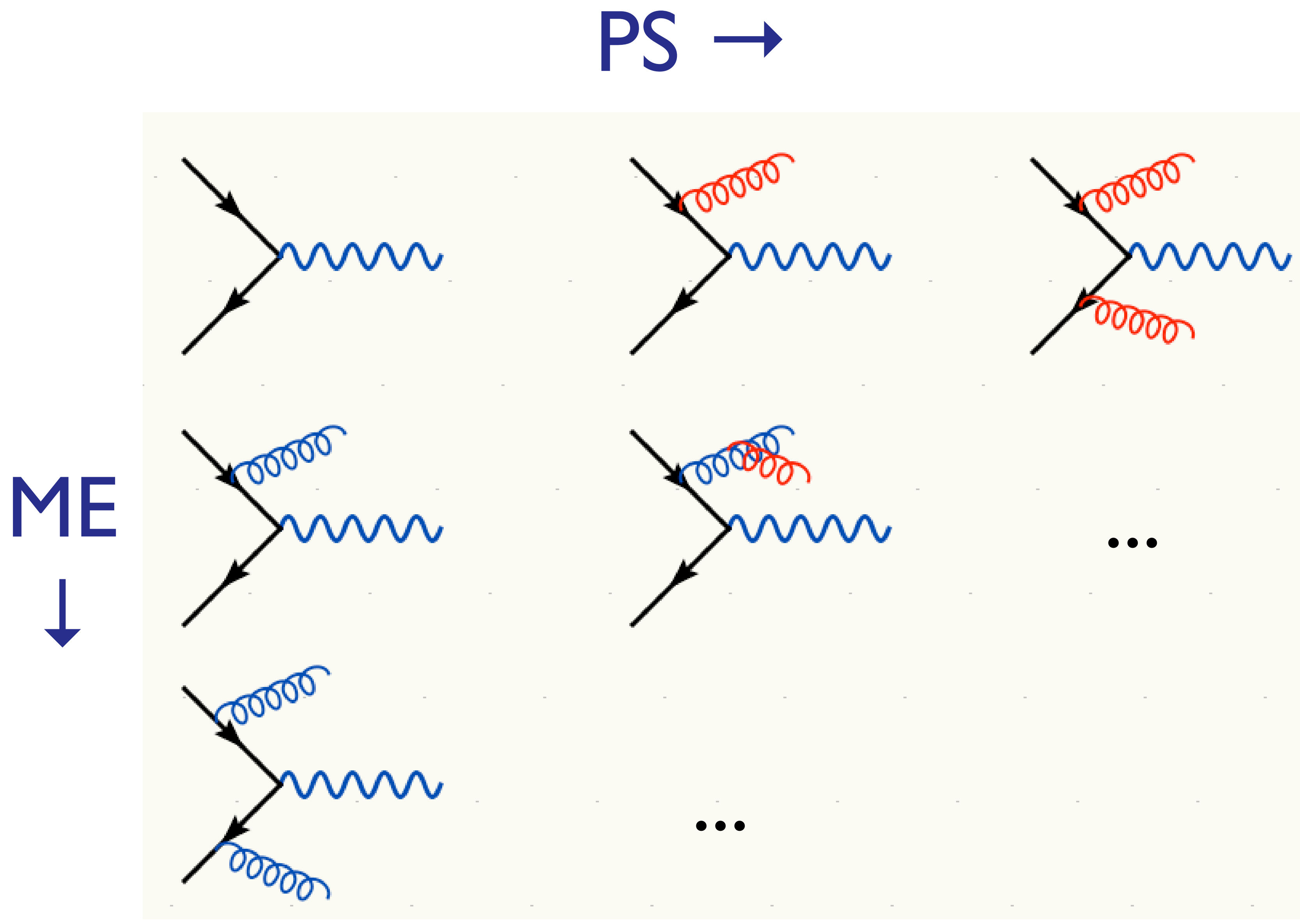
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



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

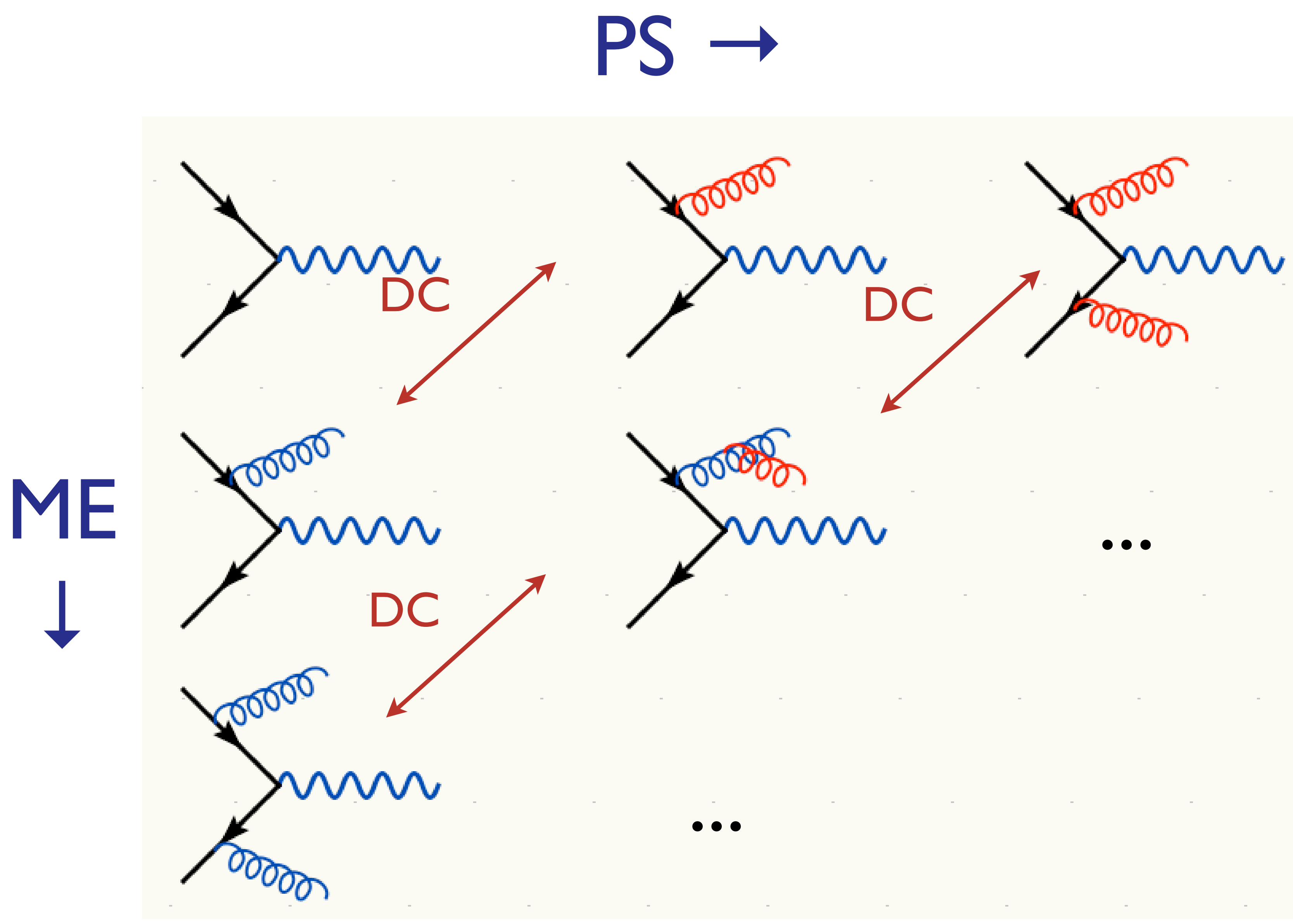
Merging ME with PS

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



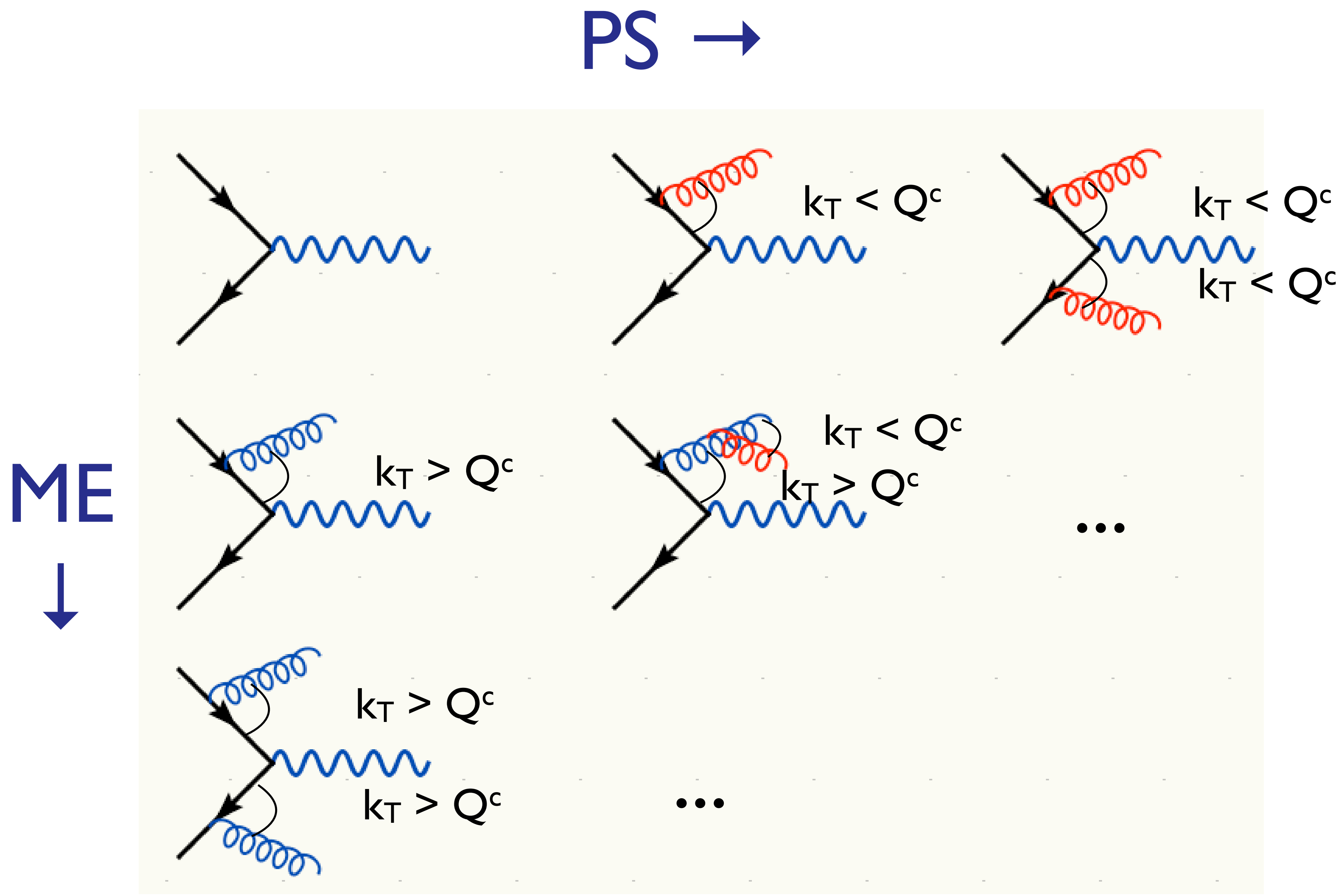
Merging ME with PS

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



Merging ME with PS

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

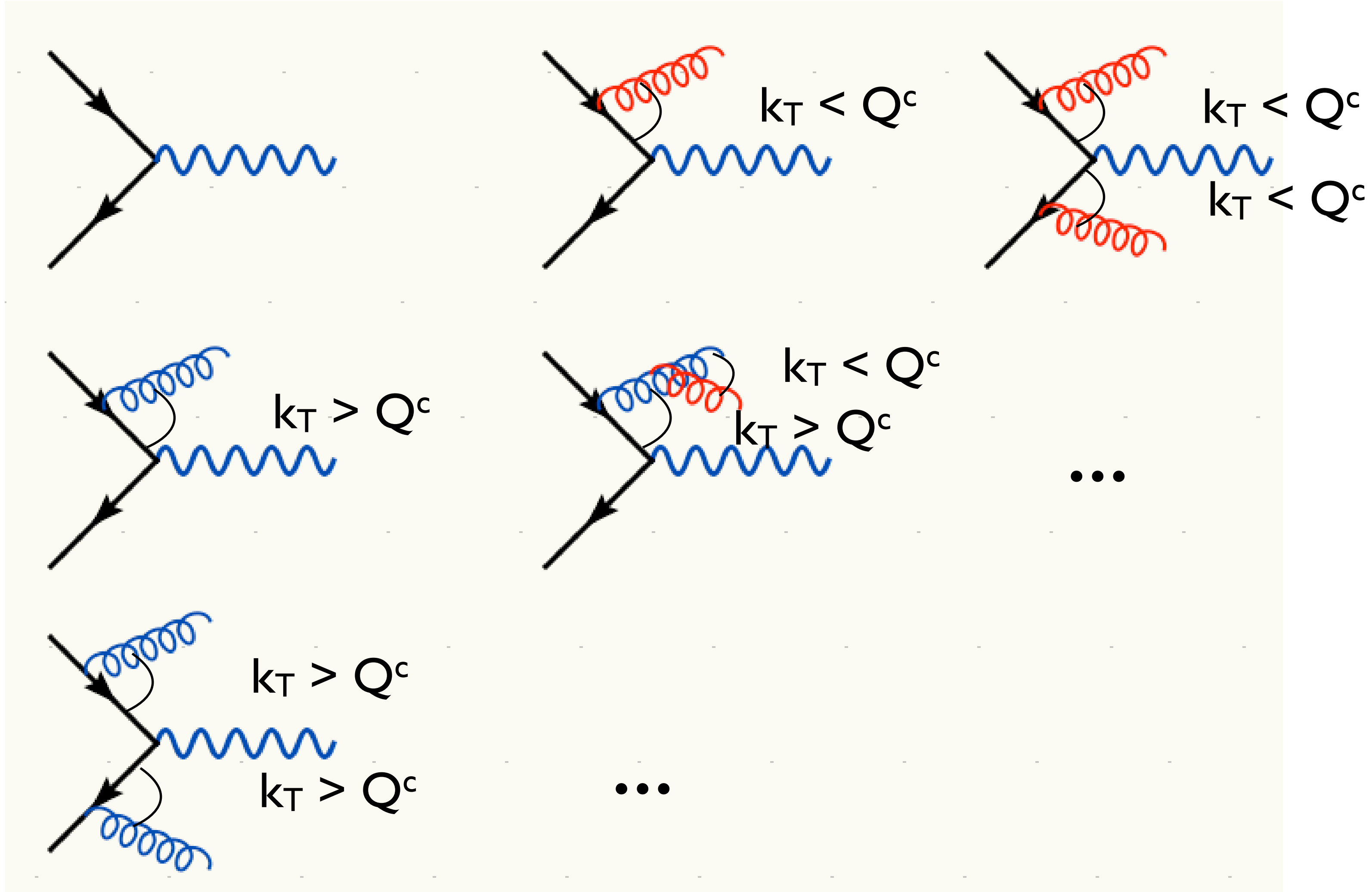


Merging ME with PS

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

PS →

ME ↓

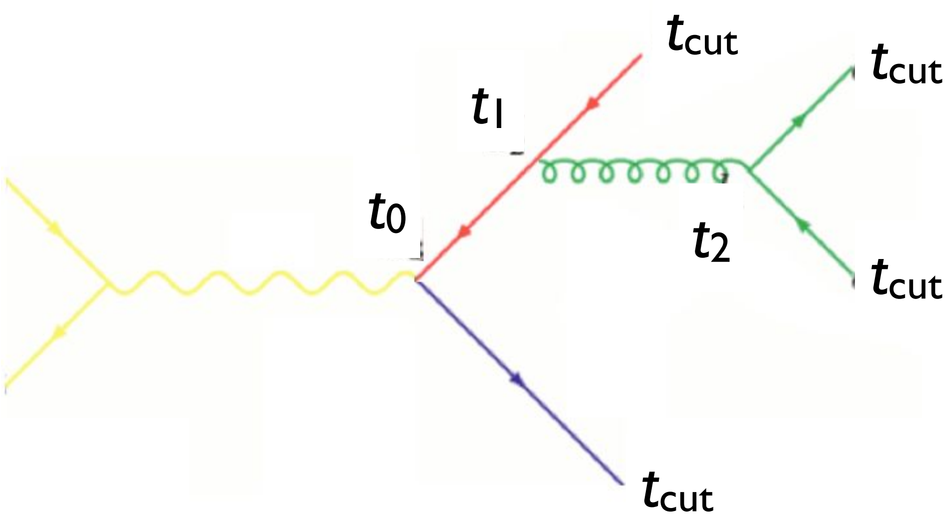


Double counting between ME and PS easily avoided using phase space cut between the two: PS below cutoff, ME above cutoff.

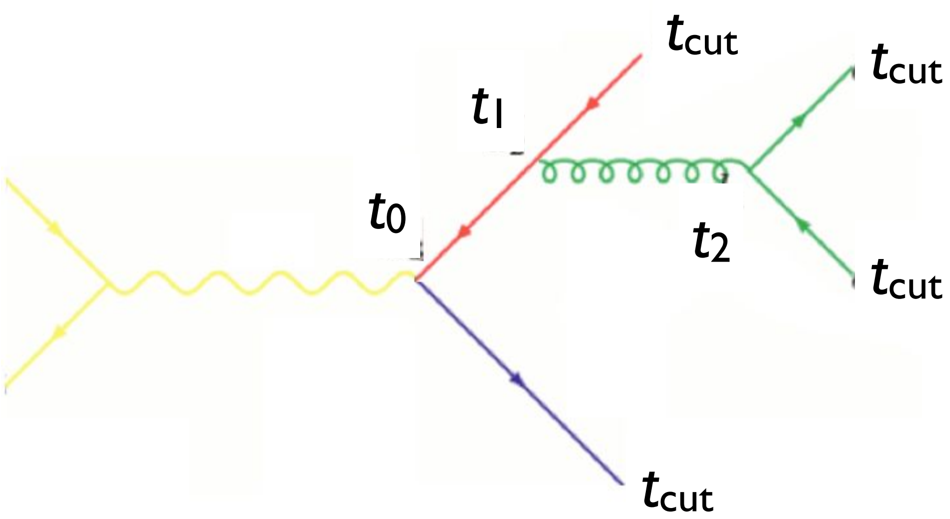
Merging ME with PS

- 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

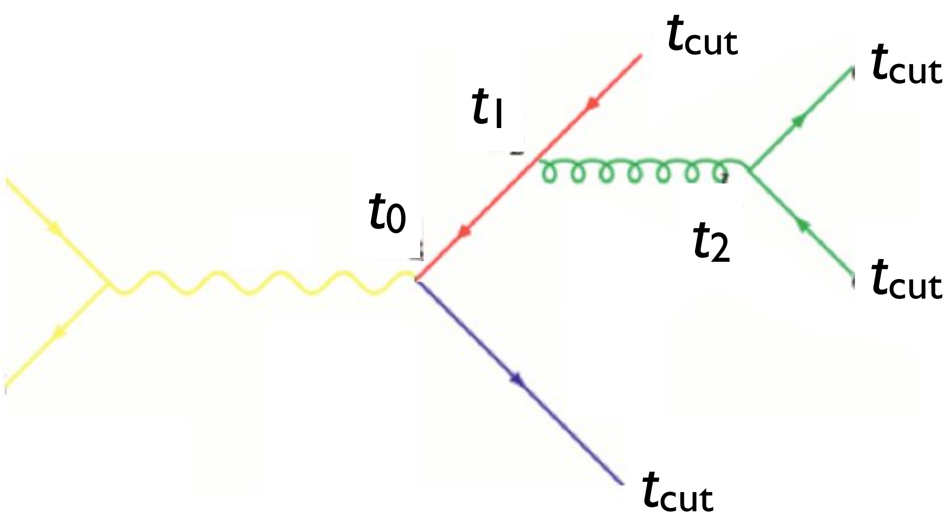


Merging ME with PS



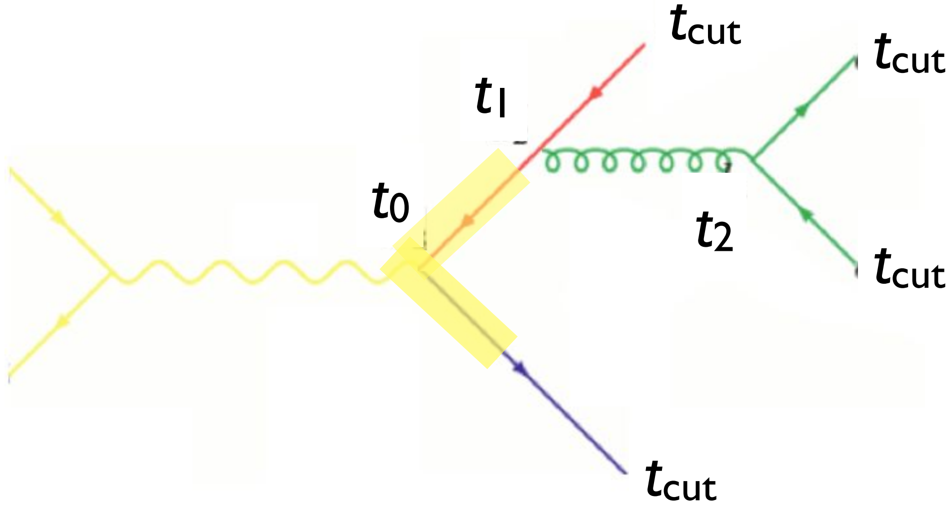
- How does the PS generate the configuration above?

Merging ME with PS



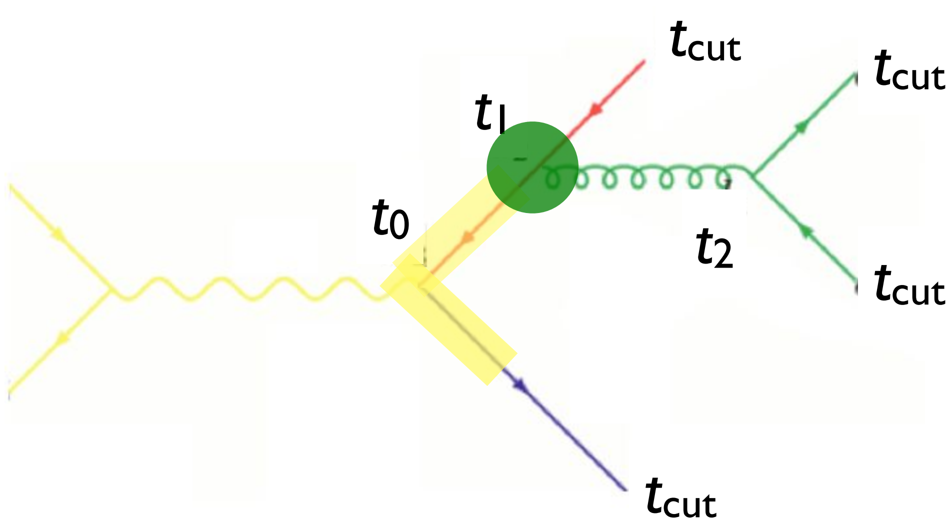
- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

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



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

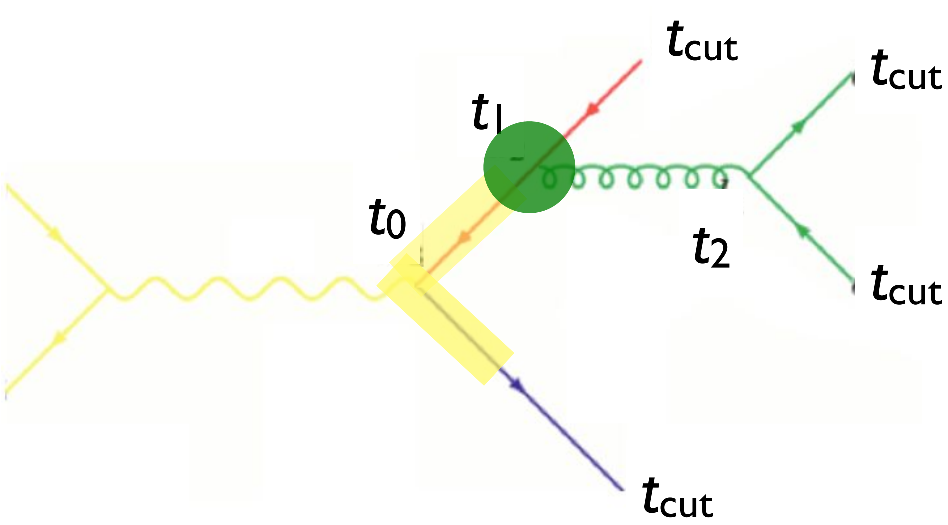
$$(\Delta_q(t_1, t_0))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z)$$



- 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



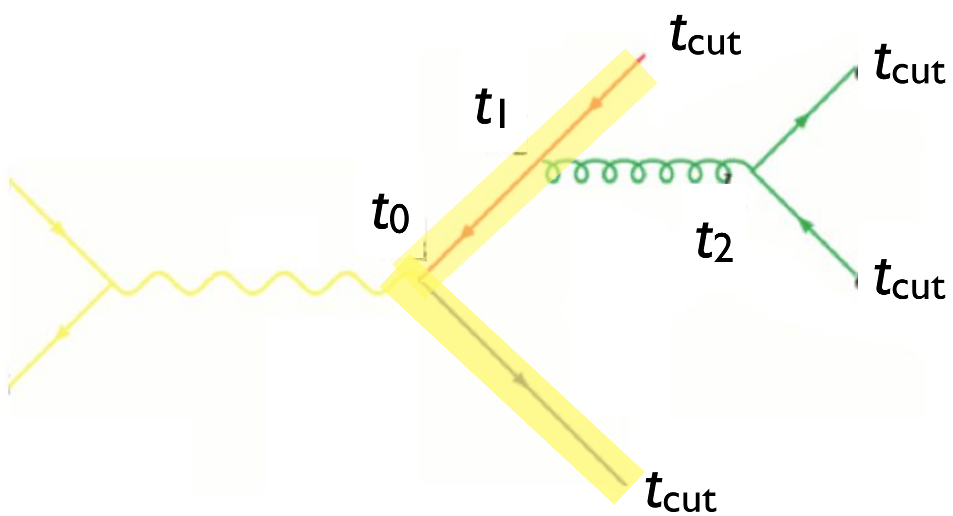
- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

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

and for the whole tree

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

Merging ME with PS

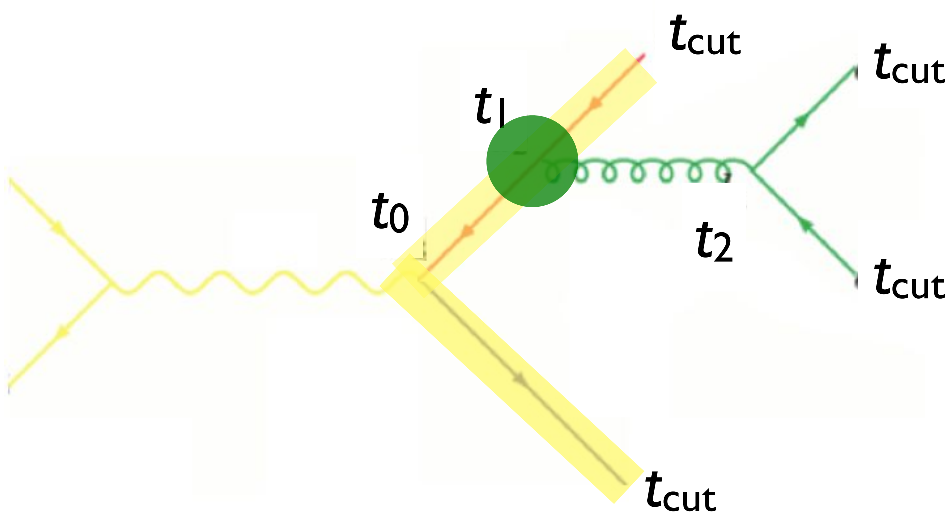


- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

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

and for the whole tree

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



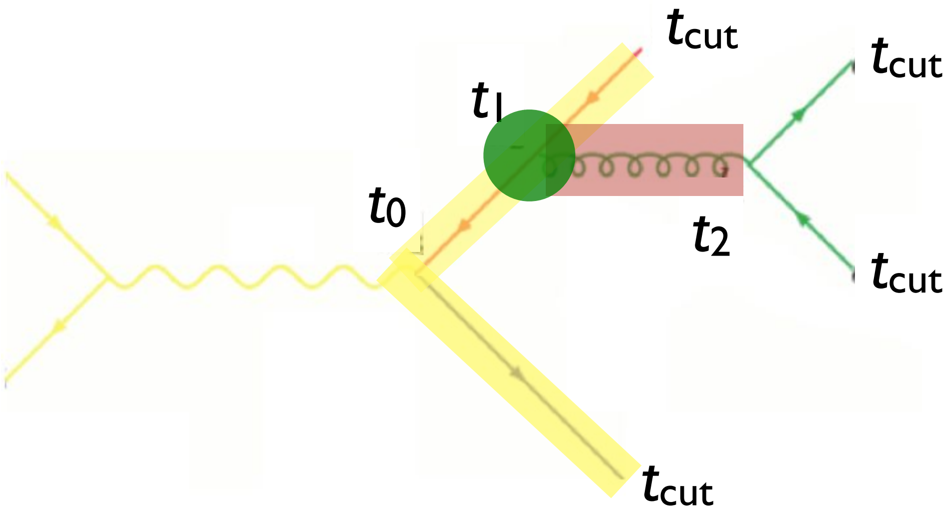
- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

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

and for the whole tree

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

Merging ME with PS

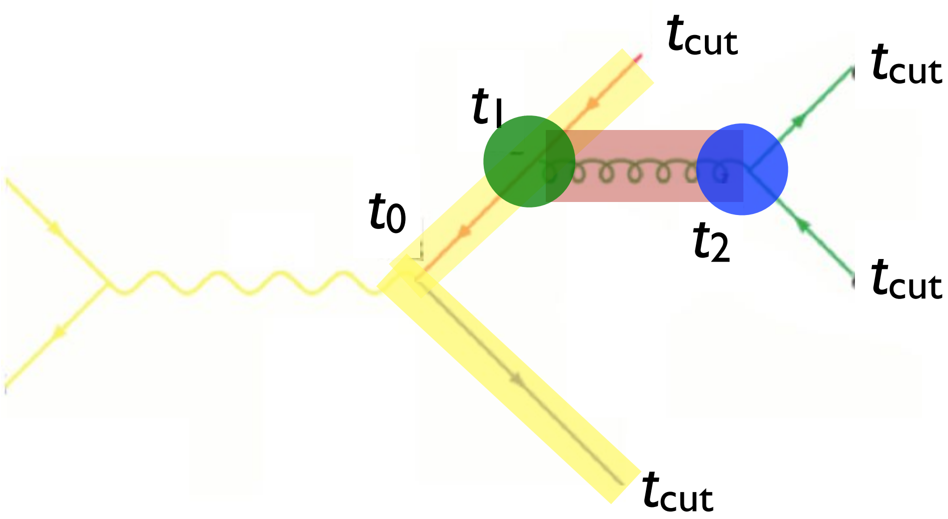


- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

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

and for the whole tree

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

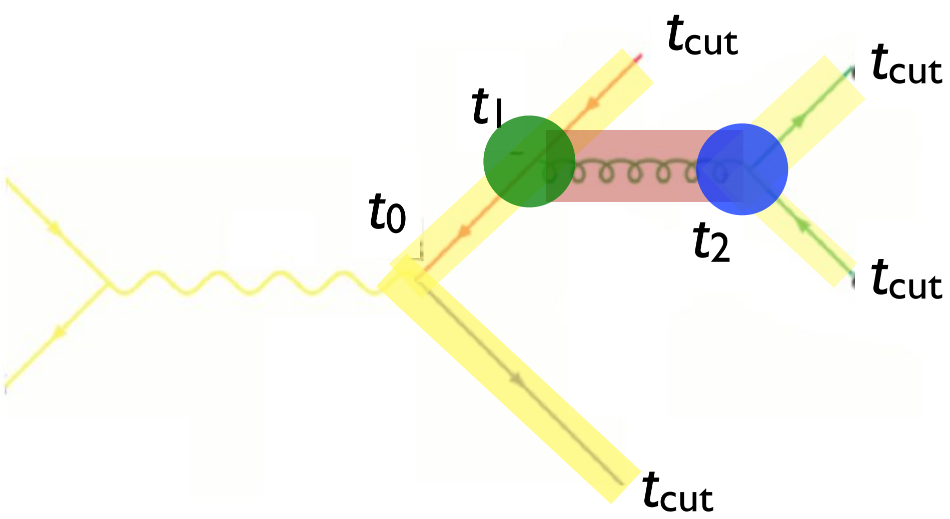


- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

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

and for the whole tree

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



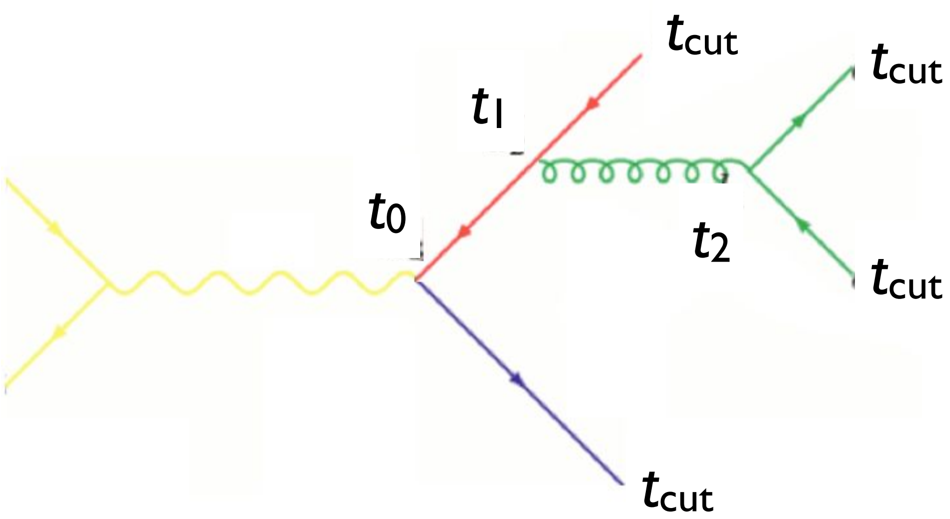
- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

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

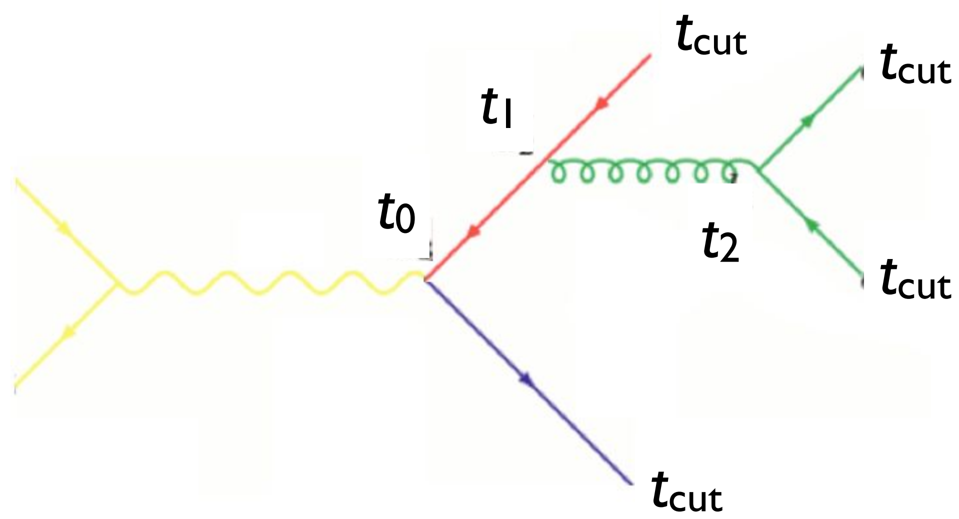
and for the whole tree

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

Merging ME with PS



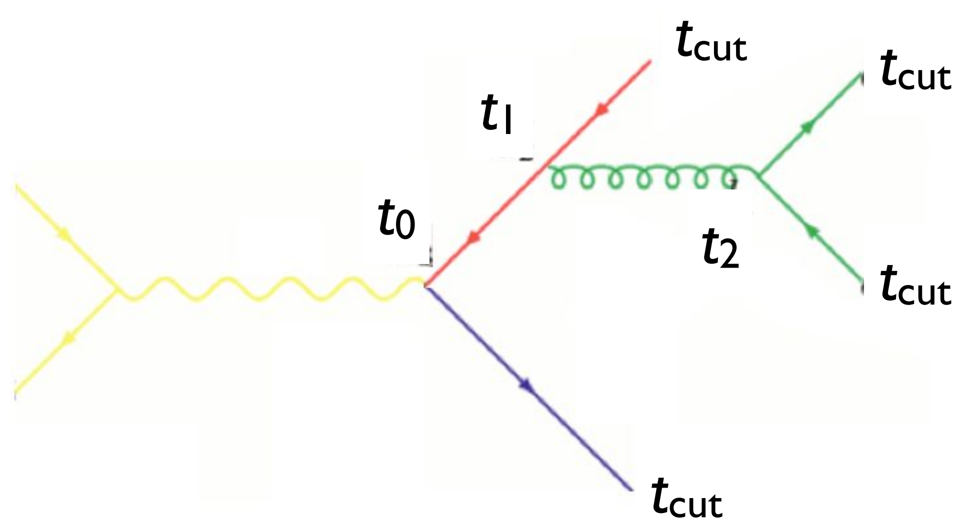
$$(\Delta_q(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z) \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$



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

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

Merging ME with PS

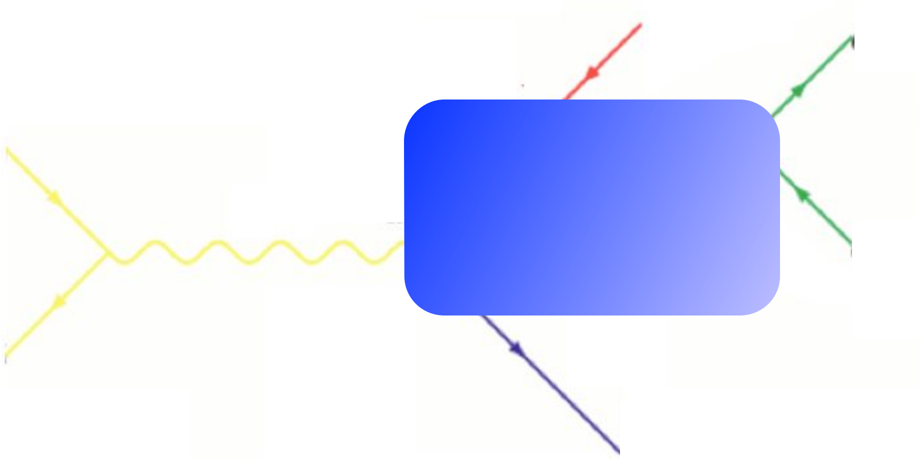


$$(\Delta_q(t_{\rm cut},t_0))^2\Delta_g(t_2,t_1)(\Delta_q({\rm cut},t_2))^2\frac{\alpha_s(t_1)}{2\pi}P_{gq}(z)\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$

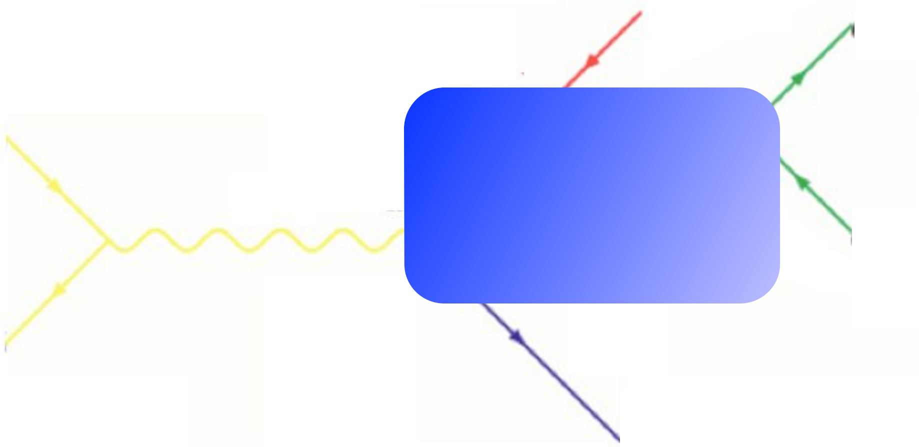
Corresponds to the matrix element
BUT with α_s evaluated at the scale of each splitting

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

Merging ME with PS



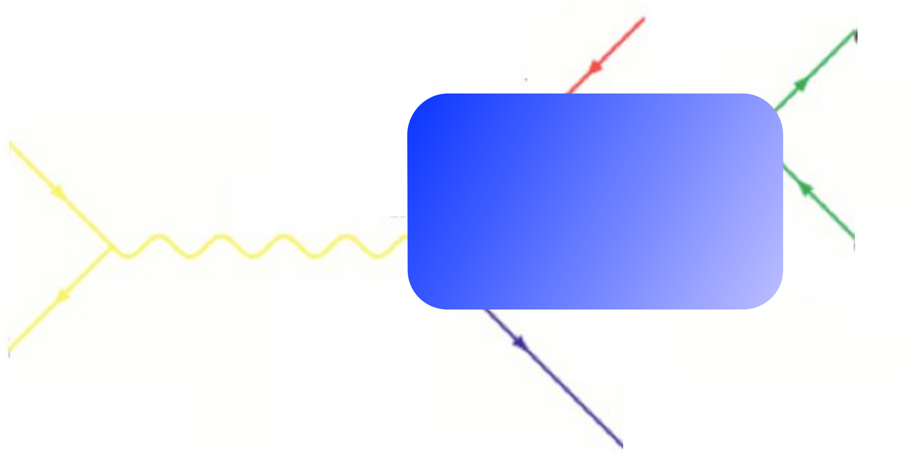
$$|\mathcal{M}|^2(\hat{s}, p_3, p_4, ...)$$



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

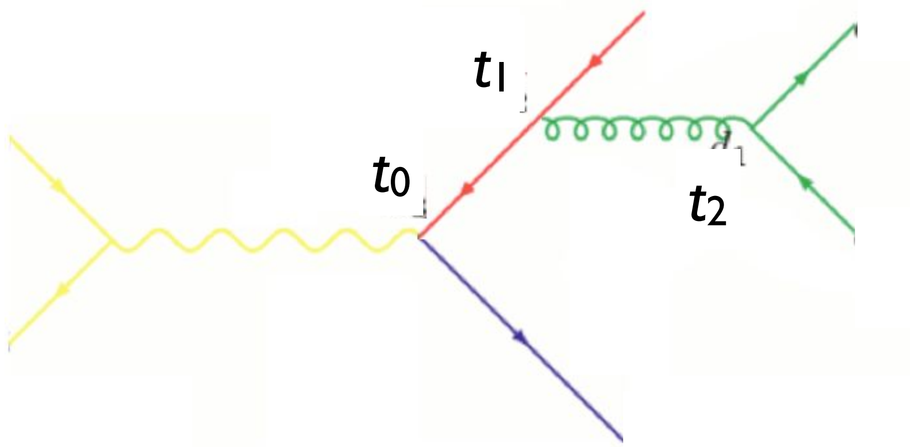
- To get an equivalent treatment of the corresponding matrix element, do as follows:

Merging ME with PS



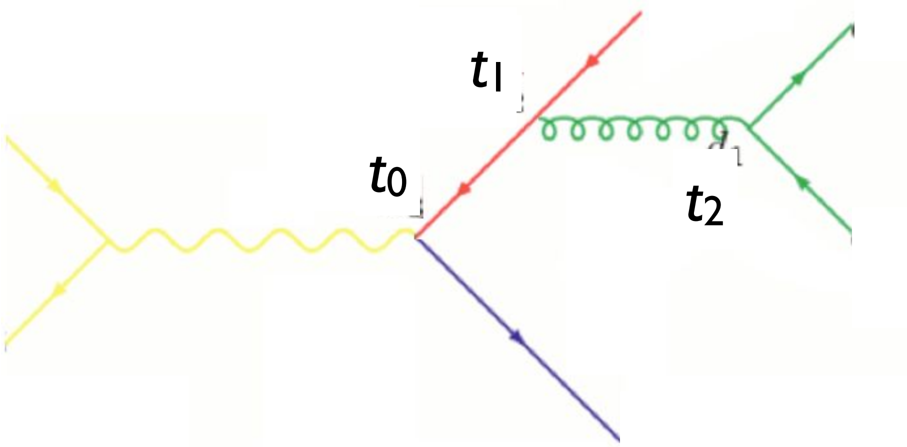
$$|\mathcal{M}|^2(\hat{s}, p_3, p_4, \ldots)$$

- 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"



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

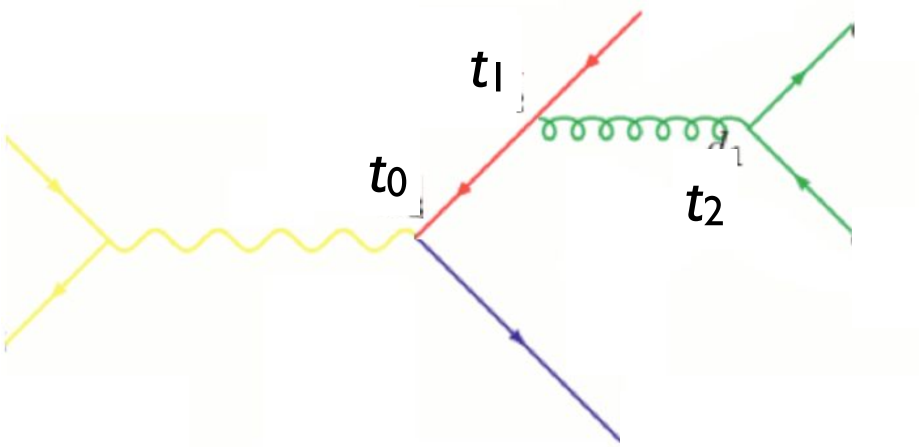
- To get an equivalent treatment of the corresponding matrix element, do as follows:
  1. Cluster the event using some clustering algorithm
     - this gives us a corresponding "parton shower history"



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

- 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

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

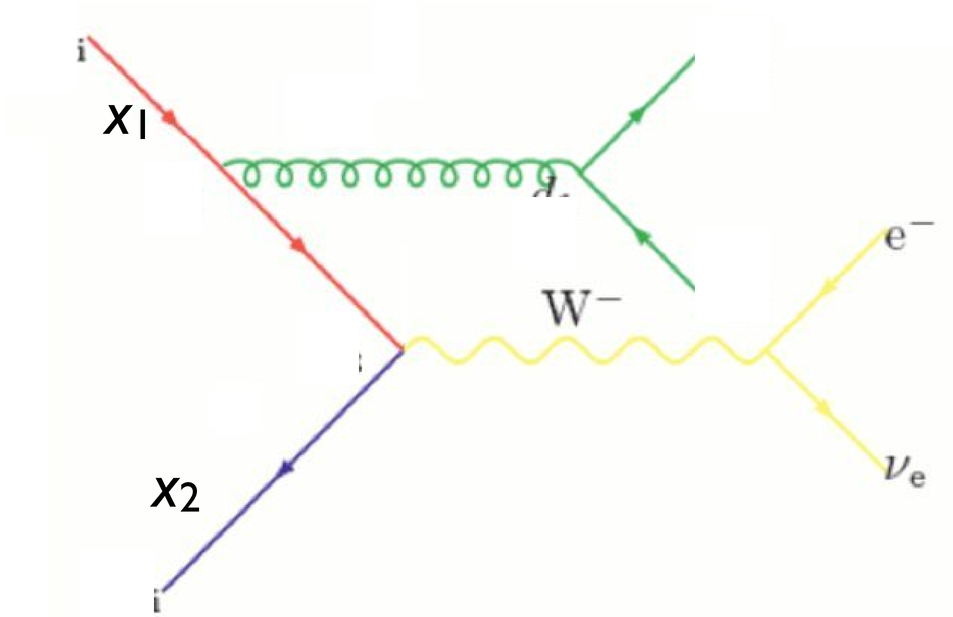


$$|\mathcal{M}|^2(\hat{s}, p_3, p_4, \ldots)$$

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

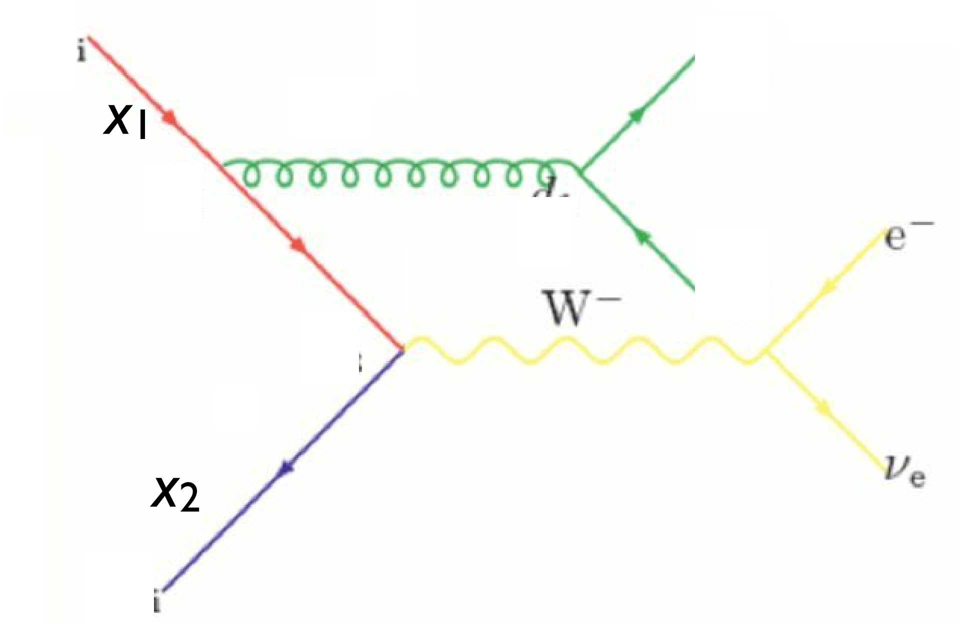
1. Cluster the event using some clustering algorithm
   - this gives us a corresponding "parton shower history"
2. Reweight α_s in each clustering vertex with the clustering scale
$$|\mathcal{M}|^2 \to |\mathcal{M}|^2 \frac{\alpha_s(t_1)}{\alpha_s(t_0)} \frac{\alpha_s(t_2)}{\alpha_s(t_0)}$$
3. Use some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

Matching for initial state radiation



Matching for initial state radiation

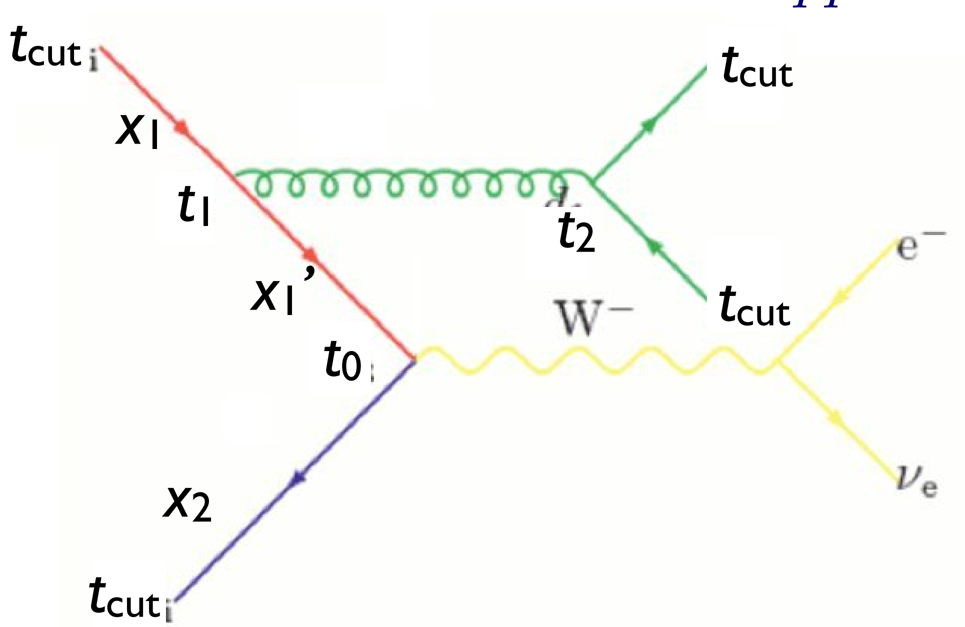
- We are of course not interested in e^+e^- but p-p(bar) - what happens for initial state radiation?



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:

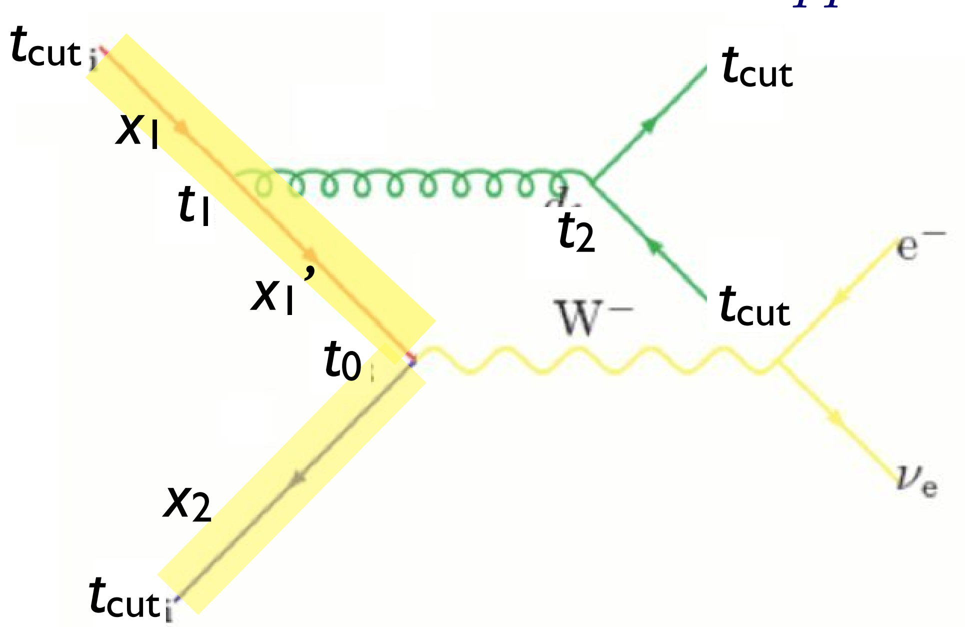
$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} \frac{P_{gq}(z)}{z} \frac{f_q(x_1, t_1)}{f_q(x_1', t_1)} \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$
$$\times \hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s}, \ldots) f_q(x_1', t_0) f_{\bar{q}}(x_2,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



Matching for initial state radiation

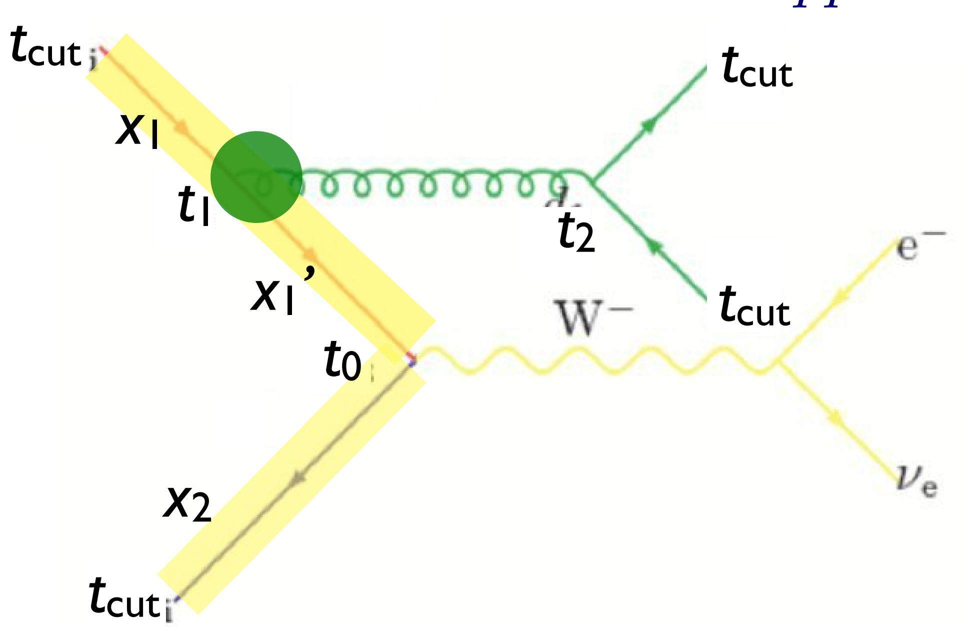
- We are of course not interested in e^+e^- but p-p(bar) - what happens for initial state radiation?
- Let's do the same exercise as before:

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



- 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:

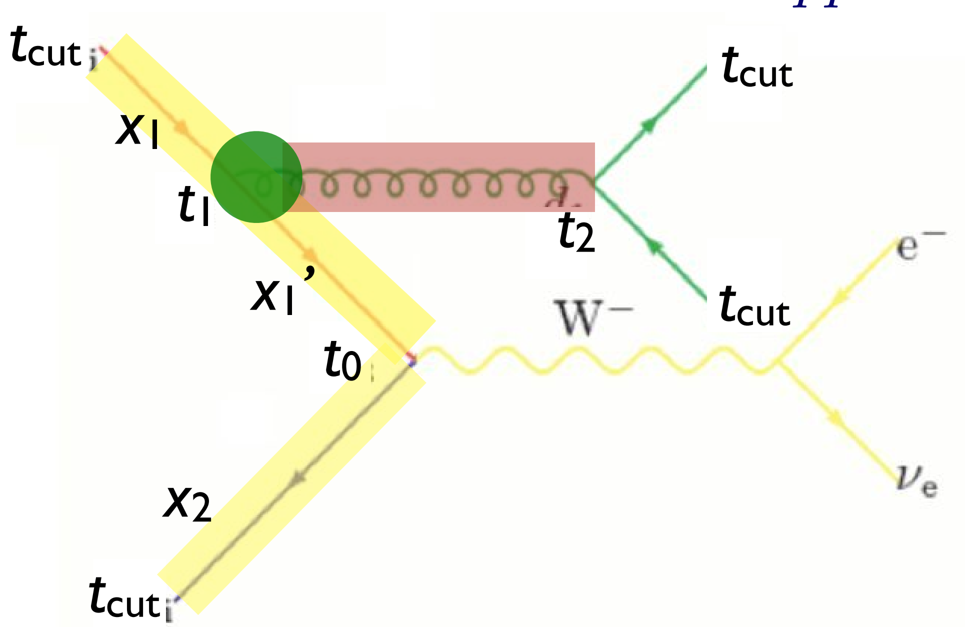
$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut},t_0))^2 \Delta_g(t_2,t_1)(\Delta_q(t_{\rm cut},t_2))^2 \frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1,t_1)}{f_q(x_1',t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$
$$\times \hat\sigma_{q\bar q\to e\nu}(\hat s,\ldots) f_q(x_1',t_0) f_{\bar q}(x_2,t_0)$$



- 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:

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

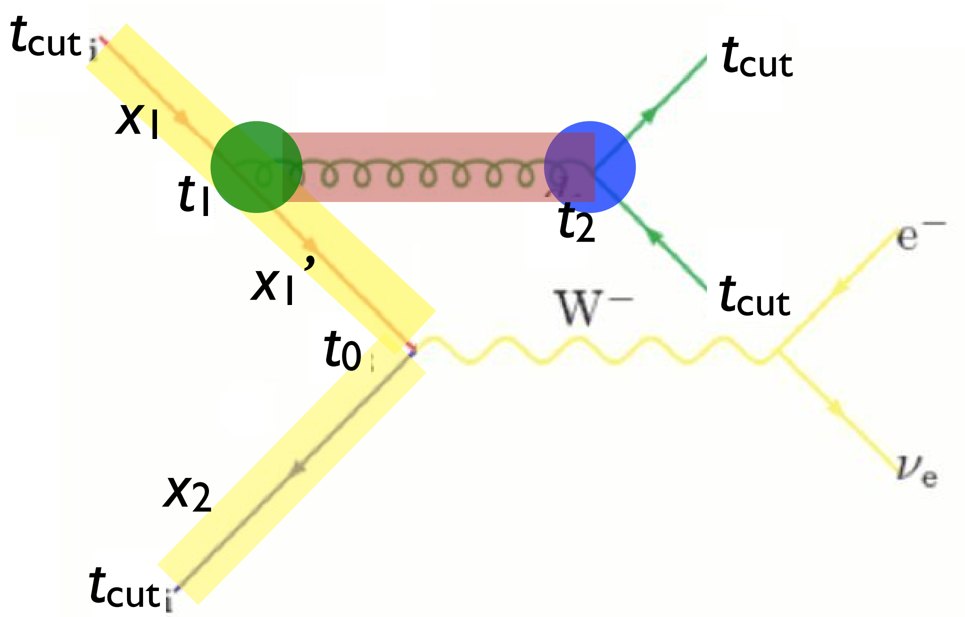


Matching for initial state radiation

- 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:

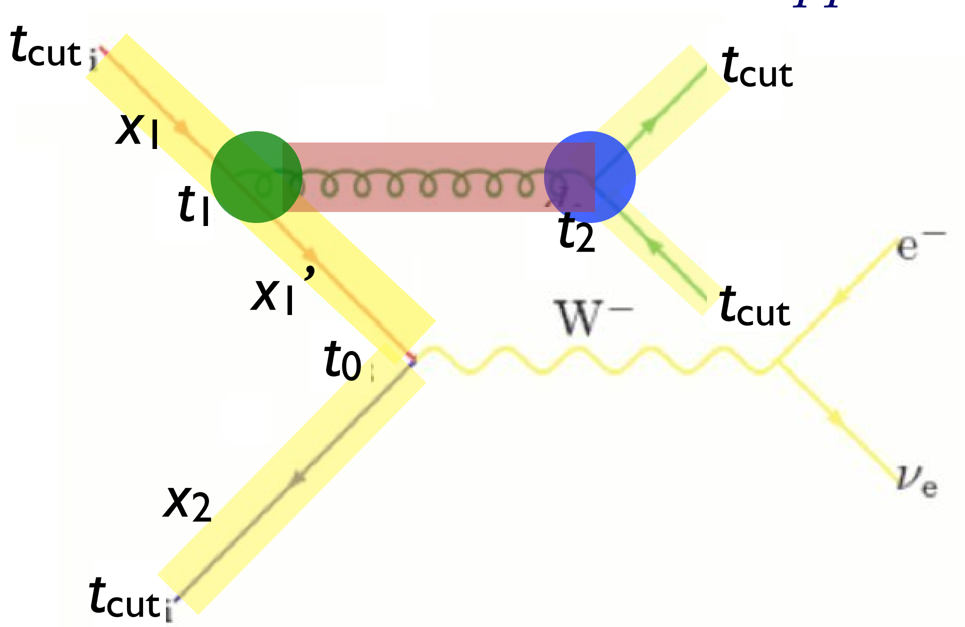
$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1, t_1)}{f_q(x_1', t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$

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



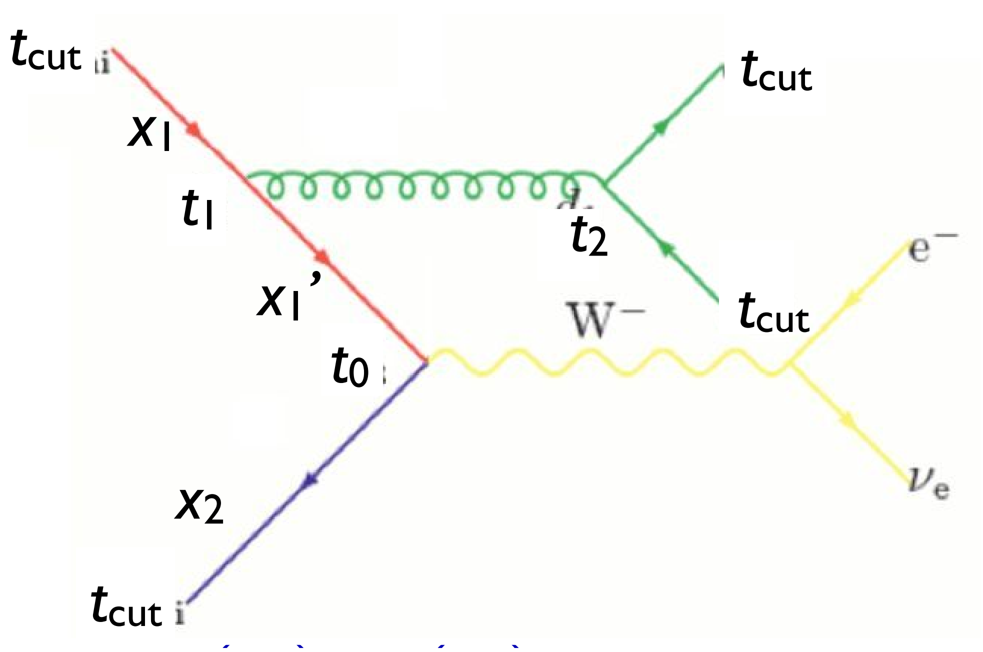
- We are of course not interested in e^+e^- but p-p(bar) - what happens for initial state radiation?
- Let's do the same exercise as before:

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



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

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

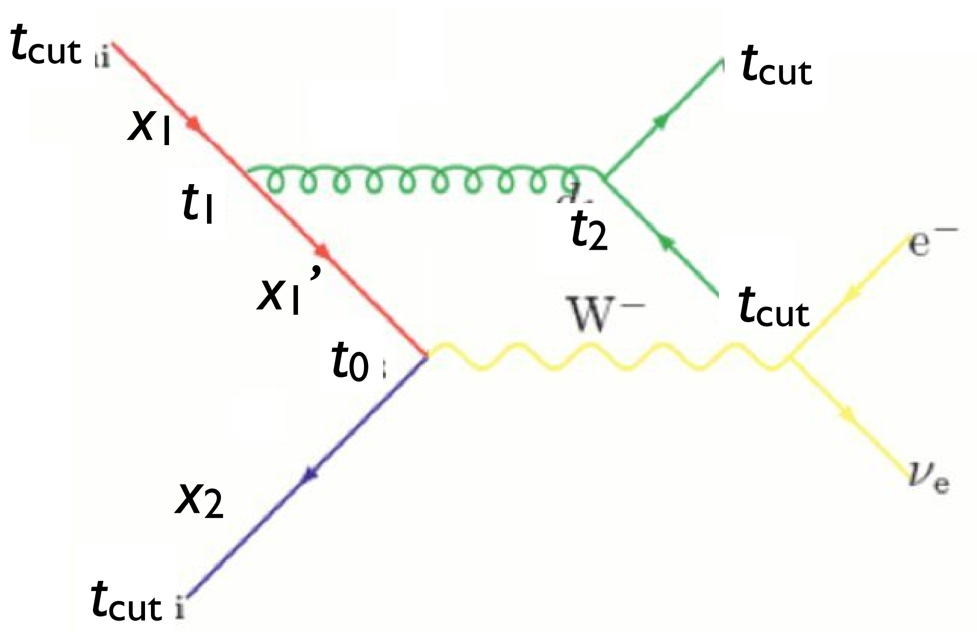


Matching for initial state radiation

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

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

ME with α_s evaluated at the scale of each splitting



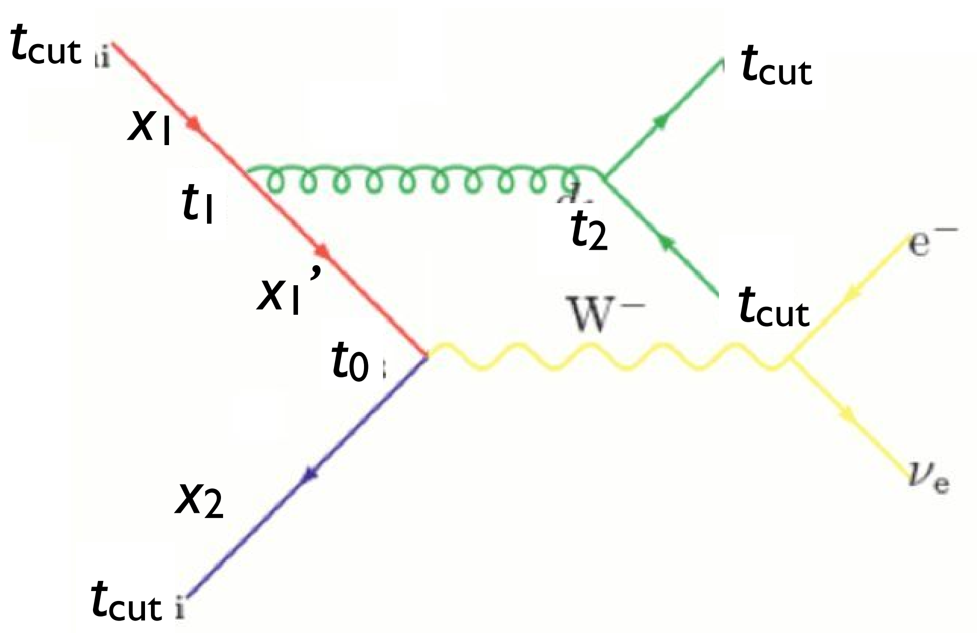
Matching for initial state radiation

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

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

ME with α_s evaluated at the scale of each splitting

PDF reweighting



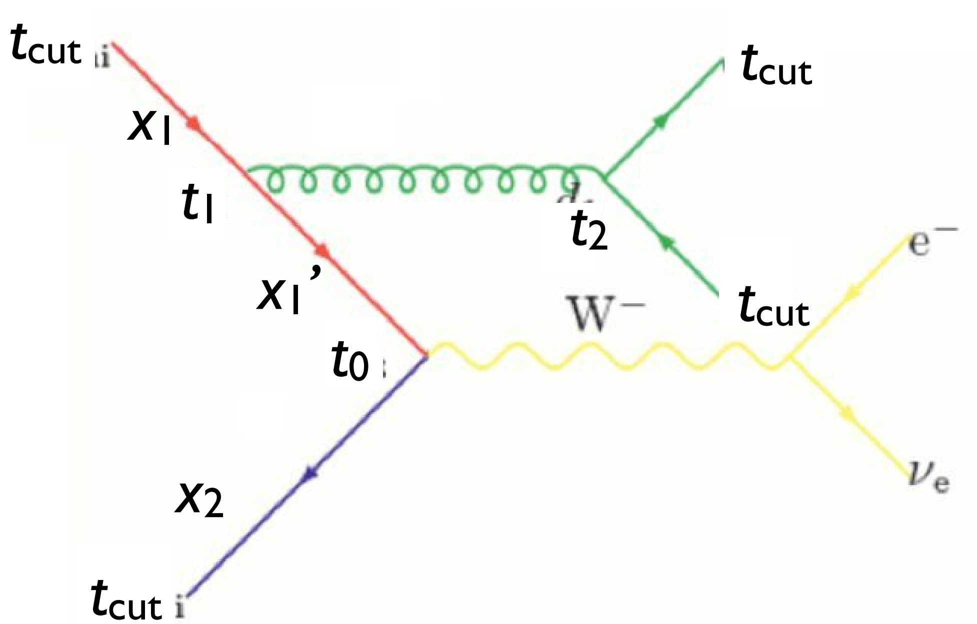
$$(\Delta_{Iq}(t_{\rm cut},t_0))^2\Delta_g(t_2,t_1)(\Delta_q(t_{\rm cut},t_2))^2\frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1,t_1)}{f_q(x_1',t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$

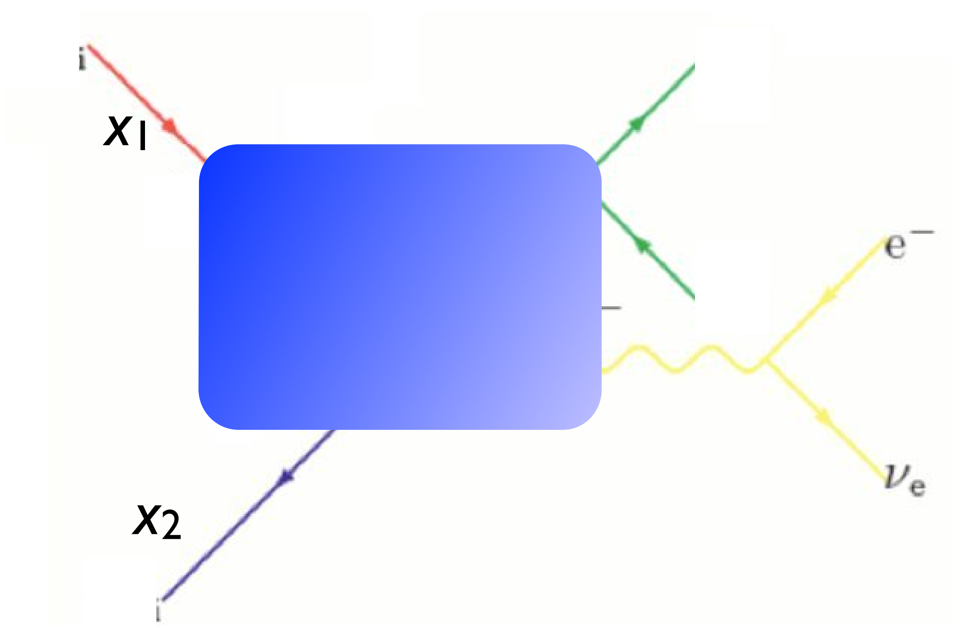
$$\times\hat\sigma_{q\bar q\to e\nu}(\hat s,\ldots)f_q(x_1',t_0)f_{\bar q}(x_2,t_0)$$

ME with α_s evaluated at the scale of each splitting

PDF reweighting

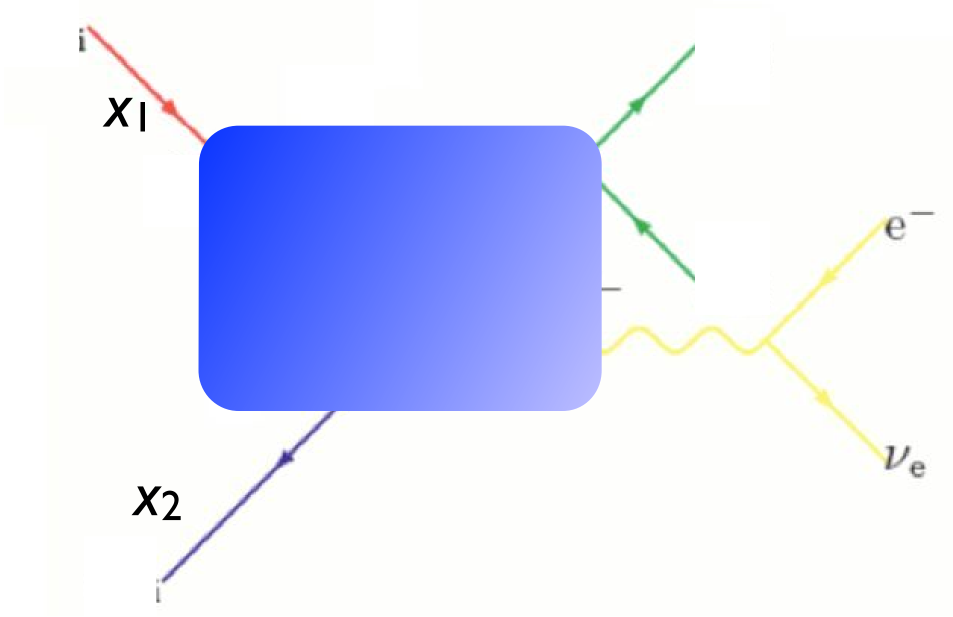
Sudakov suppression due to non-branching above scale $t_{\rm cut}$




x1
x2
e−
νe

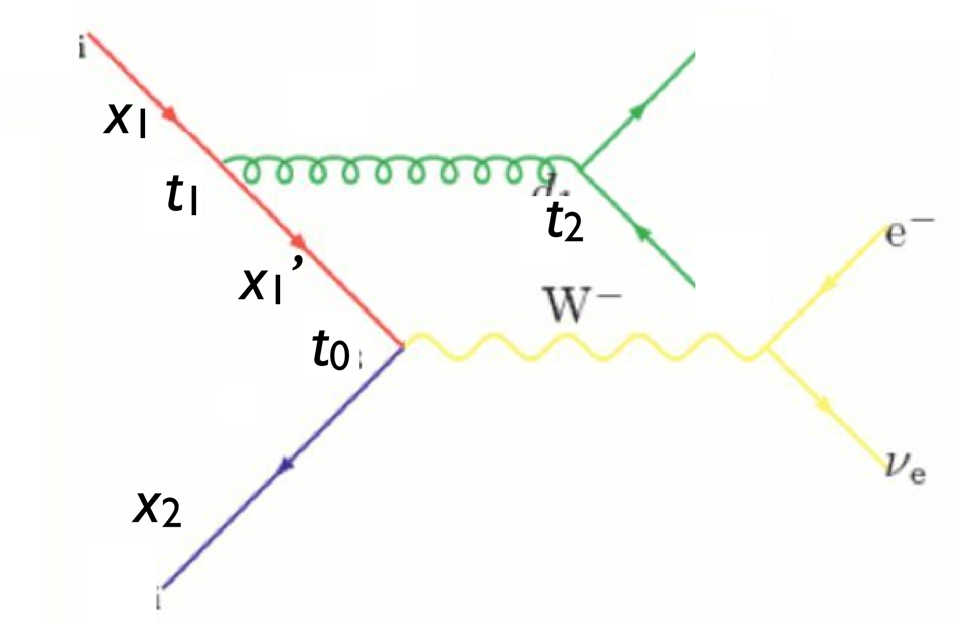
Matching for initial state radiation

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



Matching for initial state radiation

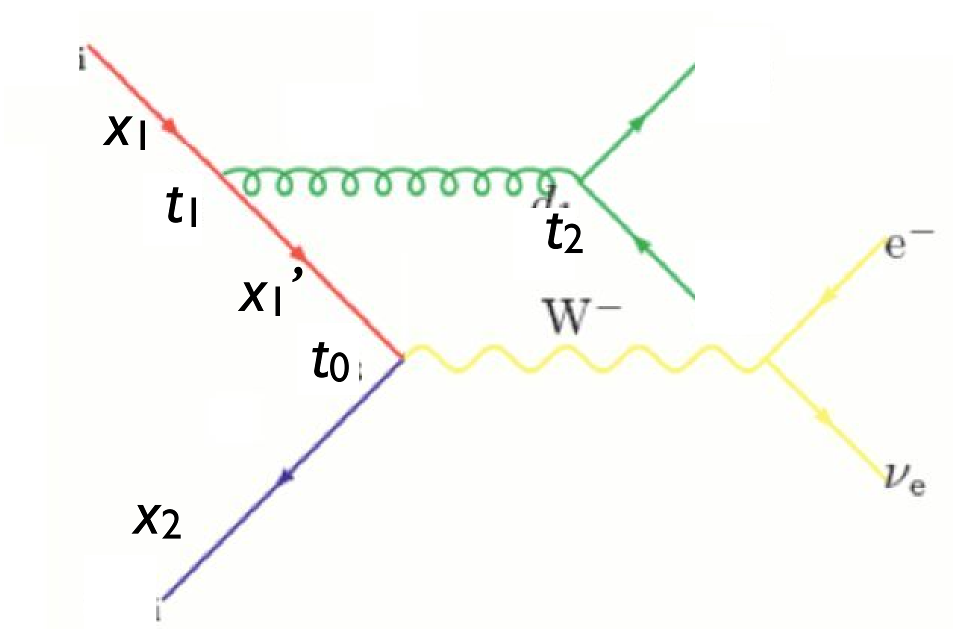
- Again, use a clustering scheme to get a parton shower history



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

$$|\mathcal{M}|^2 \to |\mathcal{M}|^2 \frac{\alpha_s(t_1)}{\alpha_s(t_0)} \frac{\alpha_s(t_2)}{\alpha_s(t_0)} \frac{f_q(x'_1, t_0)}{f_q(x'_1, t_1)}$$



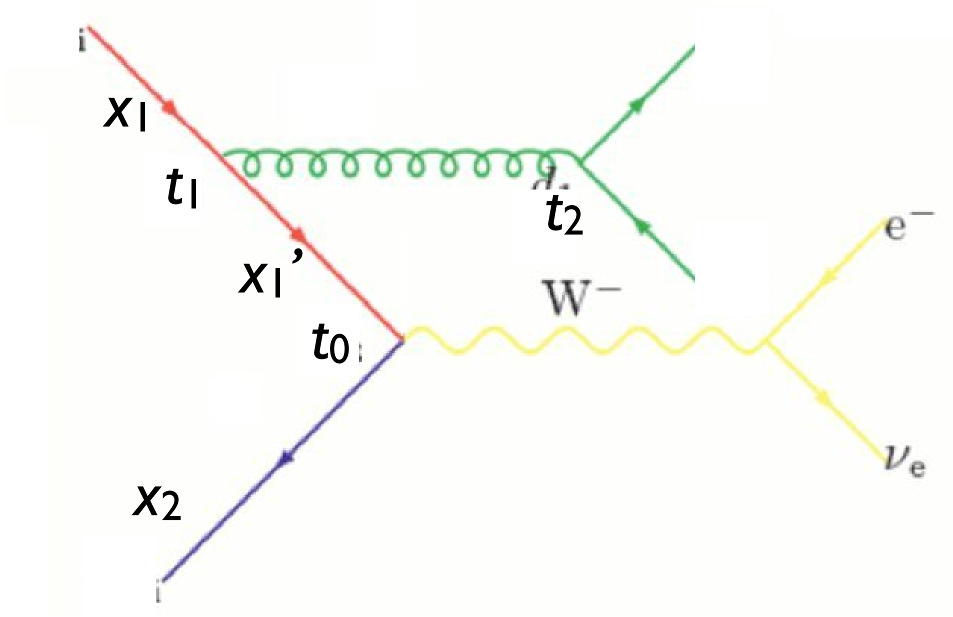
Matching for initial state radiation

- 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:

$$\mathcal{P}_{\text{event}} = \hat{\sigma}(x_1, x_2, p_3, p_4, \dots) f_q(x_1, t_1) f_{\bar{q}}(x_2, t_0)$$



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

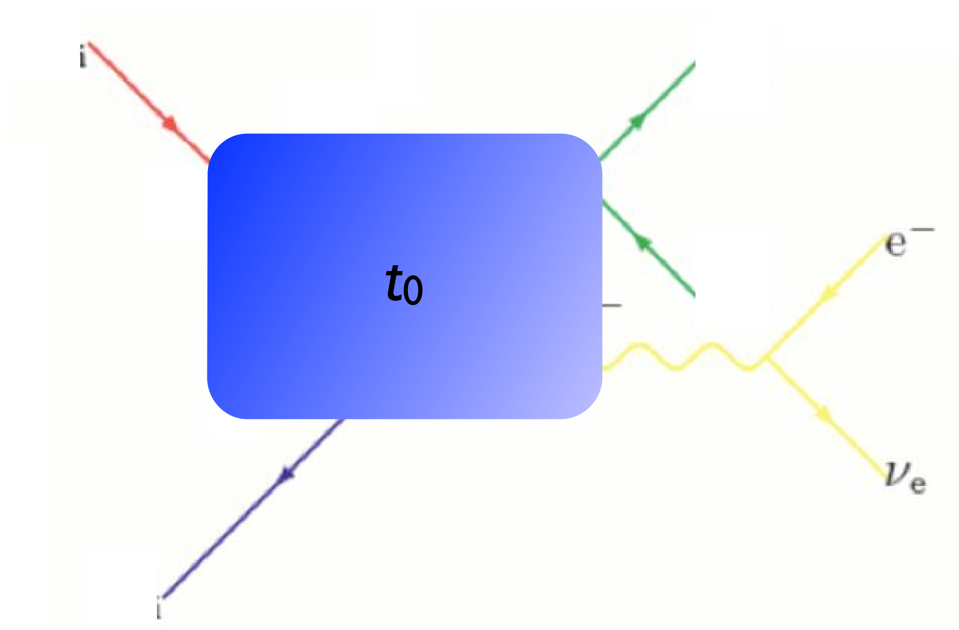
[Catani, Krauss, Kuhn, Webber 2001]
[Krauss 2002]

- 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

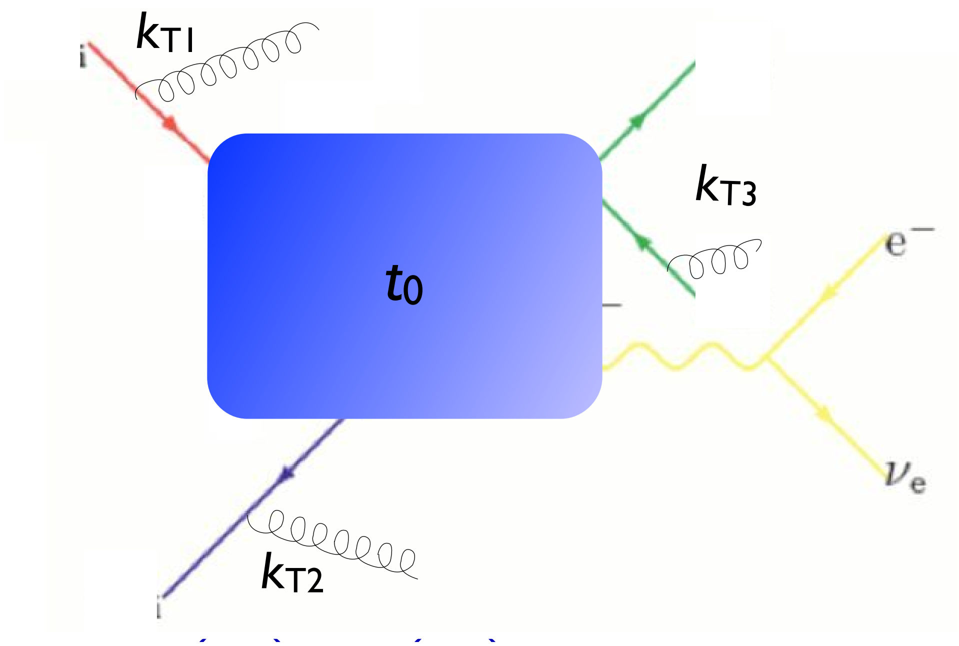
[Catani, Krauss, Kuhn, Webber 2001]
[Krauss 2002]

- 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_{cut}.



CKKW matching

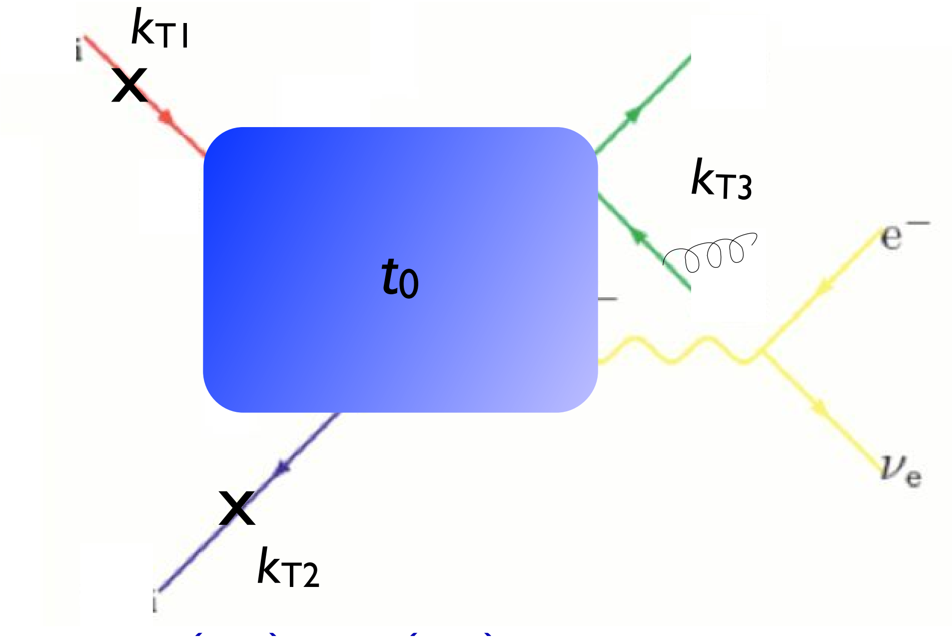
[Catani, Krauss, Kuhn, Webber 2001]
[Krauss 2002]

- 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 $t_{\rm cut}$.



[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.

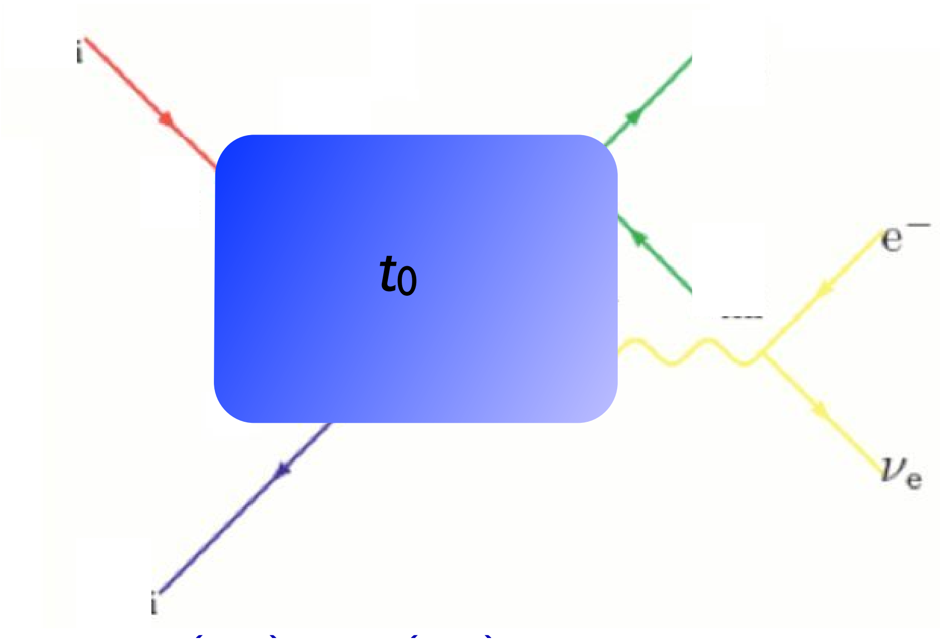
- Perform "truncated showering": Run the parton shower starting at t_0, but forbid any showers above the cutoff scale $t_{\rm cut}$.

✓ Best theoretical treatment of matrix element

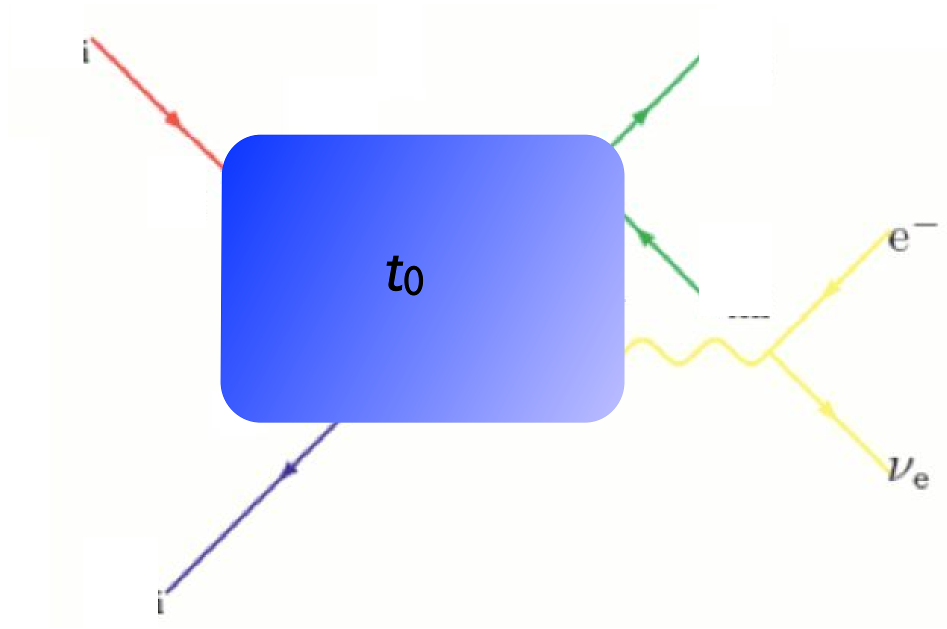
- Requires dedicated PS implementation
- Mismatch between analytical Sudakov and (non-NLL) shower
- Implemented in Sherpa (v. 1.1)

CKKW-L matching

[Lönnblad 2002]
[Hoeche et al. 2009]



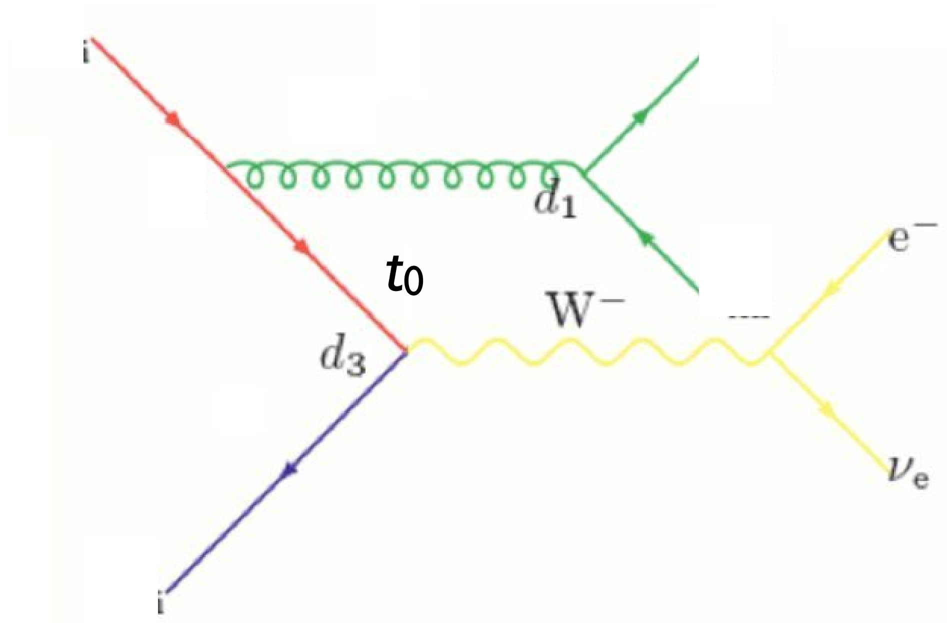
[Lönnblad 2002]
[Hoeche et al. 2009]



- Cluster back to "parton shower history"

CKKW-L matching

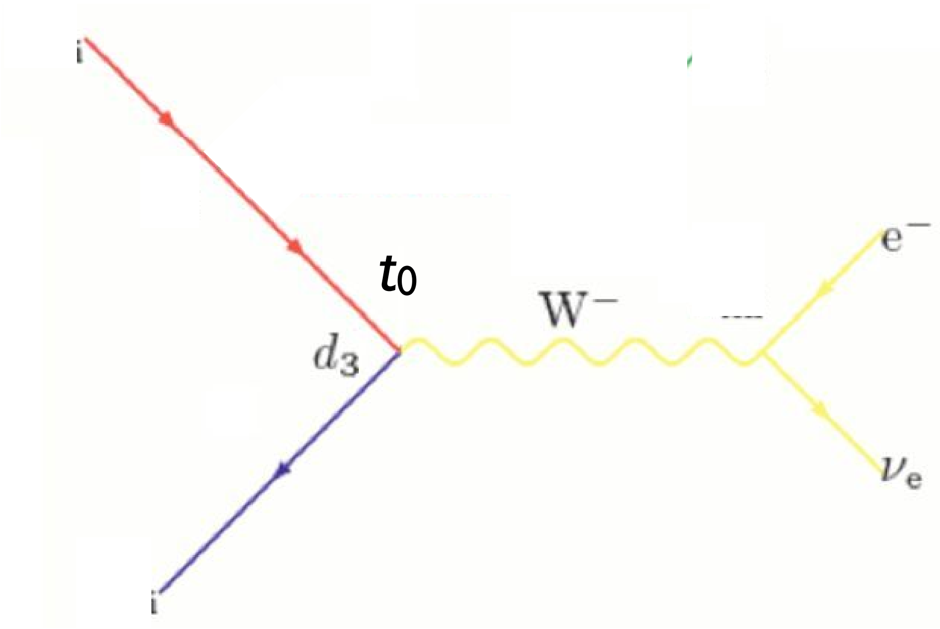
[Lönnblad 2002]
[Hoeche et al. 2009]



- Cluster back to "parton shower history"

CKKW-L matching

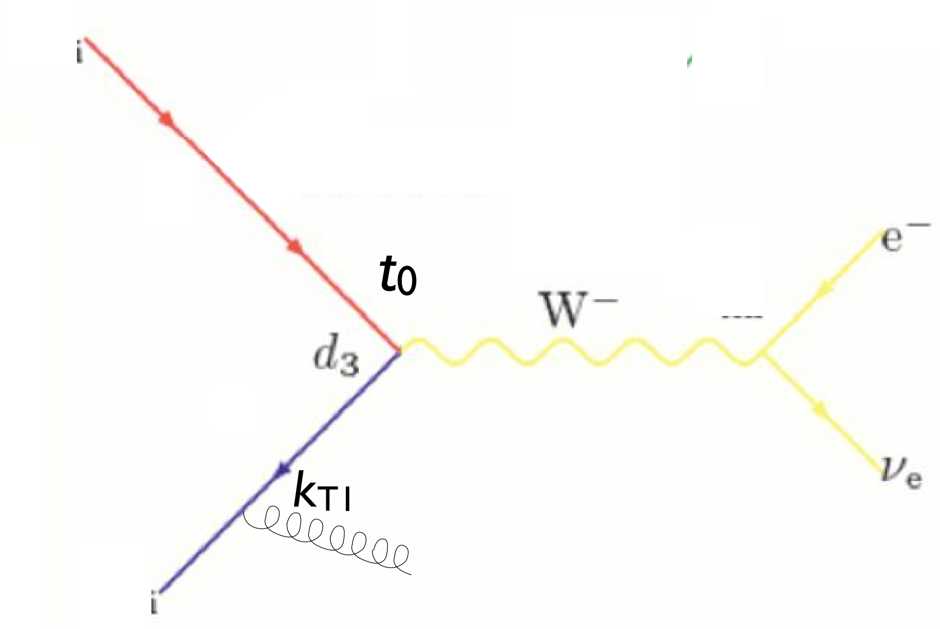
[Lönnblad 2002]
[Hoeche et al. 2009]



- 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

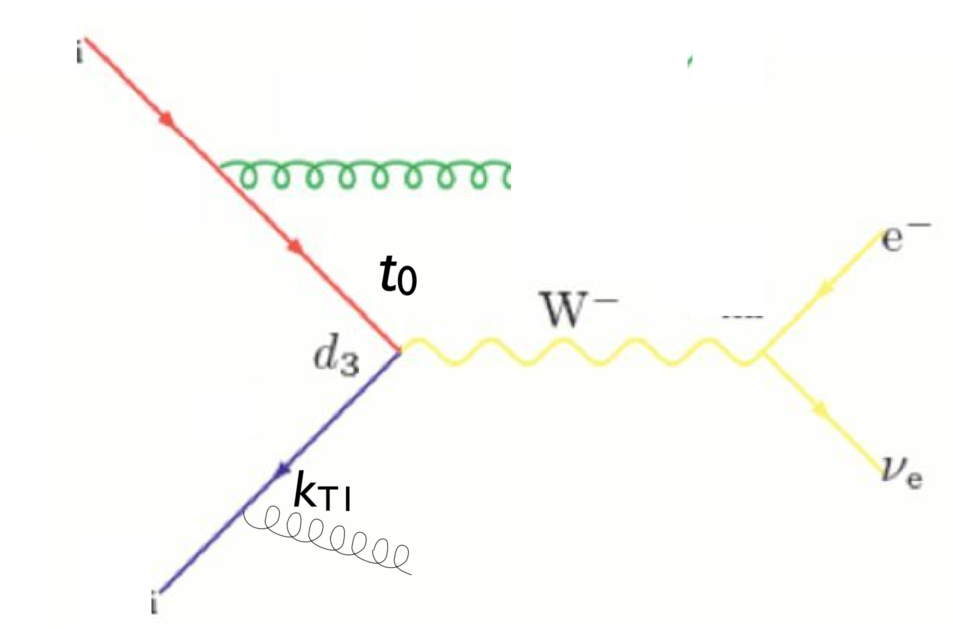
[Lönnblad 2002]
[Hoeche et al. 2009]



- 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

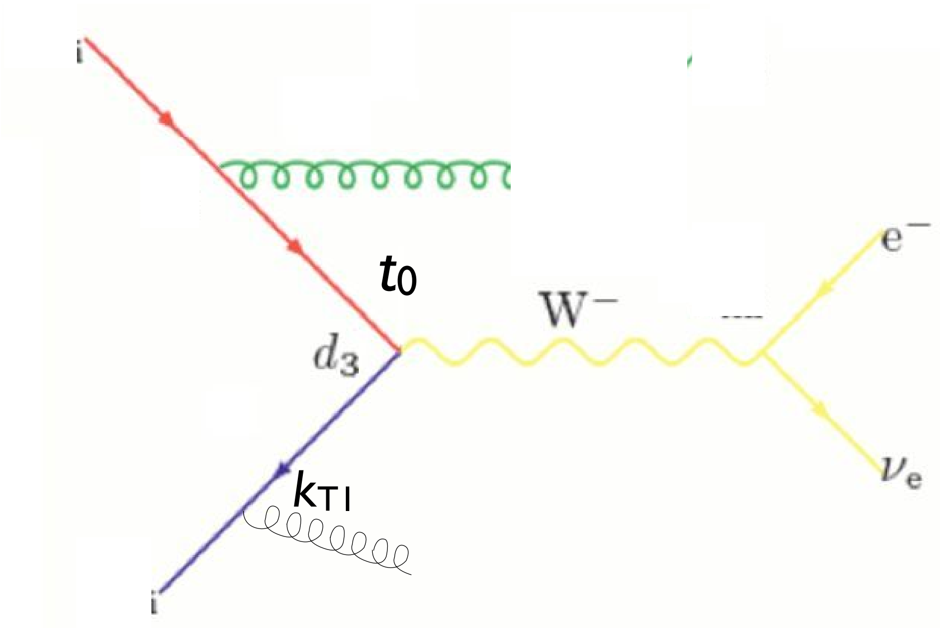
[Lönnblad 2002]
[Hoeche et al. 2009]



- 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

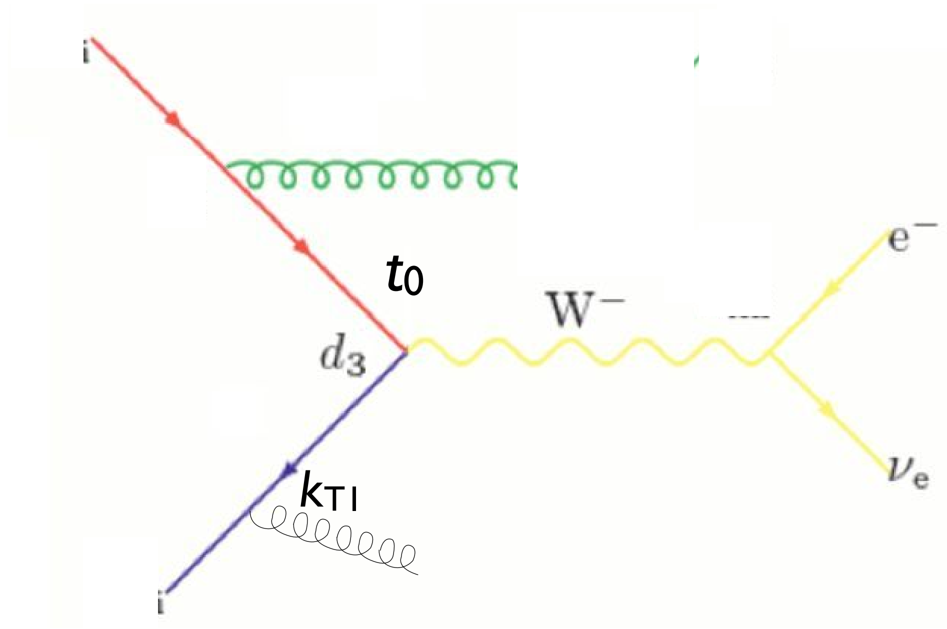
[Lönnblad 2002]
[Hoeche et al. 2009]



- 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
- 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

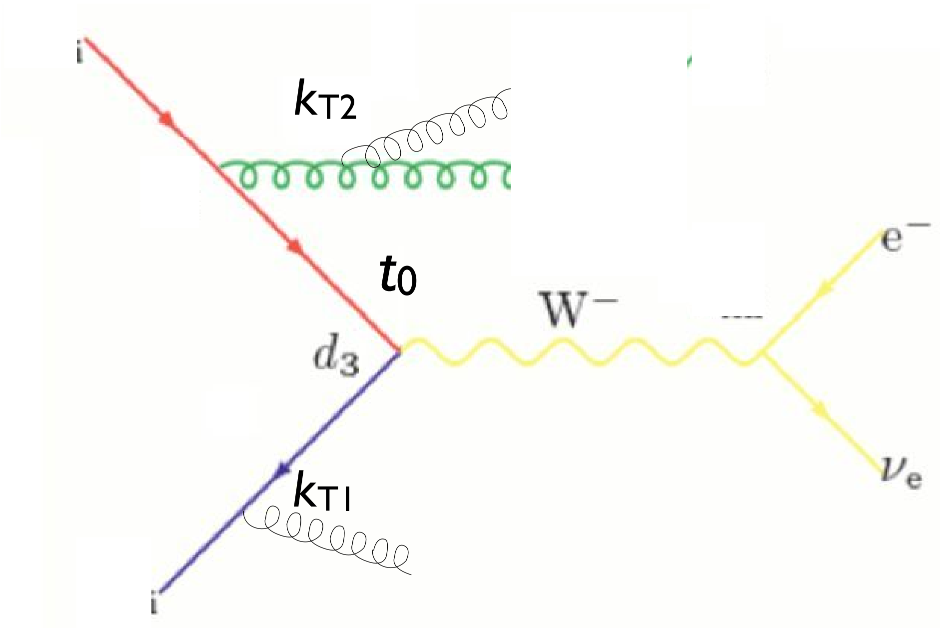
[Lönnblad 2002]
[Hoeche et al. 2009]



- 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

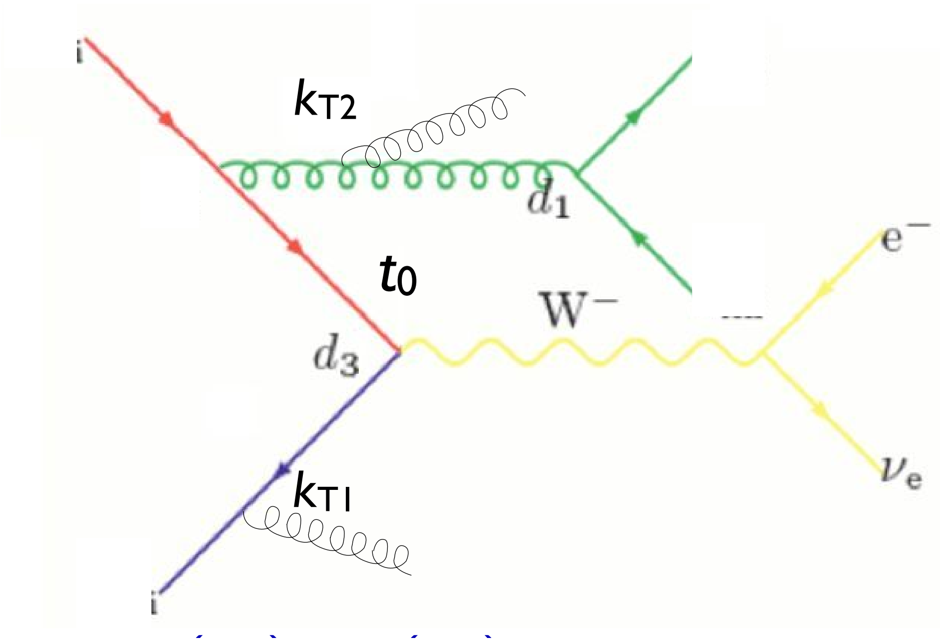
[Lönnblad 2002]
[Hoeche et al. 2009]



- 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
- 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

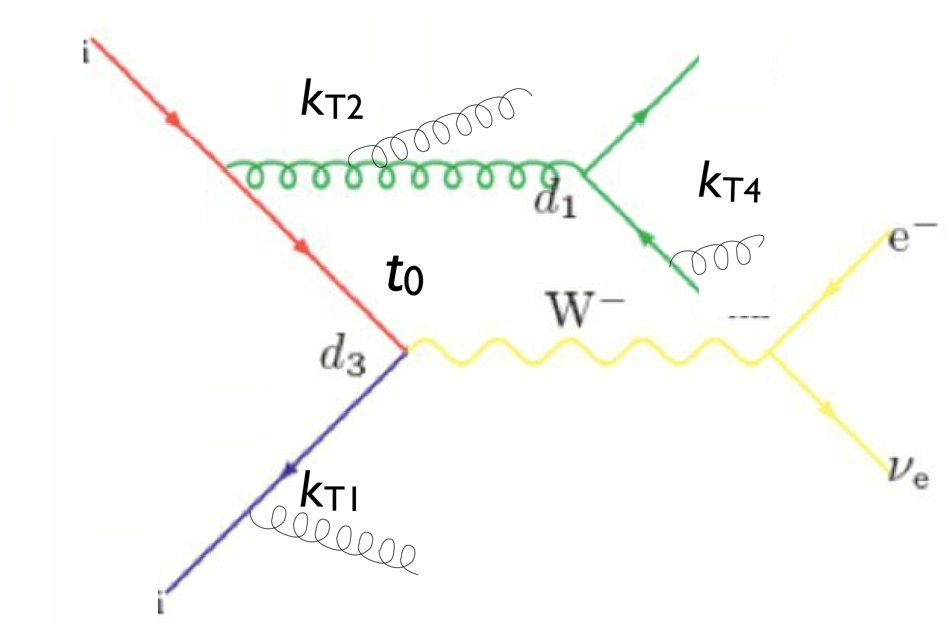
[Lönnblad 2002]
[Hoeche et al. 2009]



- 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
- 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

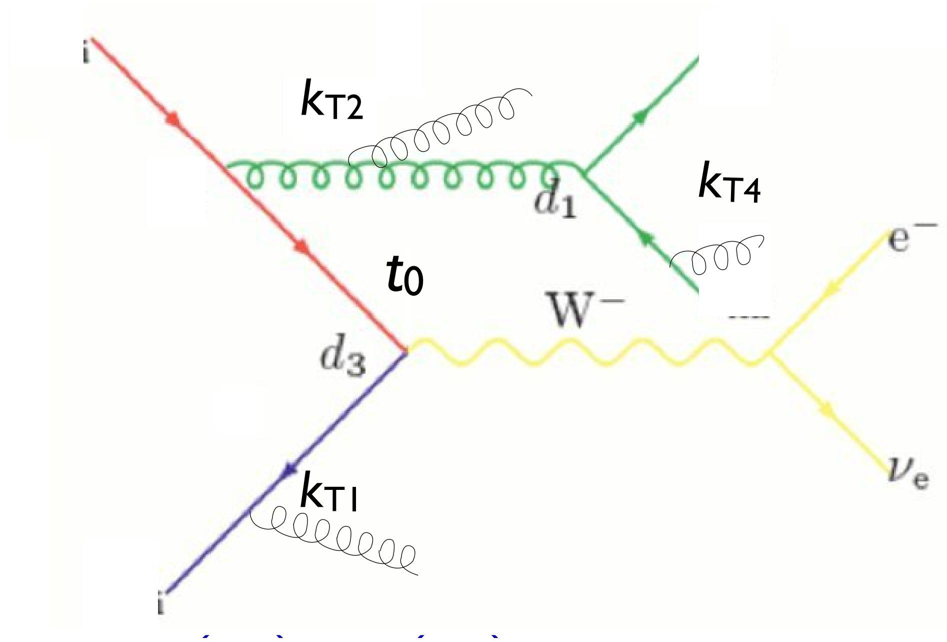
[Lönnblad 2002]
[Hoeche et al. 2009]



- 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
- 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

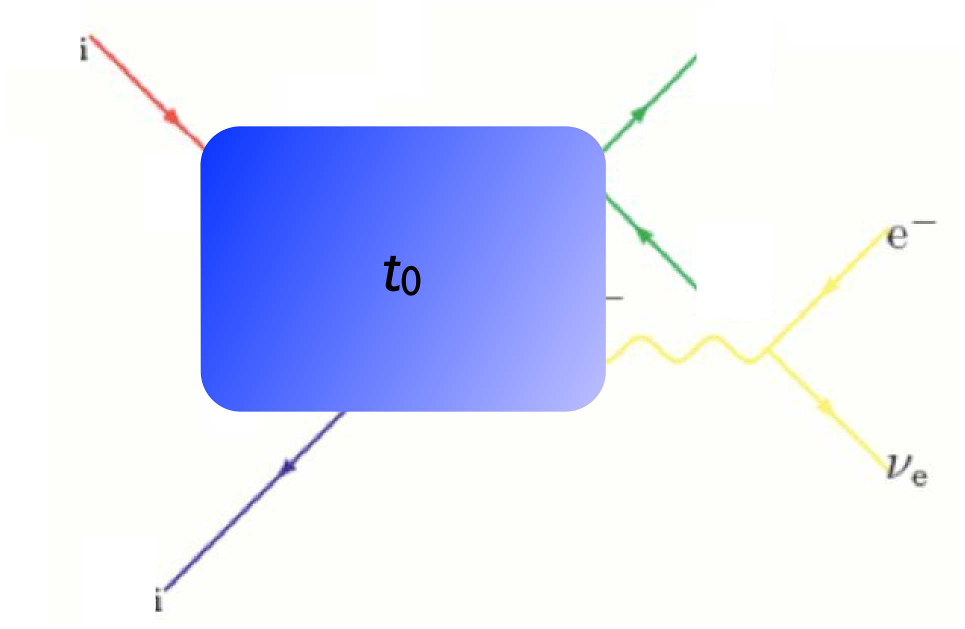
[Lönnblad 2002]
[Hoeche et al. 2009]



✓ Automatic agreement between Sudakov and shower

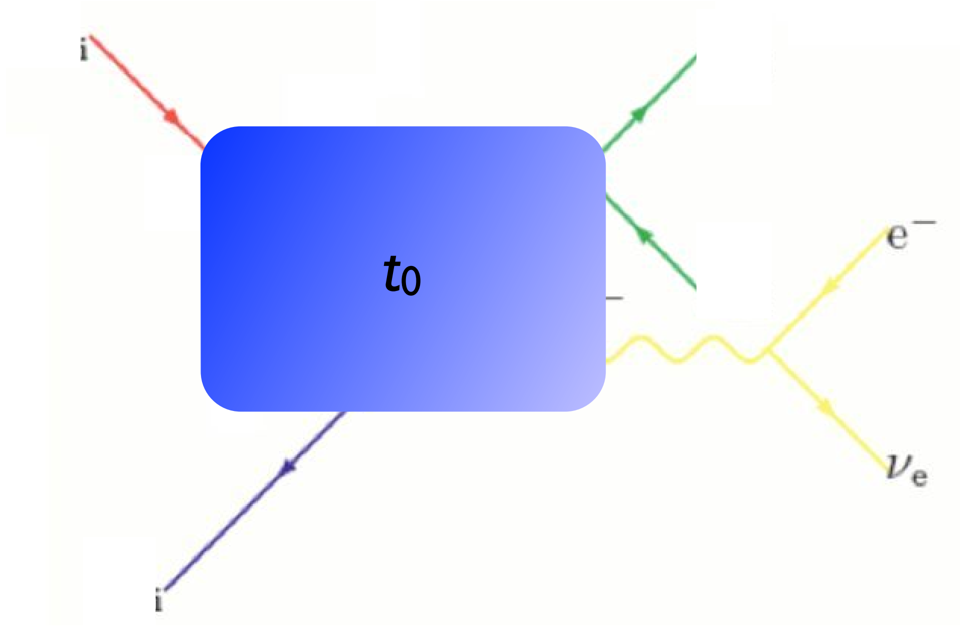
- Requires dedicated PS implementation
  ➡ Need multiple implementations to compare between showers
- Implemented in Ariadne, Sherpa (v. 1.2), and Pythia 8

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



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

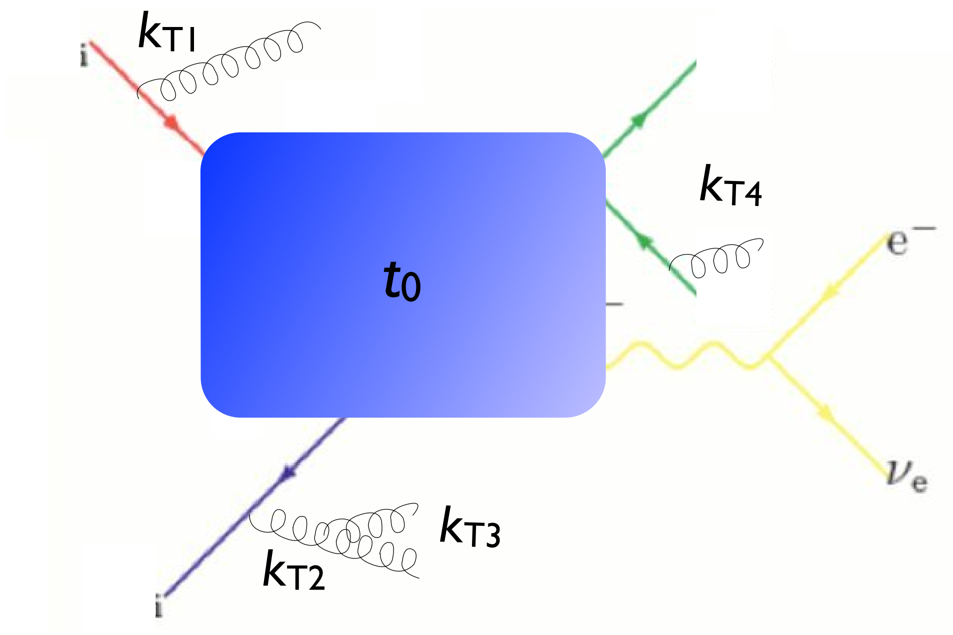
- The simplest way to do the Sudakov suppression is to run the shower on the event, starting from t_0!



MLM matching

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

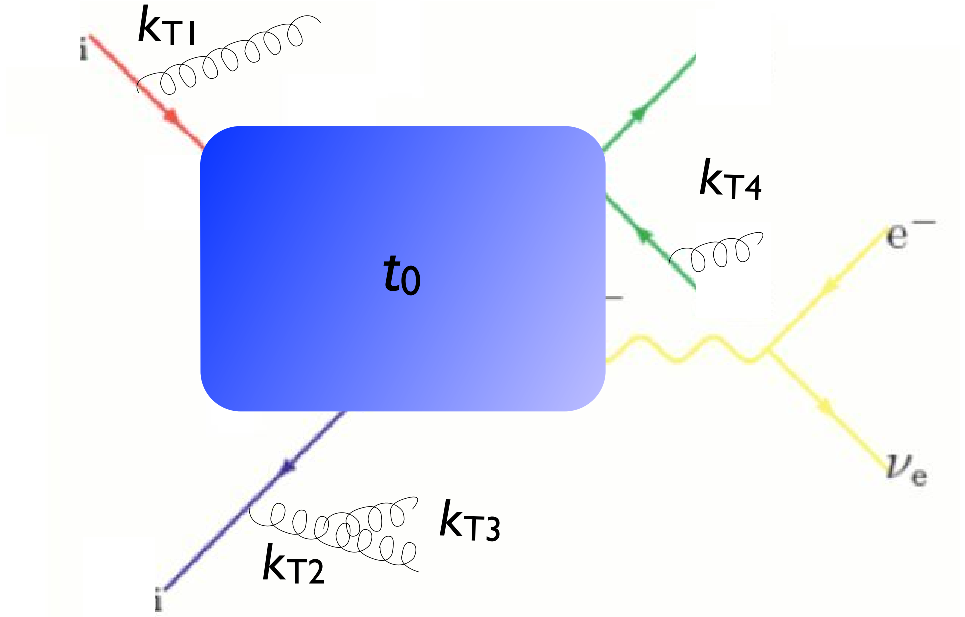
- The simplest way to do the Sudakov suppression is to run the shower on the event, starting from t_0!



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!

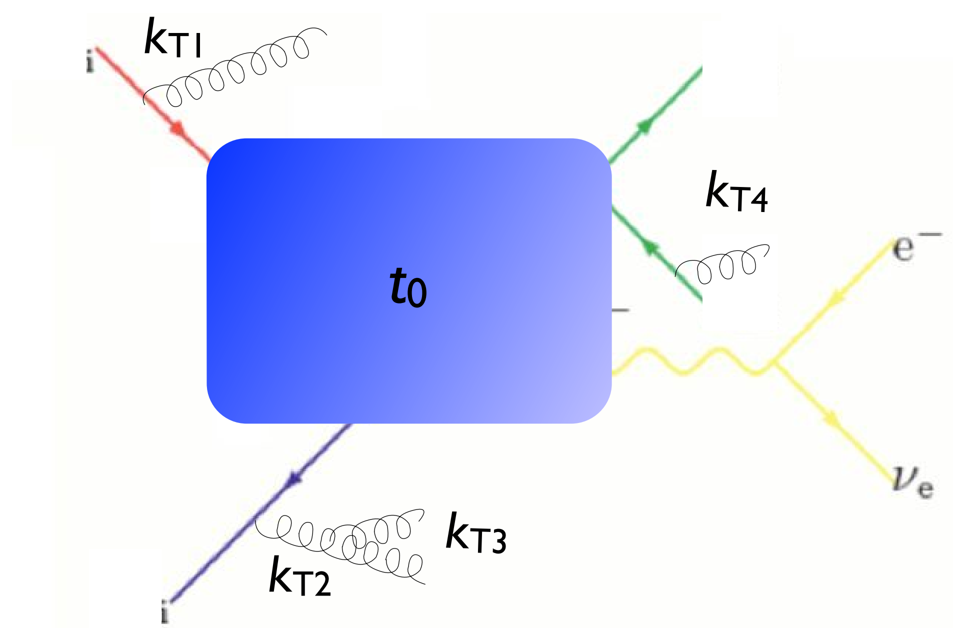


- Perform jet clustering after PS - if hardest jet $k_{T1} > t_{cut}$ or there are jets not matched to partons, reject the event

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

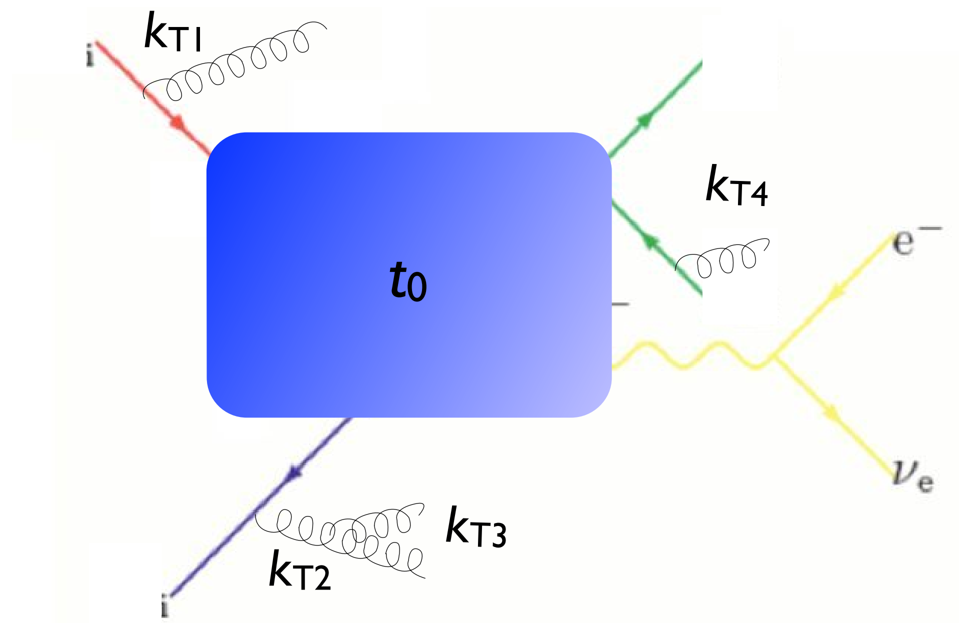
$$(\Delta_{Iq}(t_{\rm cut},t_0))^2(\Delta_q(t_{\rm cut},t_0))^2$$

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

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!



✓ Simplest available scheme

✓ Allows matching with any shower, without modification

➡ Sudakov suppression not exact, minor mismatch with shower

- Implemented in AlpGen, HELAC, MadGraph+Pythia 6

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

matching schemes

- We have a number of choices to make in the above procedure. The most important are:
  1. The clustering scheme used to determine the parton shower history of the ME event
  2. What to use for the scale Q^2 (factorization scale)
  3. How to divide the phase space between parton showers and matrix elements

Cluster schemes

1. The clustering scheme used inside MadGraph and Sherpa 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:

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

and

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

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

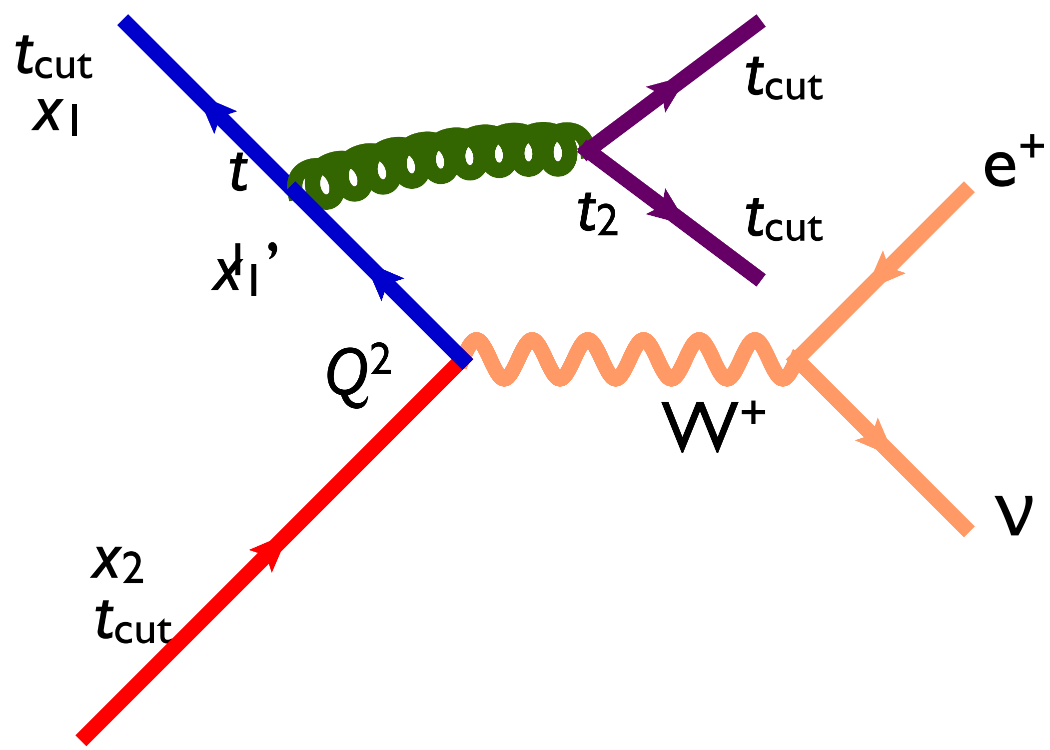
Find the smallest k_{Tij} (or $k_{Ti\mathrm{beam}}$), combine partons *i* and *j* (or *i* and the beam), and continue until you reach a 2 → 2 (or 2 → 1) scattering.

2. In AlpGen a more naive cone algorithm is used.

- Cannot use the standard k_T clustering:
  - MadGraph and Sherpa only allow clustering according to valid diagrams in the process. This means that, e.g., two quarks or quark-antiquark of different flavor are never clustered, and the clustering always gives a physically allowed parton shower history.
  - If there is an on-shell propagator in the diagram (e.g. a top quark), only clustering according to diagrams with this propagator is allowed.

Hard scale

2. The clustering provides a convenient choice for factorization scale Q^2:



Cluster back to the 2 → 2 (here qq → W⁻g) system, and use the W boson transverse mass in that system.

Phase-space division

3. How to divide the phase space between PS and ME:
   This is where the schemes really differ:

   AlpGen: MLM Cone
   MadGraph: MLM Cone, k_T or shower-k_T
   Sherpa: CKKW

Phase-space division

3. How to divide the phase space between PS and ME:
   This is where the schemes really differ:

   AlpGen: MLM Cone
   MadGraph: MLM Cone, k_T or shower-k_T
   Sherpa: CKKW

a. **Cone jet MLM scheme** (better suited for angular ordered showers, i.e. herwig, but works for all showers):
   - 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)

Phase-space division

3. How to divide the phase space between PS and ME:
   This is where the schemes really differ:

   AlpGen: MLM Cone
   MadGraph: MLM Cone, k_T or shower-k_T
   Sherpa: CKKW

b. **k_T-jet MLM scheme** (better suited for k_T ordered showers, i.e. pythia, but works for all showers):
   - 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)

Phase-space division

3. How to divide the phase space between PS and ME:
   This is where the schemes really differ:

   AlpGen: MLM Cone
   MadGraph: MLM Cone, k_T or shower-k_T
   Sherpa: CKKW

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 $k_T^{PS} < k_T^{match}$

Phase-space division

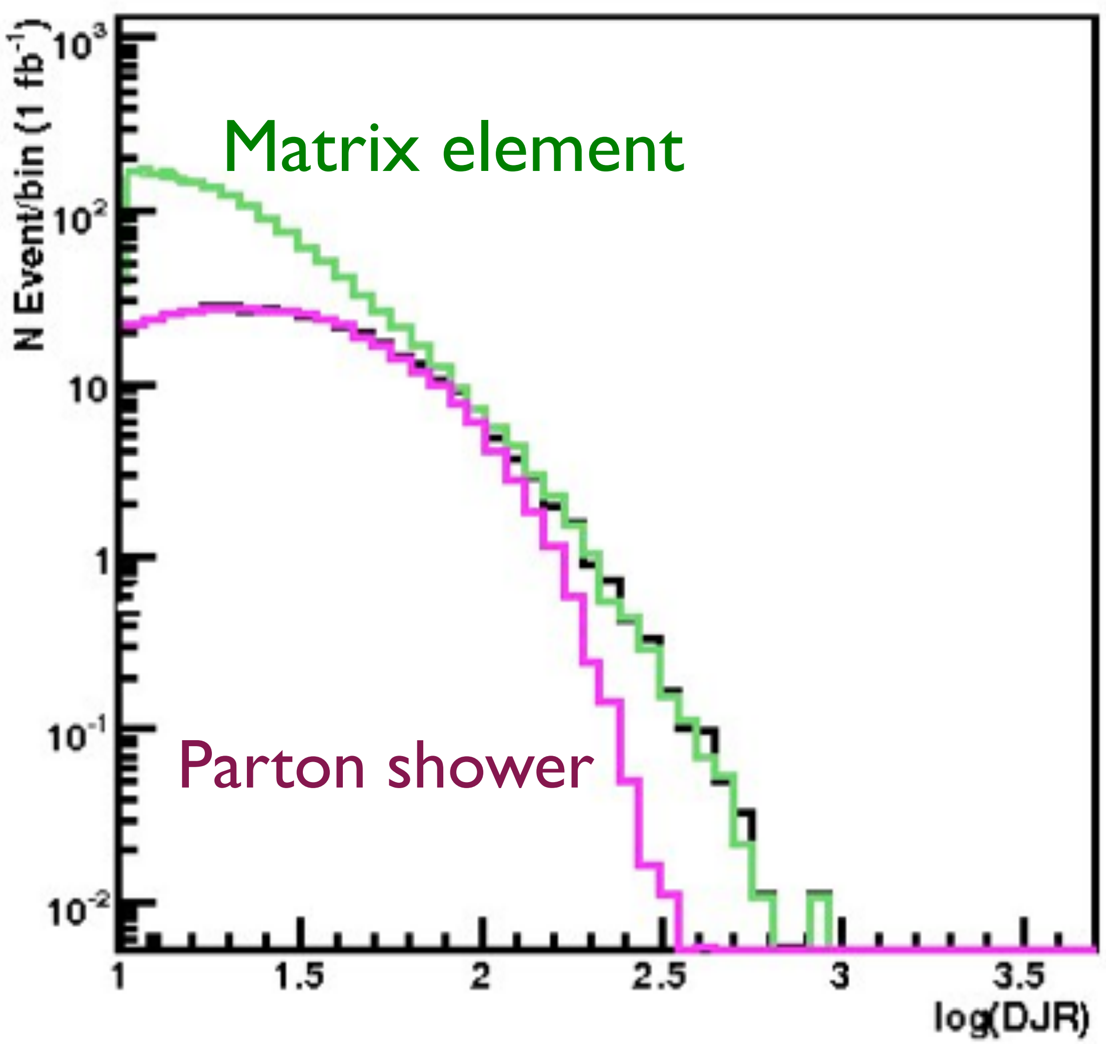
3. How to divide the phase space between PS and ME:
   This is where the schemes really differ:

   AlpGen: MLM Cone
   MadGraph: MLM Cone, k_T or shower-k_T
   Sherpa: CKKW

d. **CKKW Scheme** (Need special veto'ed shower):
   - Use cut in the Durham k_T in ME (k_T^{match})
   - Because the Durham k_T is not the same as the evolution parameter of the shower, we might miss contributions, therefore
   - Start the shower at the original scale, and after each emission, check the value of t_i:
   - if $t_i > k_T^{match}$ veto that emission, i.e. continue the shower as if that emission never happened

Back to the "matching goal"

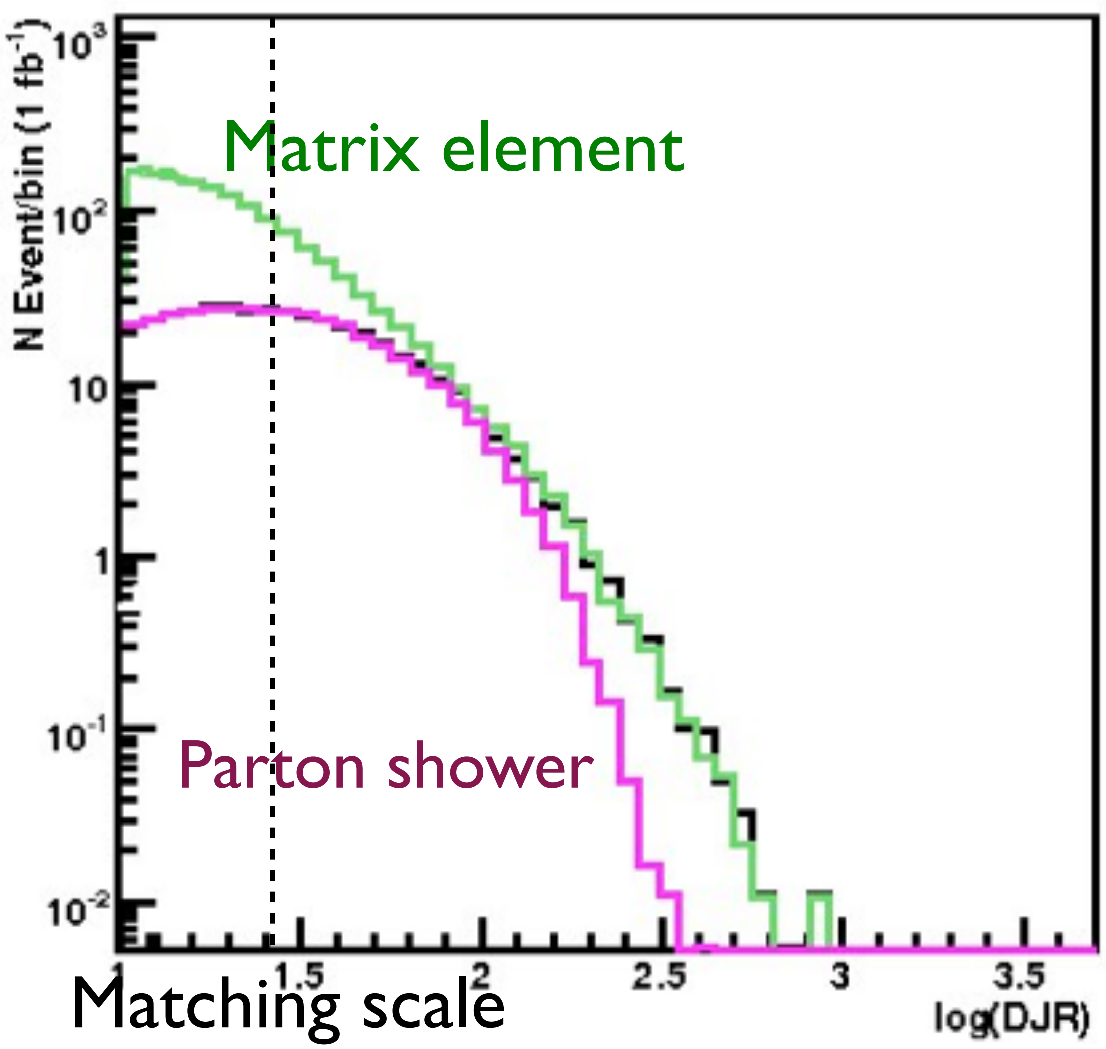
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



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

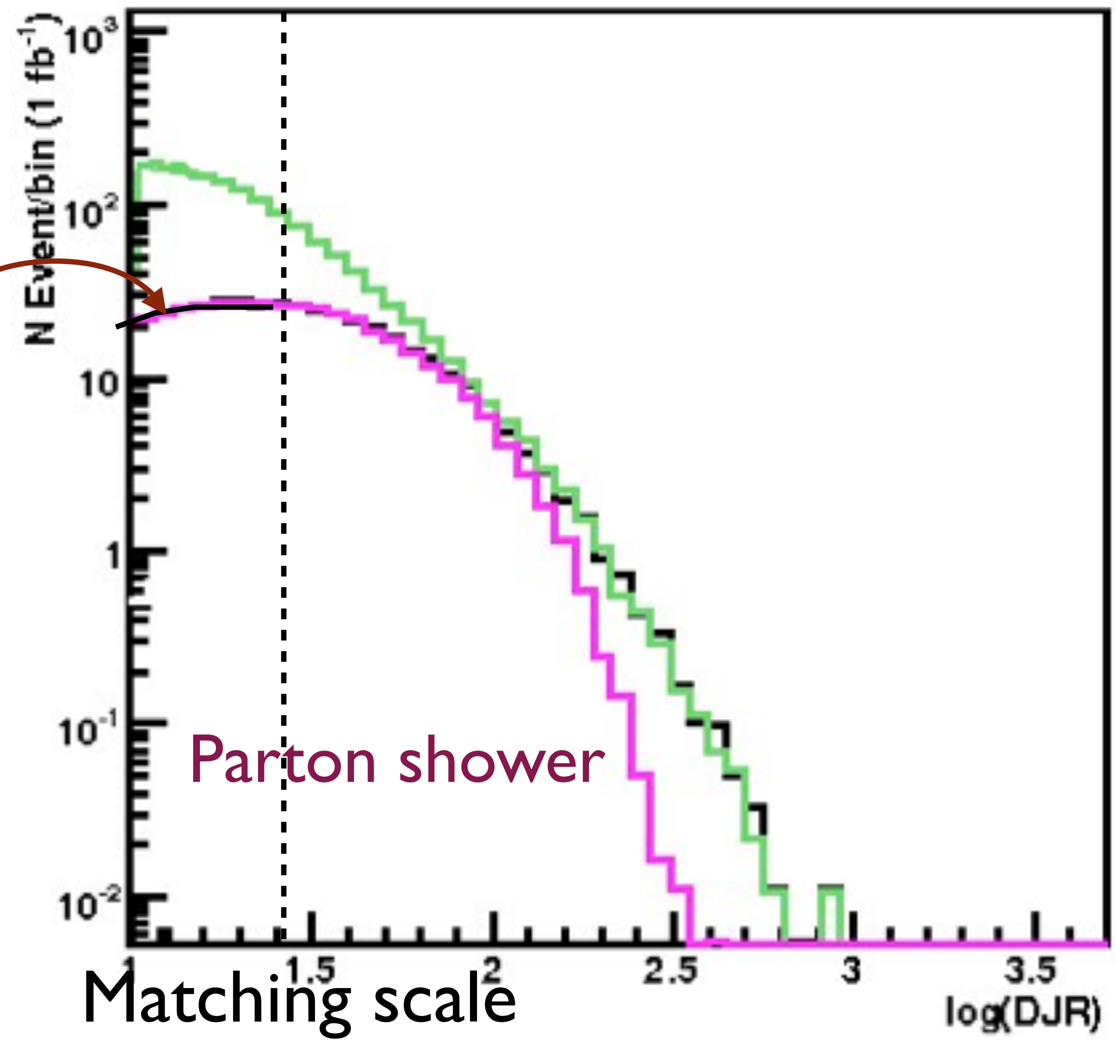


2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Back to the “matching goal”

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

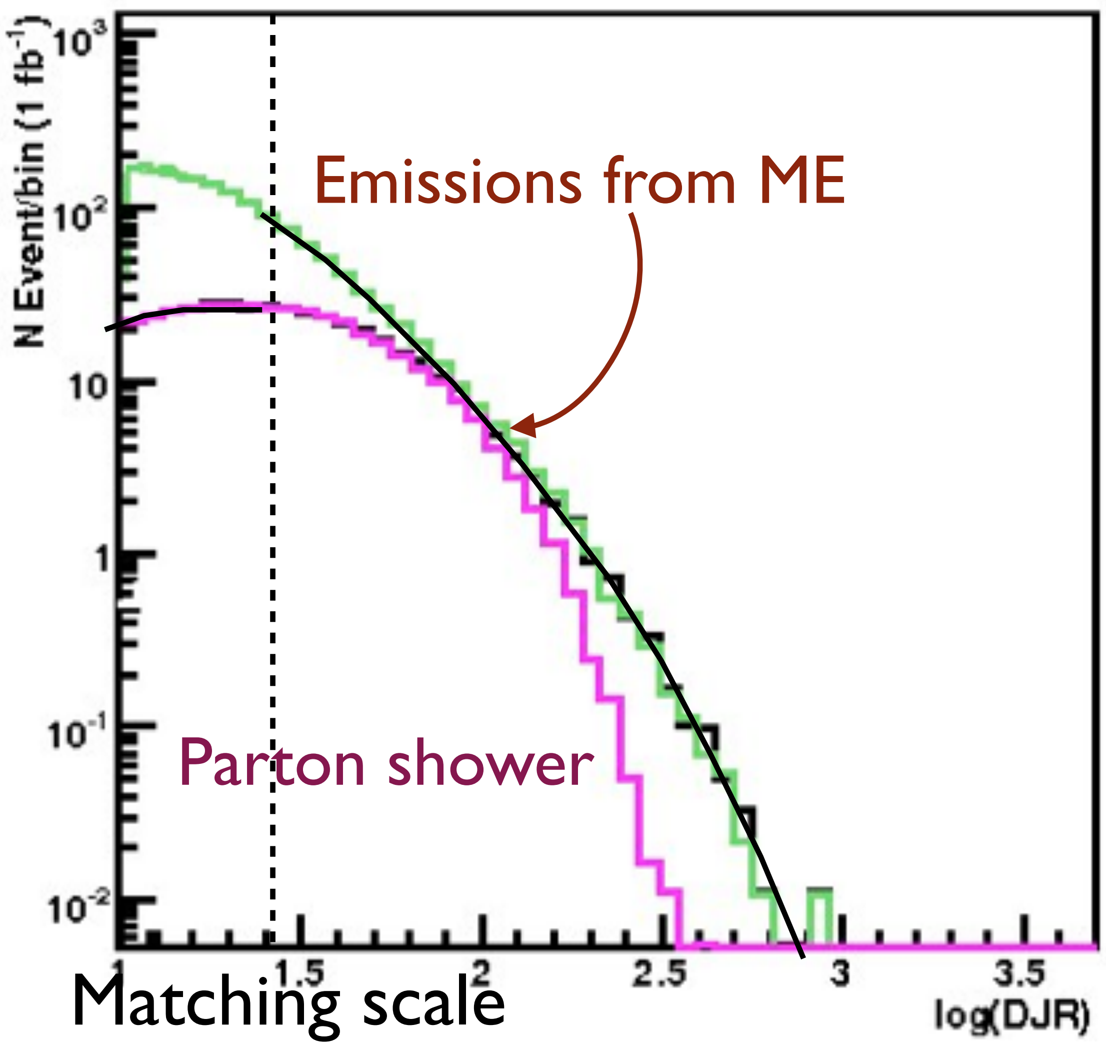
Emissions from PS



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

Back to the "matching goal"

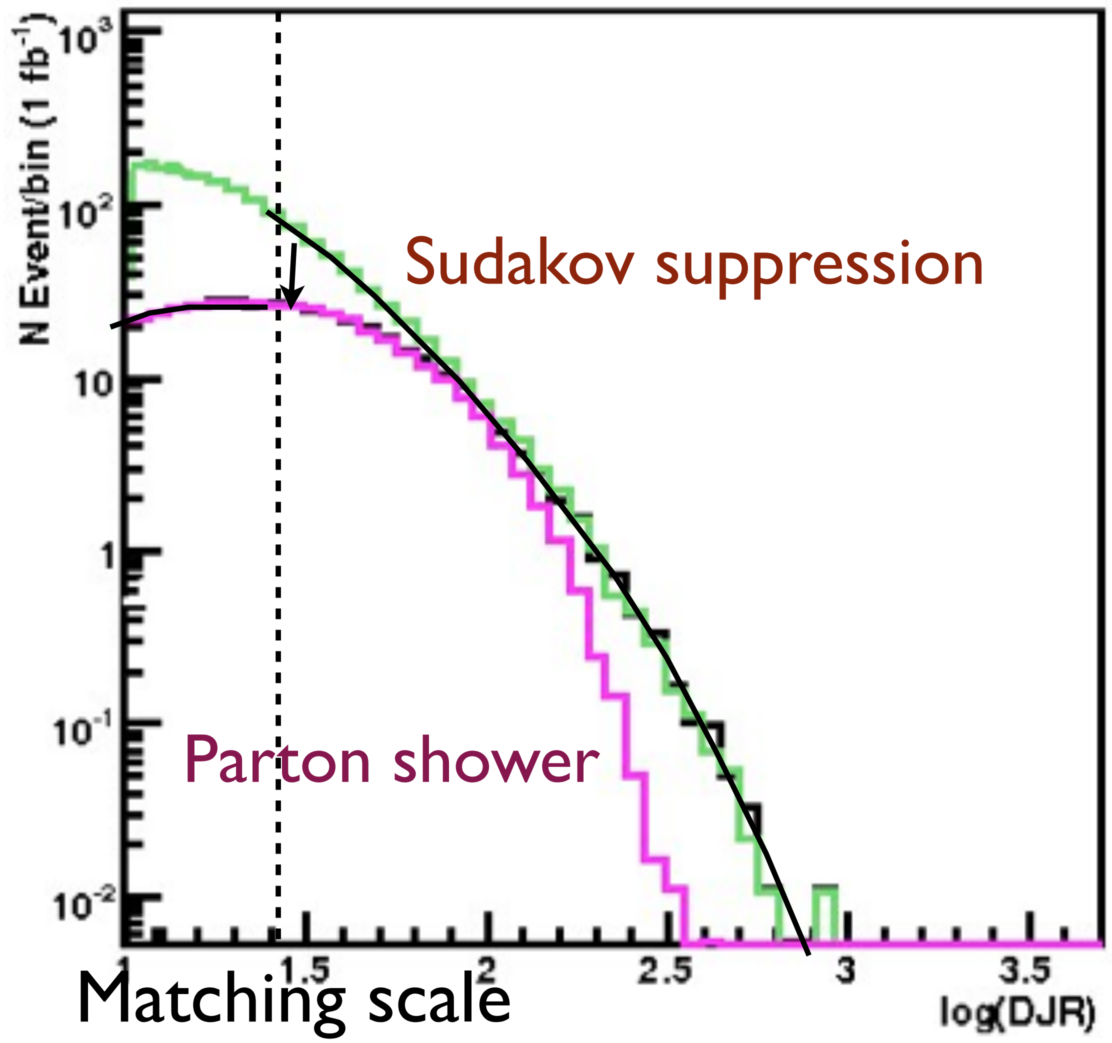
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



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

Back to the "matching goal"

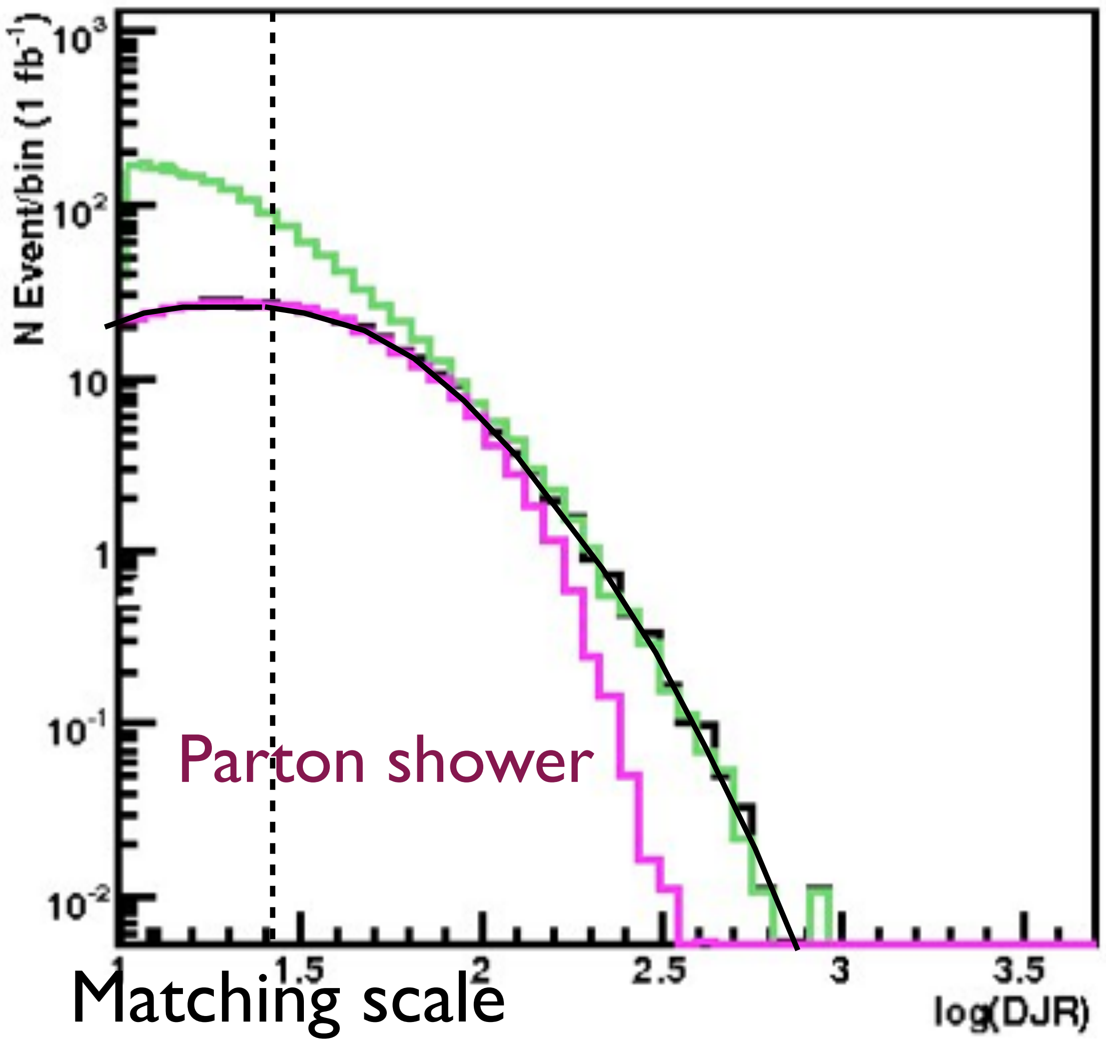
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



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

Back to the "matching goal"

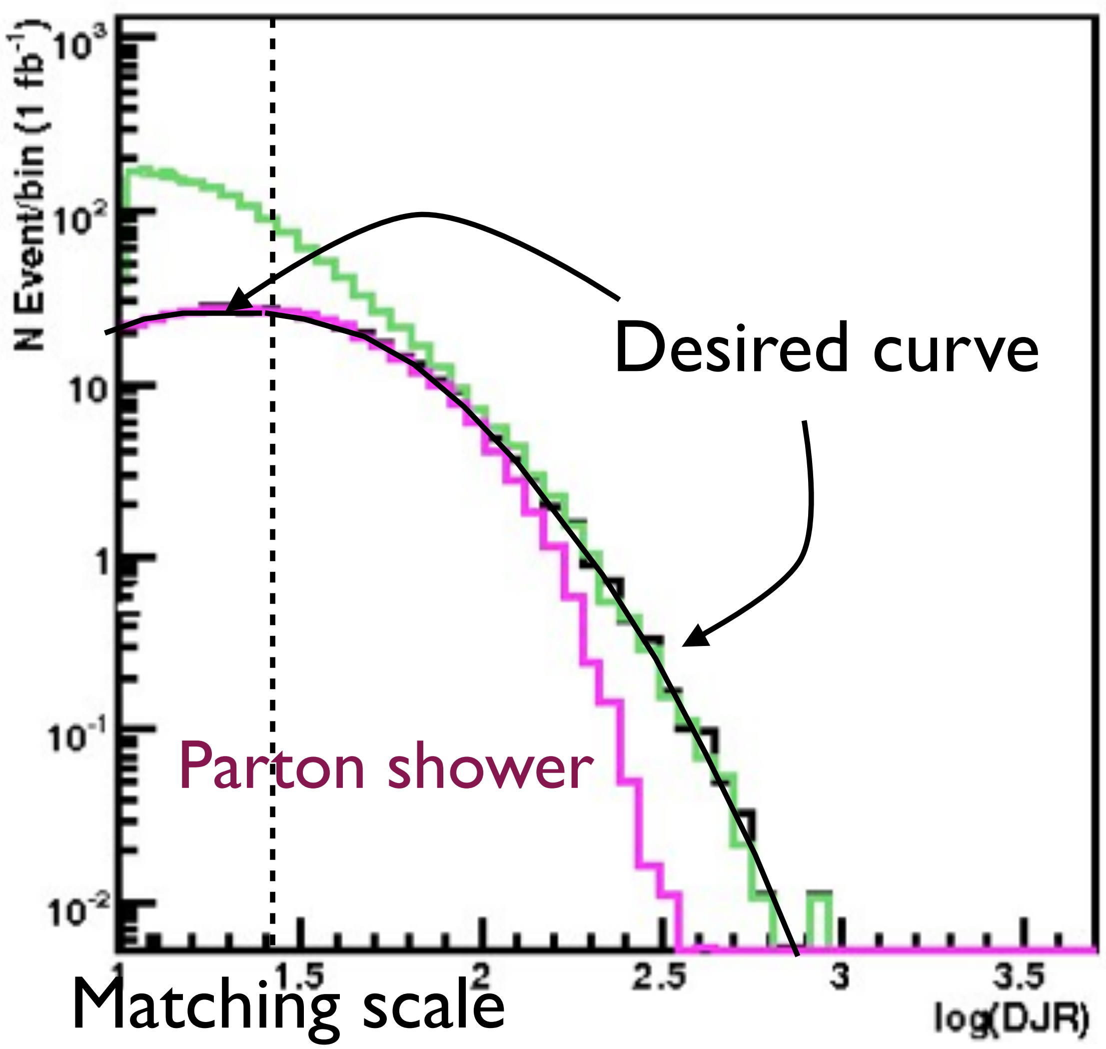
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



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

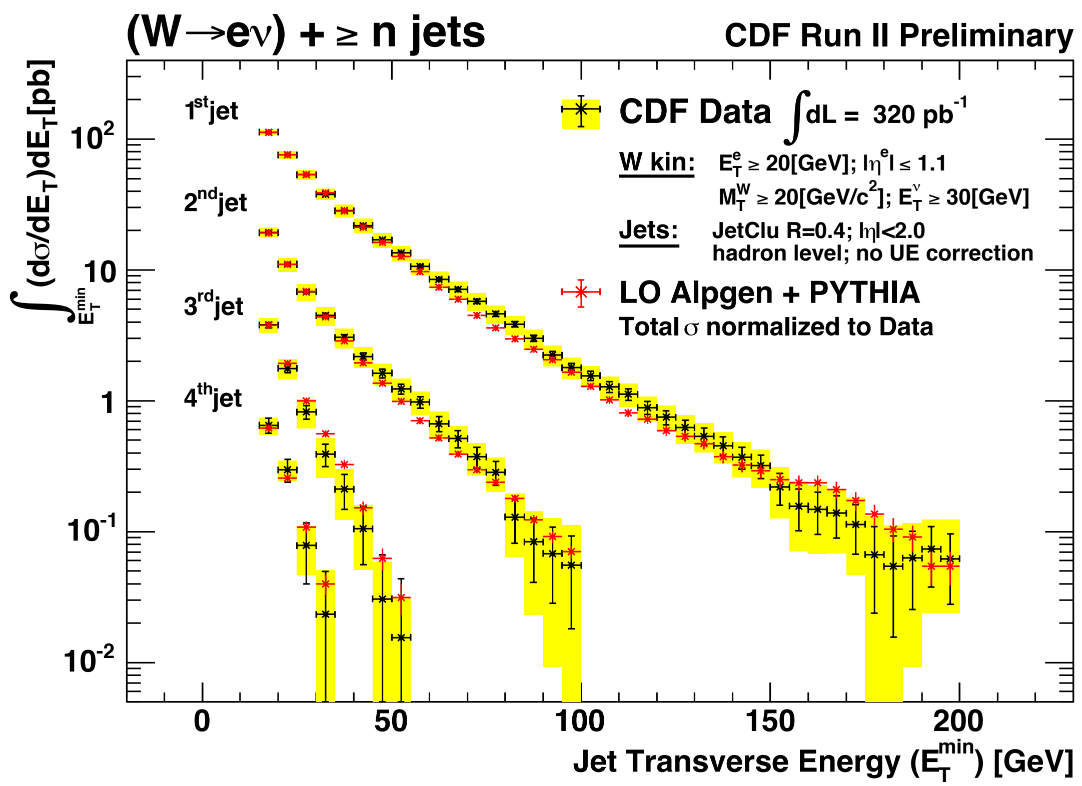


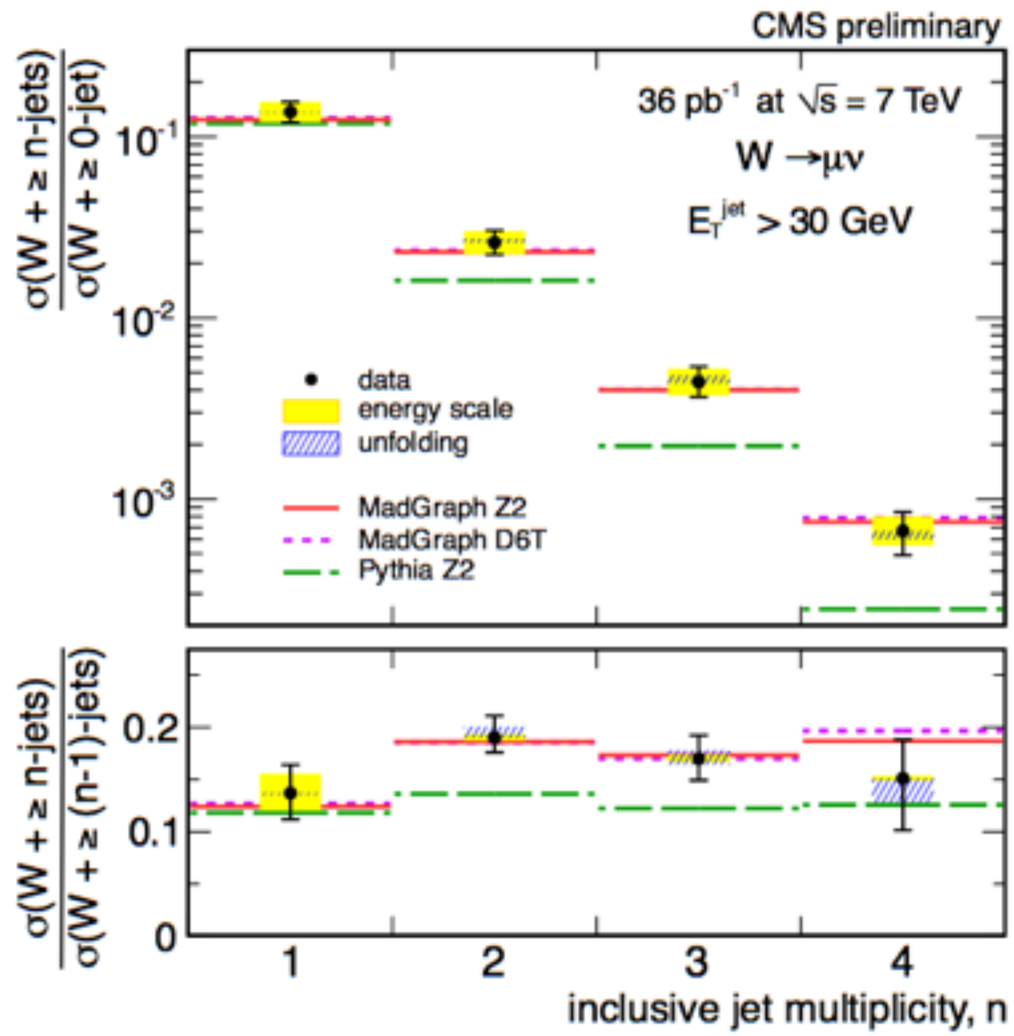
2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

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



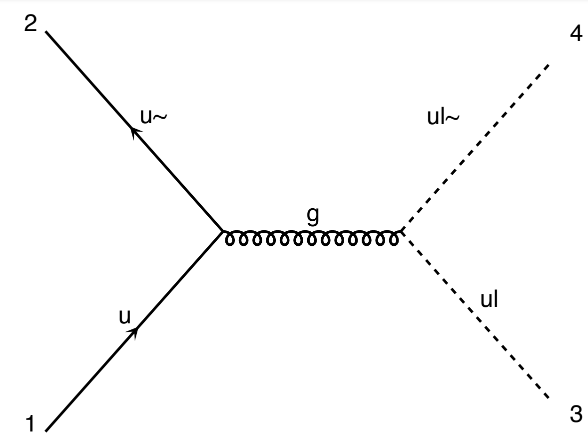


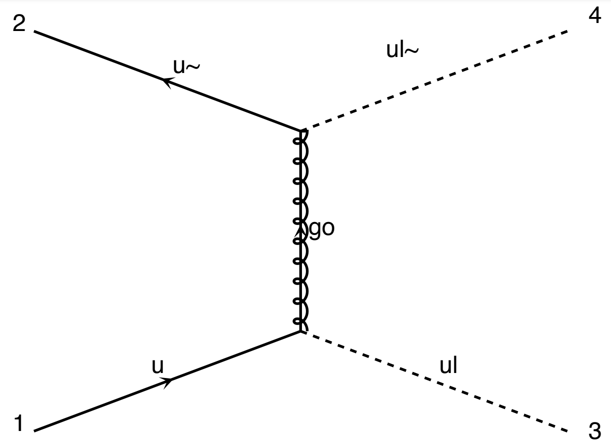
- Very good agreement at Tevatron (left) and LHC (right)
- Matched samples obtained via different matching schemes (MLM and CKKW) consistent within the expected uncertaintes.
- Pure parton shower (Pythia) doesn't describe the data beyond 1st jet.

Merging and Decay

- Double counting of decay with jets
  - ➡ Especially for BSM

Squark pair production

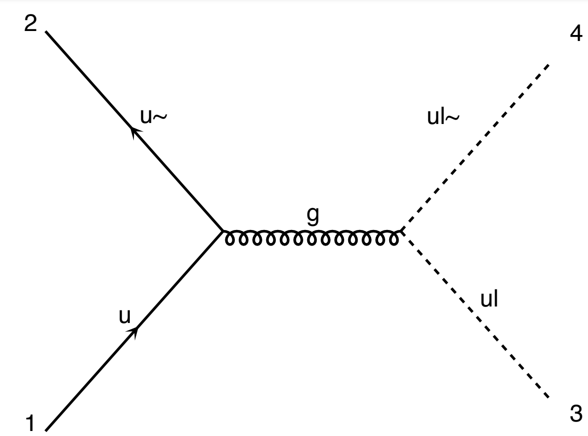


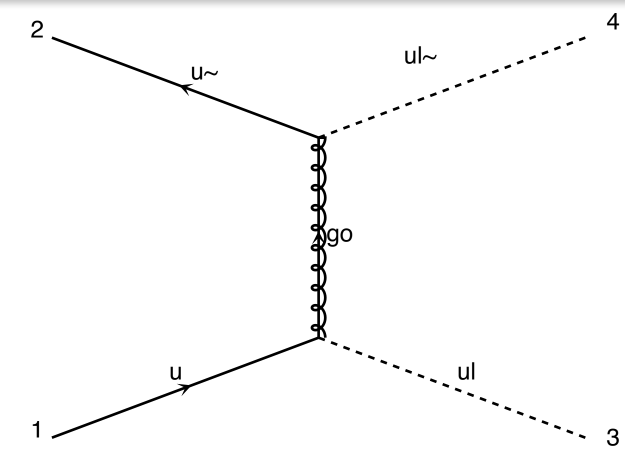


Merging and Decay

- Double counting of decay with jets
  - ➡ Especially for BSM

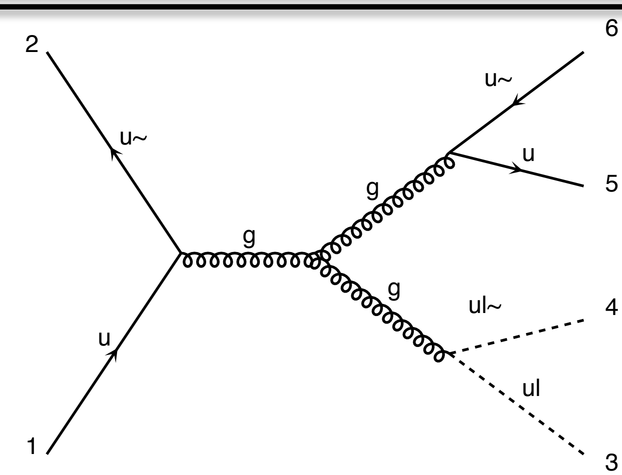
Squark pair production

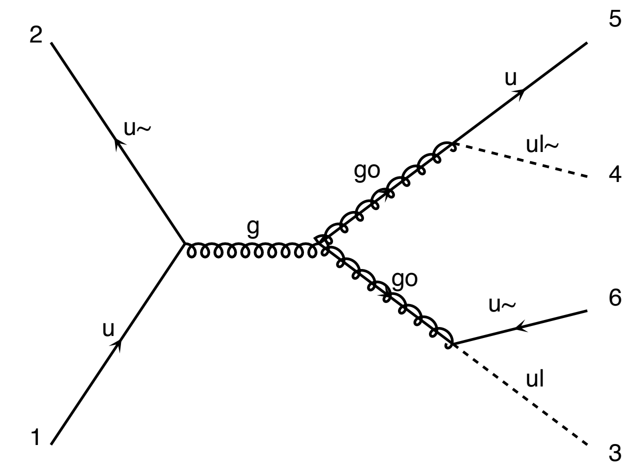




2j

- Presence of new production mechanism

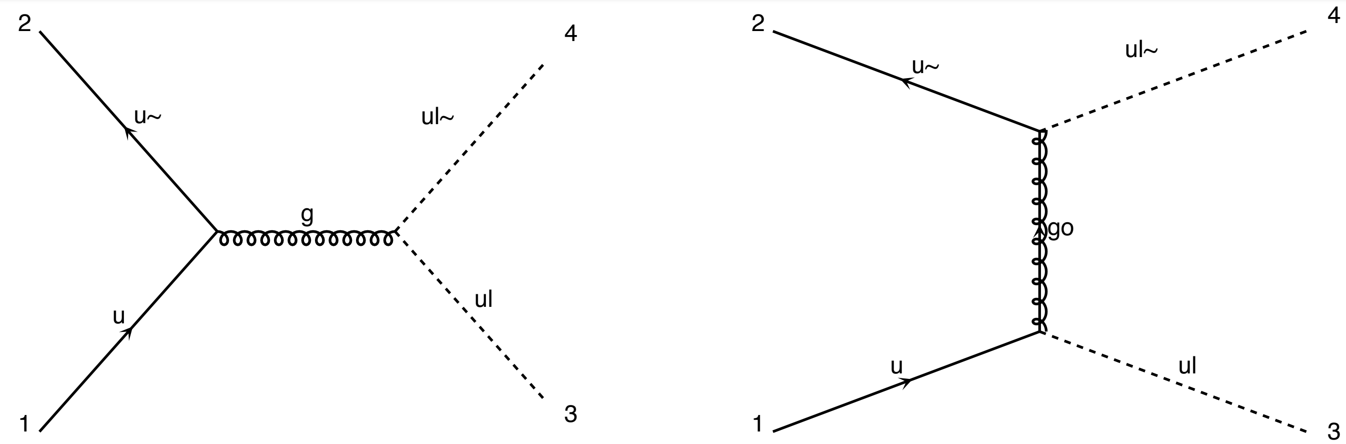




Merging and Decay

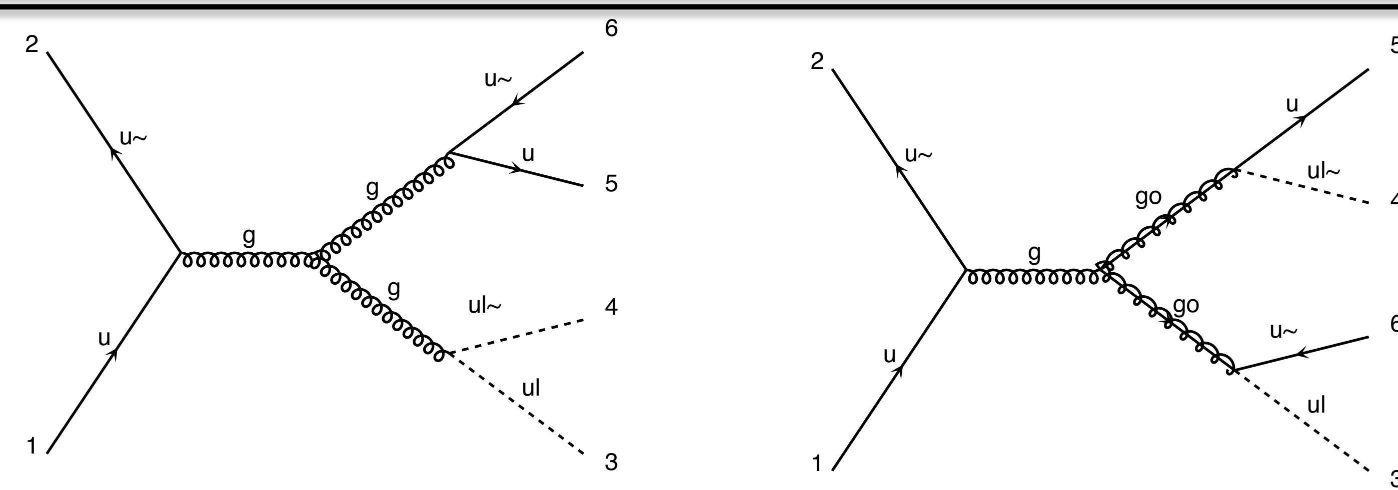
- Double counting of decay with jets
  - ➡ Especially for BSM

Squark pair production



2j

- Presence of new production mechanism



- Those sample should be consider as gluino production if gluino is on shell

Merging and Decays

- This has been solved in recent versions of MadGraph 5 by the new “$” syntax

```
mg5> import model mssm
mg5> generate p p > dr dr~ j j $ go
```

Merging and Decays

- This has been solved in recent versions of MadGraph 5 by the new "$" syntax

```
mg5> import model mssm
mg5> generate p p > dr dr~ j j $ go
```

- This removes any on-shell gluinos from the event generation (where on-shell is defined as $m \pm n \cdot \Gamma$ with n set by **`bwcutoff`** in the run_card.dat)

Merging and Decays

- This has been solved in recent versions of MadGraph 5 by the new "$" syntax

```
mg5> import model mssm
mg5> generate p p > dr dr~ j j $ go
```

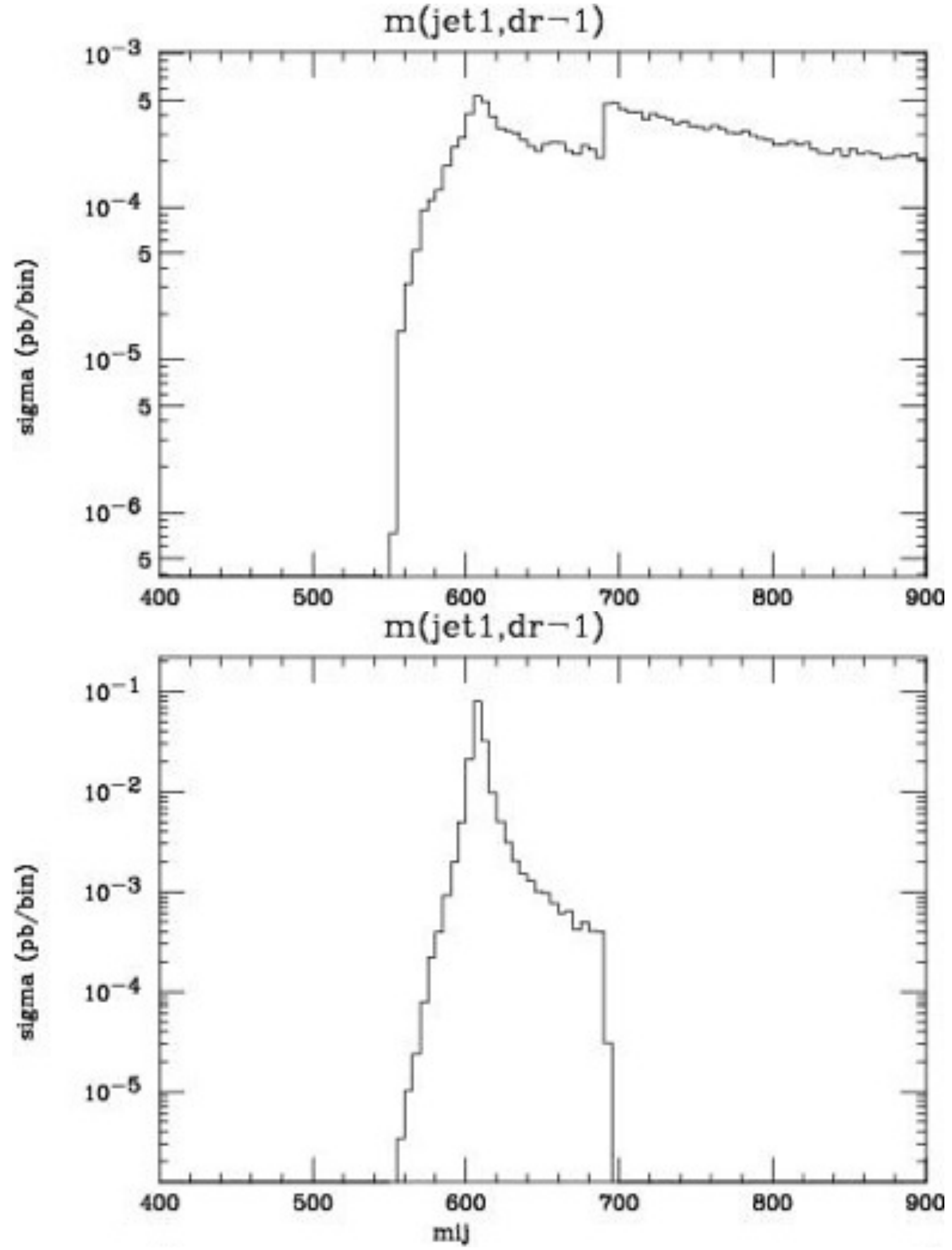
- This removes any on-shell gluinos from the event generation (where on-shell is defined as $m \pm n \cdot \Gamma$ with n set by **`bwcutoff`** in the run_card.dat)
- The corresponding region is exactly filled if you run gluino production with gluinos decaying to dr j (using the same **`bwcutoff`**).

Merging and Decays

Invariant mass distributions of d_r squark and d quark

p p > dr dr~ d $ go

p p > dr go, go > dr~ d



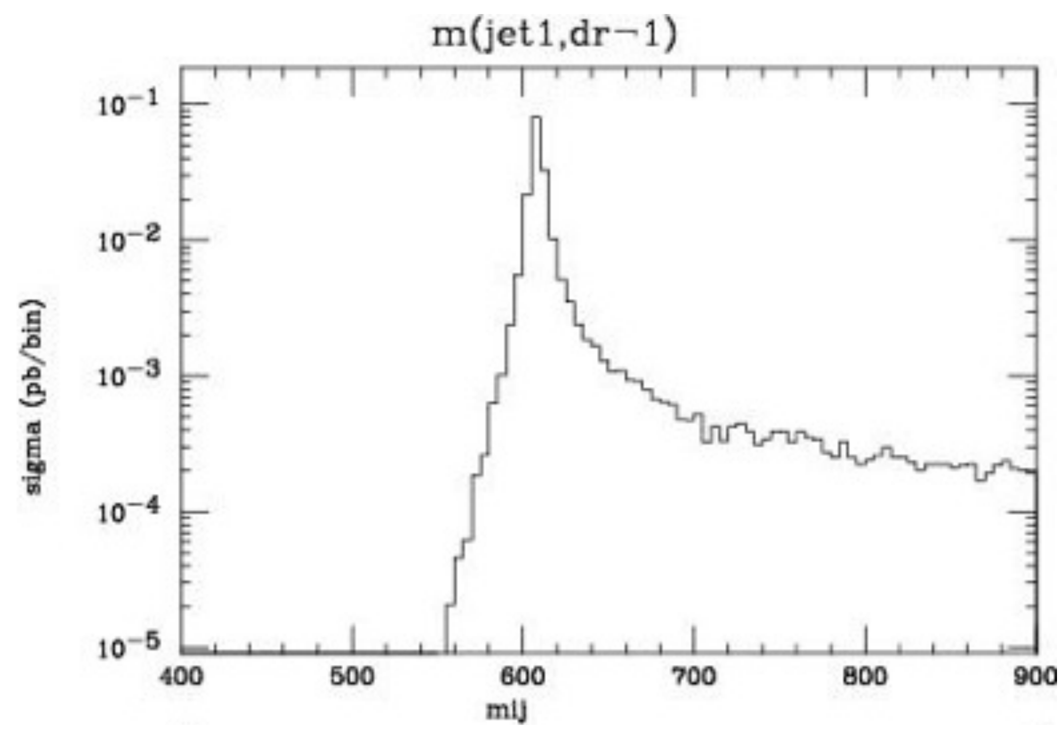
Merging and Decays

Invariant mass distributions
of d_r squark and d quark

p p > dr dr~ d $ go

+

p p > dr go, go > dr~ d



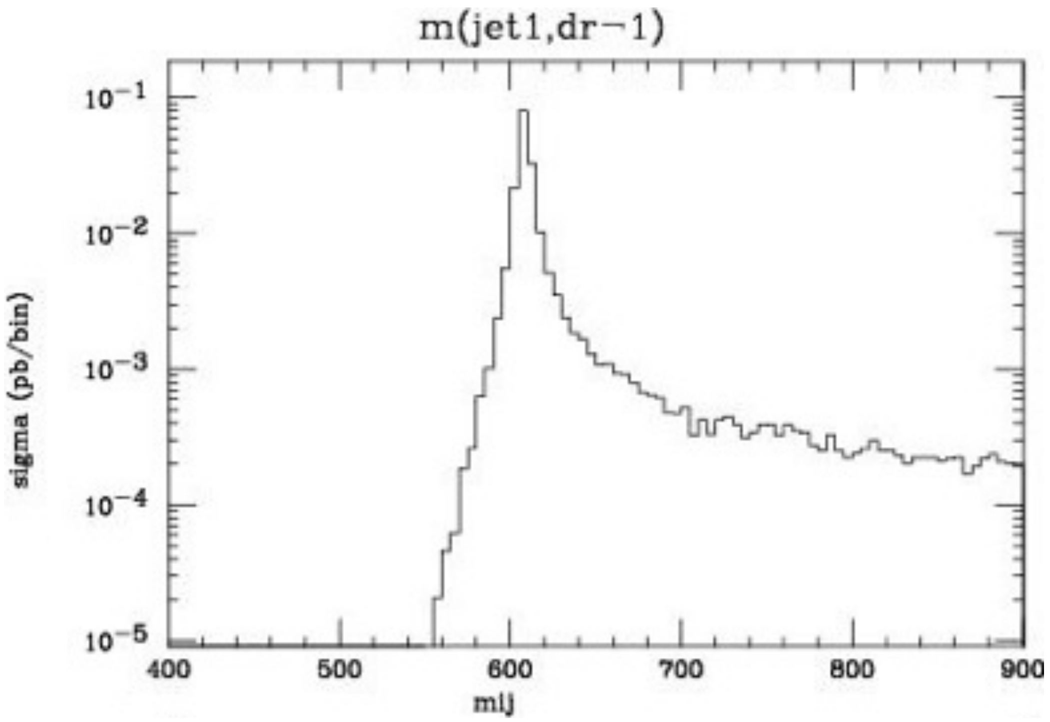
Merging and Decays

Invariant mass distributions
of d_r squark and d quark

p p > dr dr~ d $ go

+

p p > dr go, go > dr~ d



Double counting between samples completely removed!

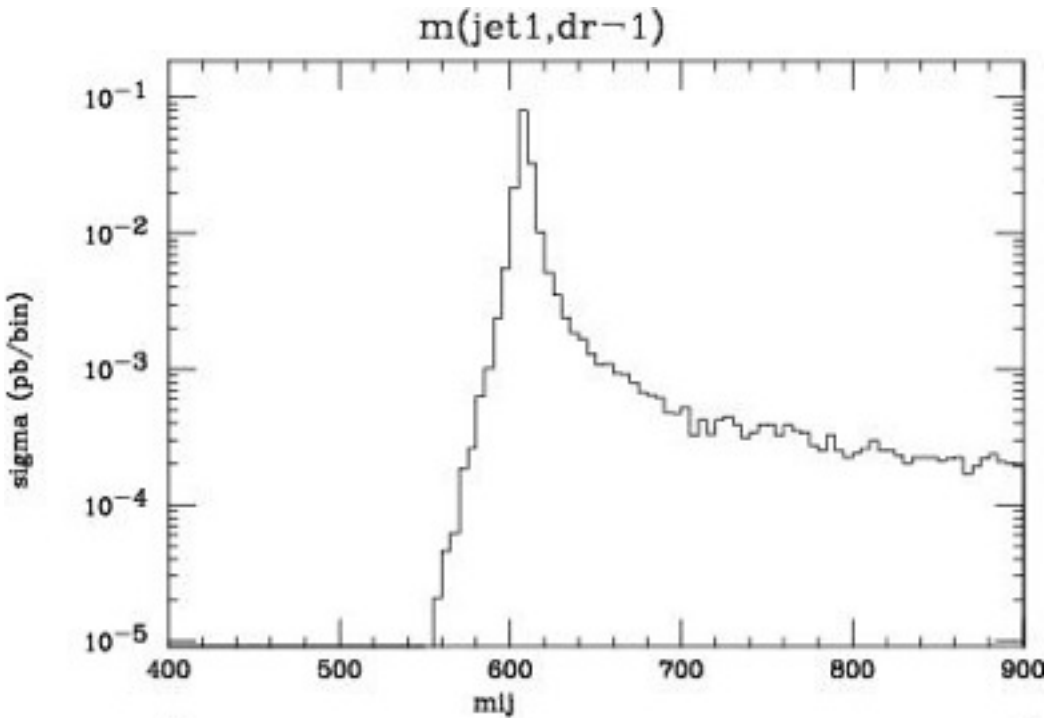
Merging and Decays

Invariant mass distributions
of d_r squark and d quark

p p > dr dr~ d $ go

+

p p > dr go, go > dr~ d



Double counting between samples completely removed!

Jet correctly handle by the matching/merging

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!

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:

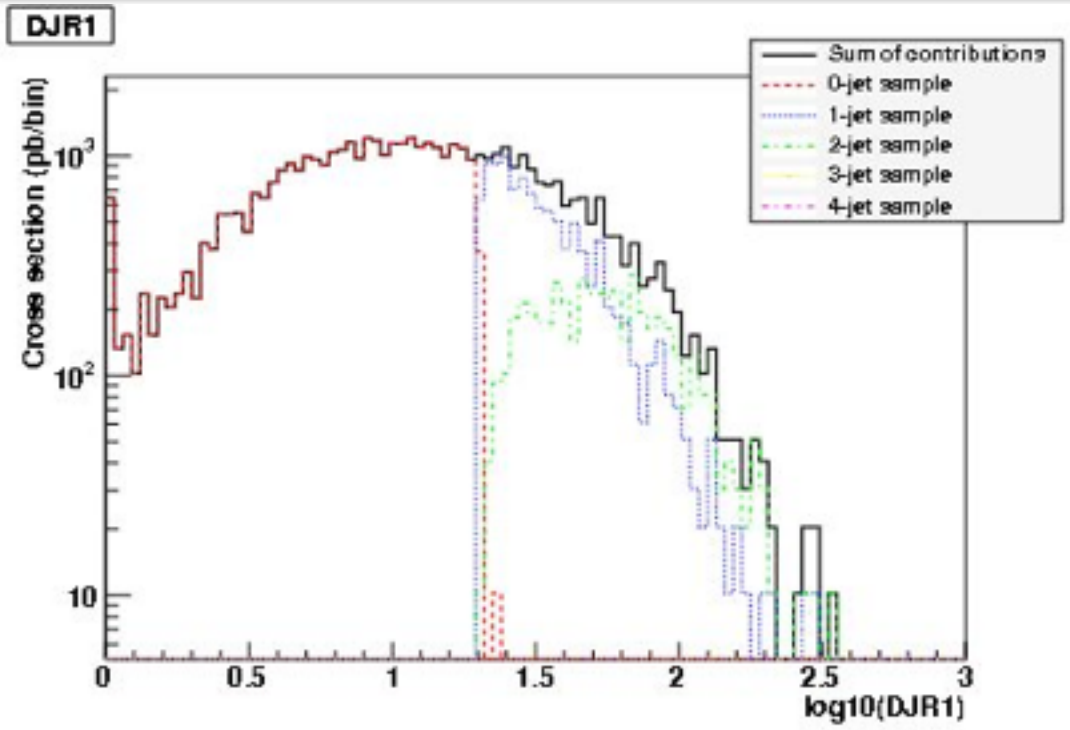
```
In pythia_card.dat:
…
! This sets the matching scale, needs to be > xqcut
QCUT = 30
! This switches from kT-MLM to shower-kT matching
! Note that MSTP(81)>=20 needed (pT-ordered shower)
SHOWERKT = T
```

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

Differential Jet Rate Plot

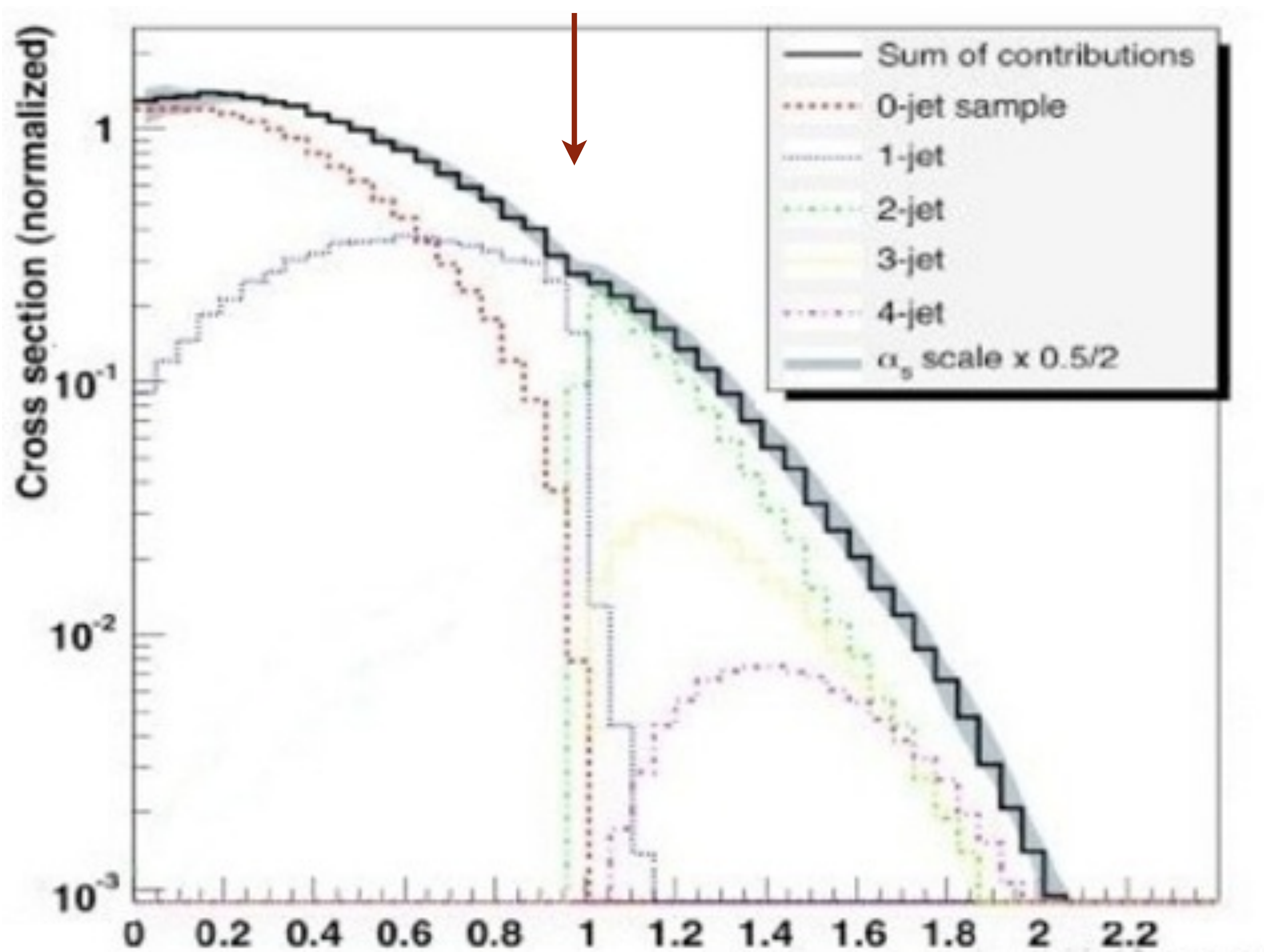
- This are the clustering scales in the kt-jet clustering scheme
- DJR1: pT of the last remaining jet
- DJR2: The minimum between the pT of the second to last remaining jet and the kt between the last two jet.
- Only radiative jet (not those from decay) should enter those plot.



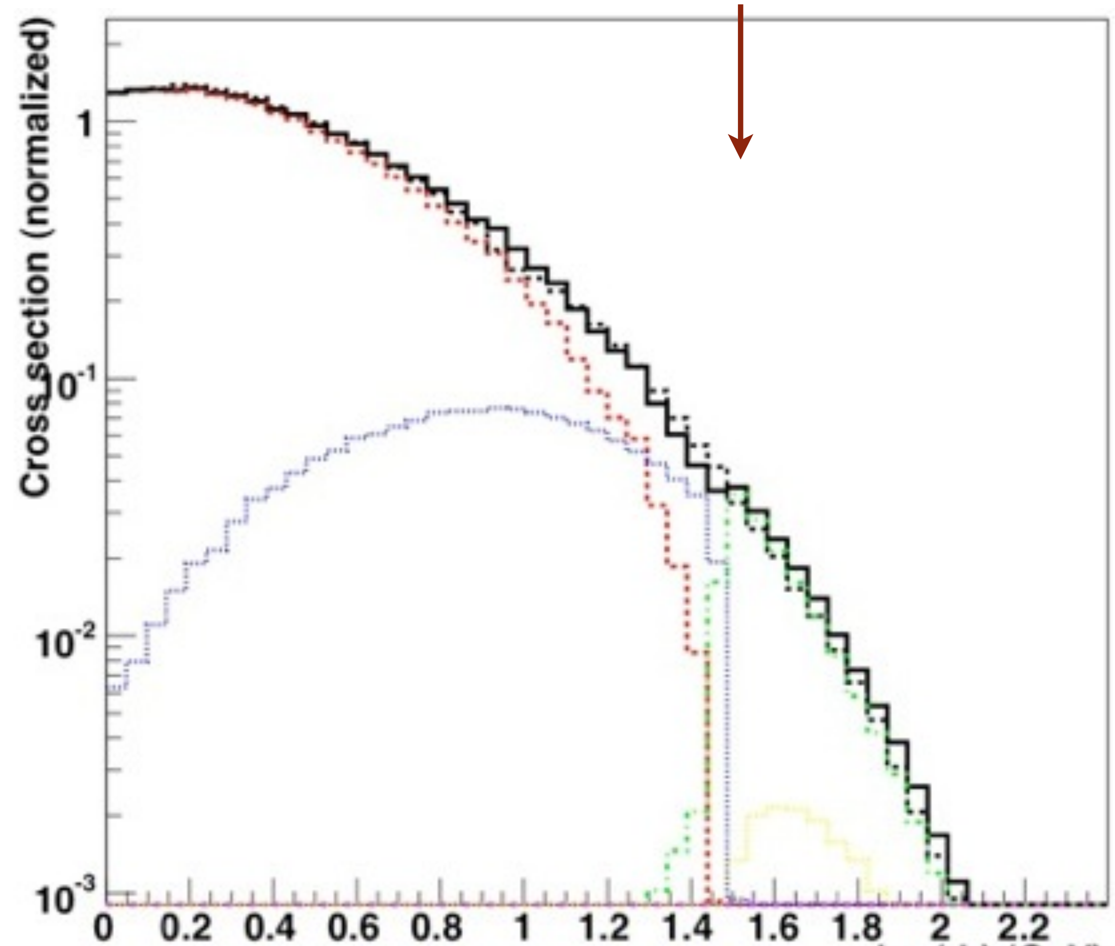
Matching Validation

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

Q^{match} = 10 GeV



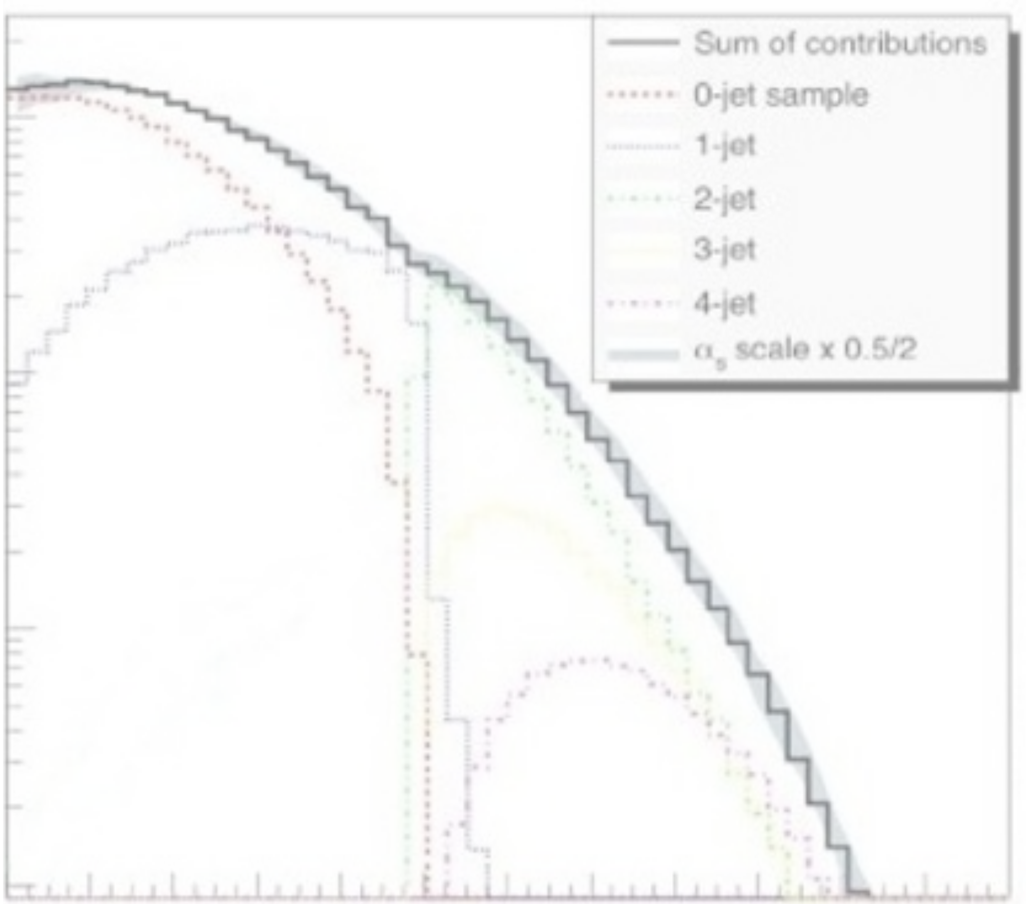
Q^{match} = 30 GeV

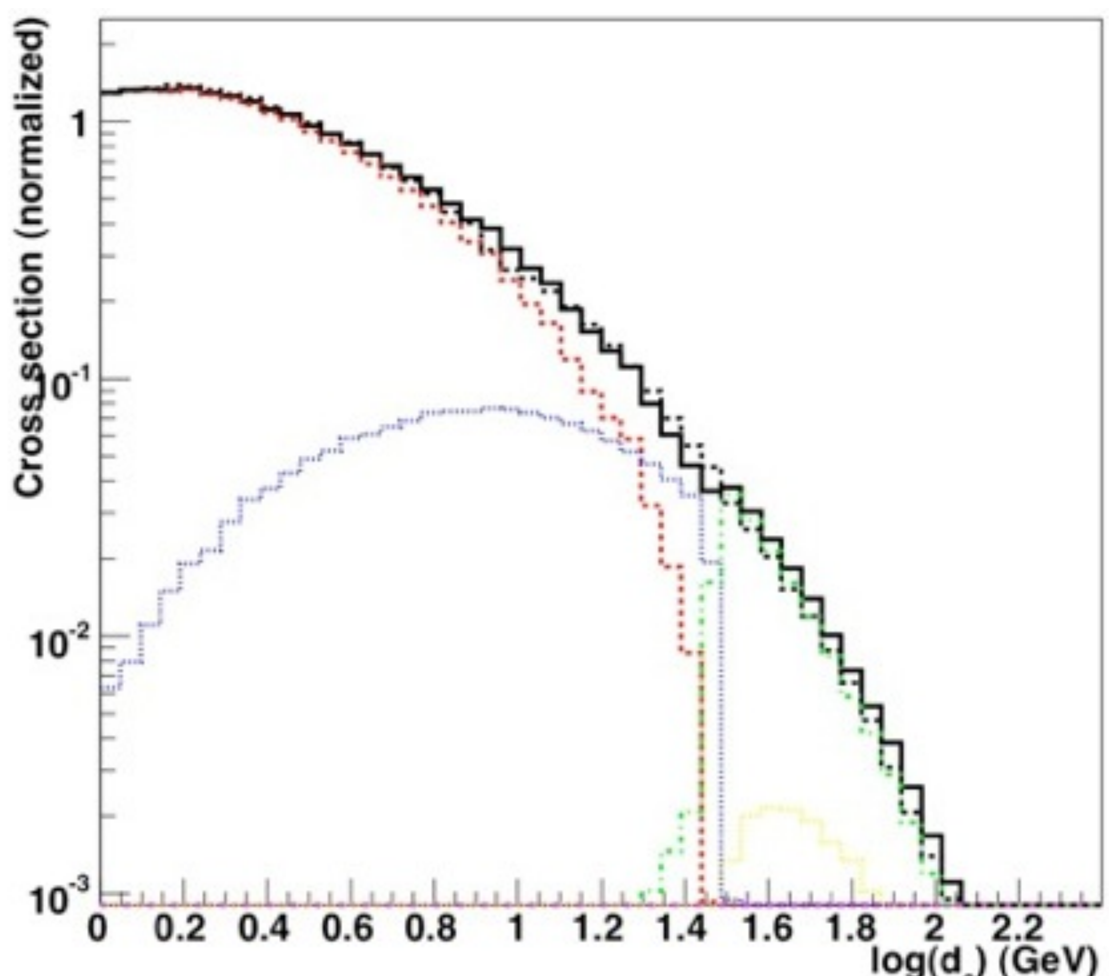


log(Differential jet rate for 1 → 2 radiated jets ~ pT(2nd jet))

Matching Validation

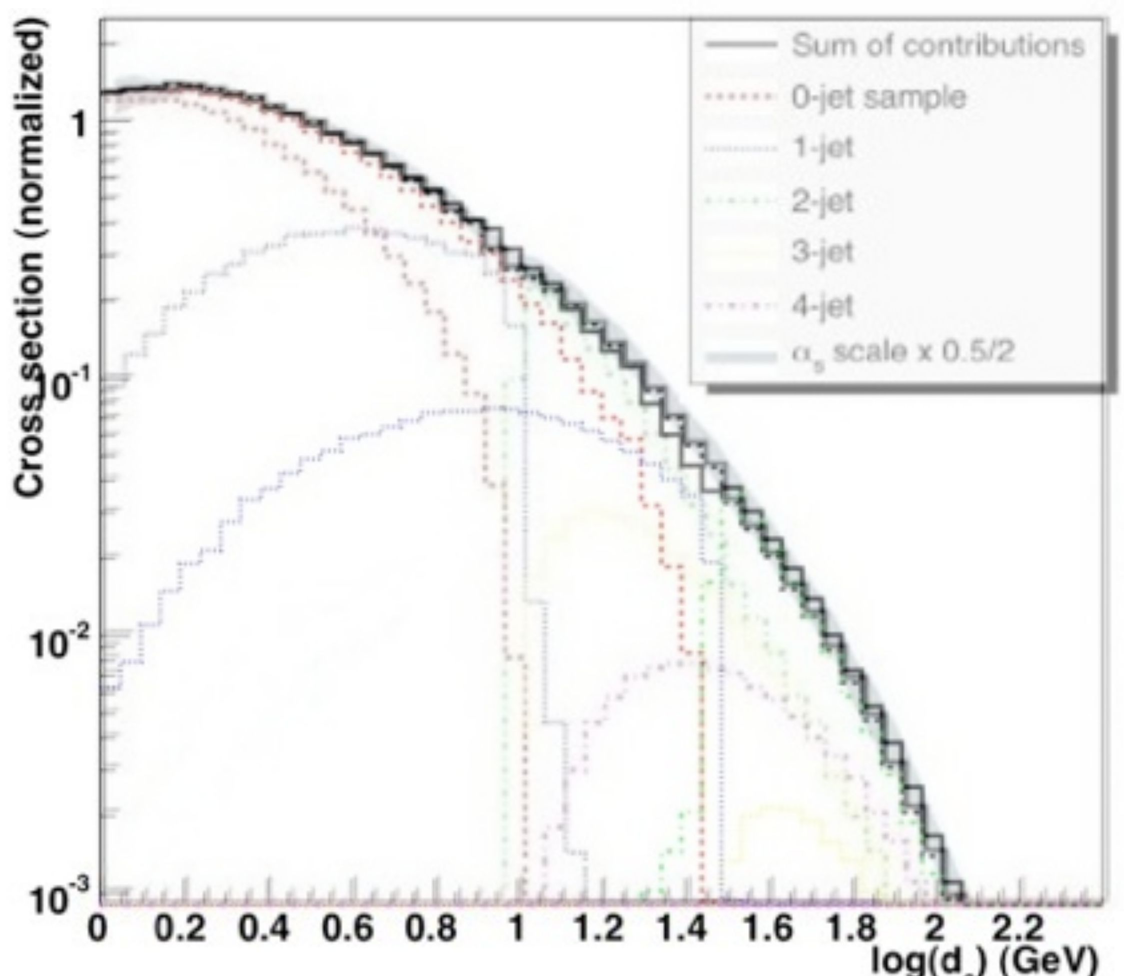
W+jets production at the Tevatron for MadGraph+Pythia
(k_T-jet MLM scheme, q^2-ordered Pythia showers)



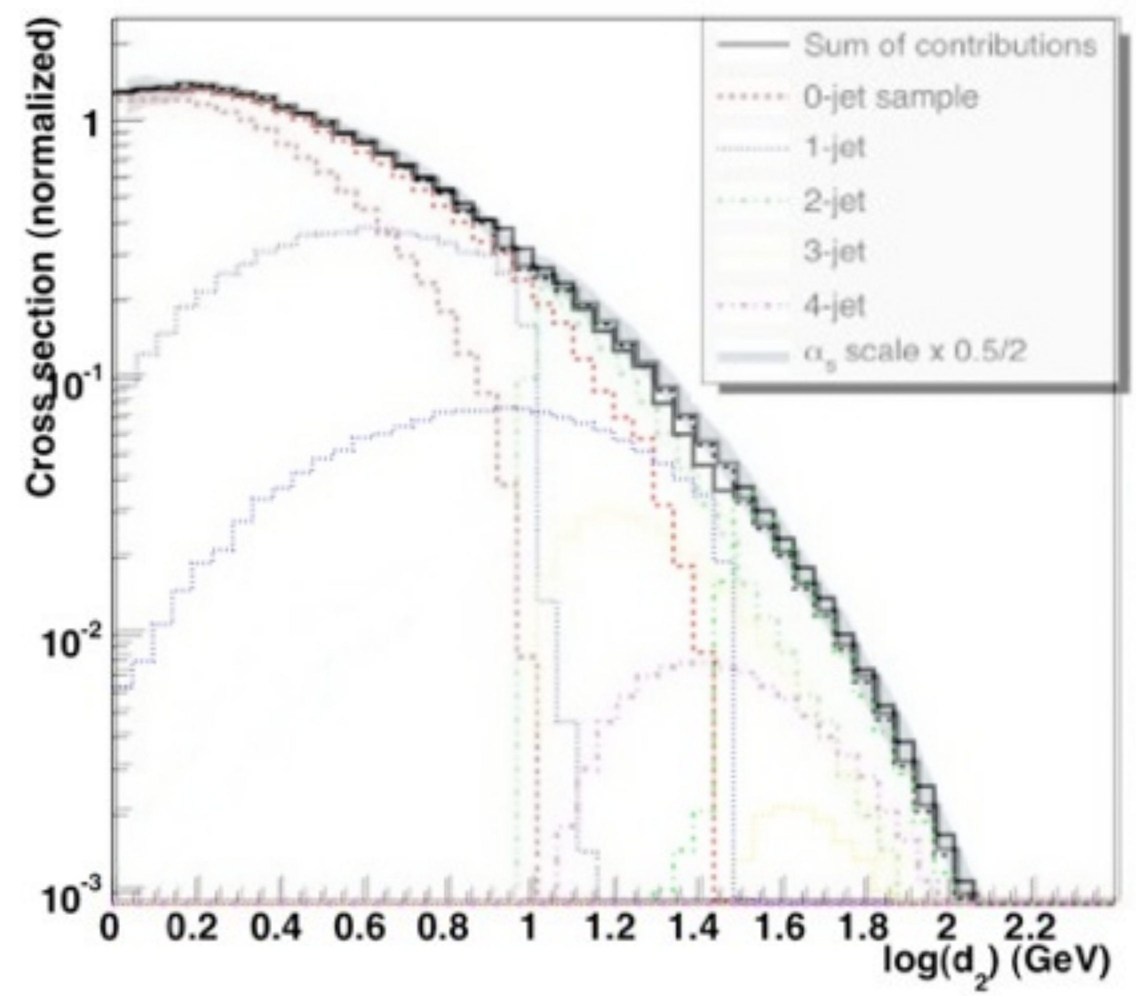


Matching Validation

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



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



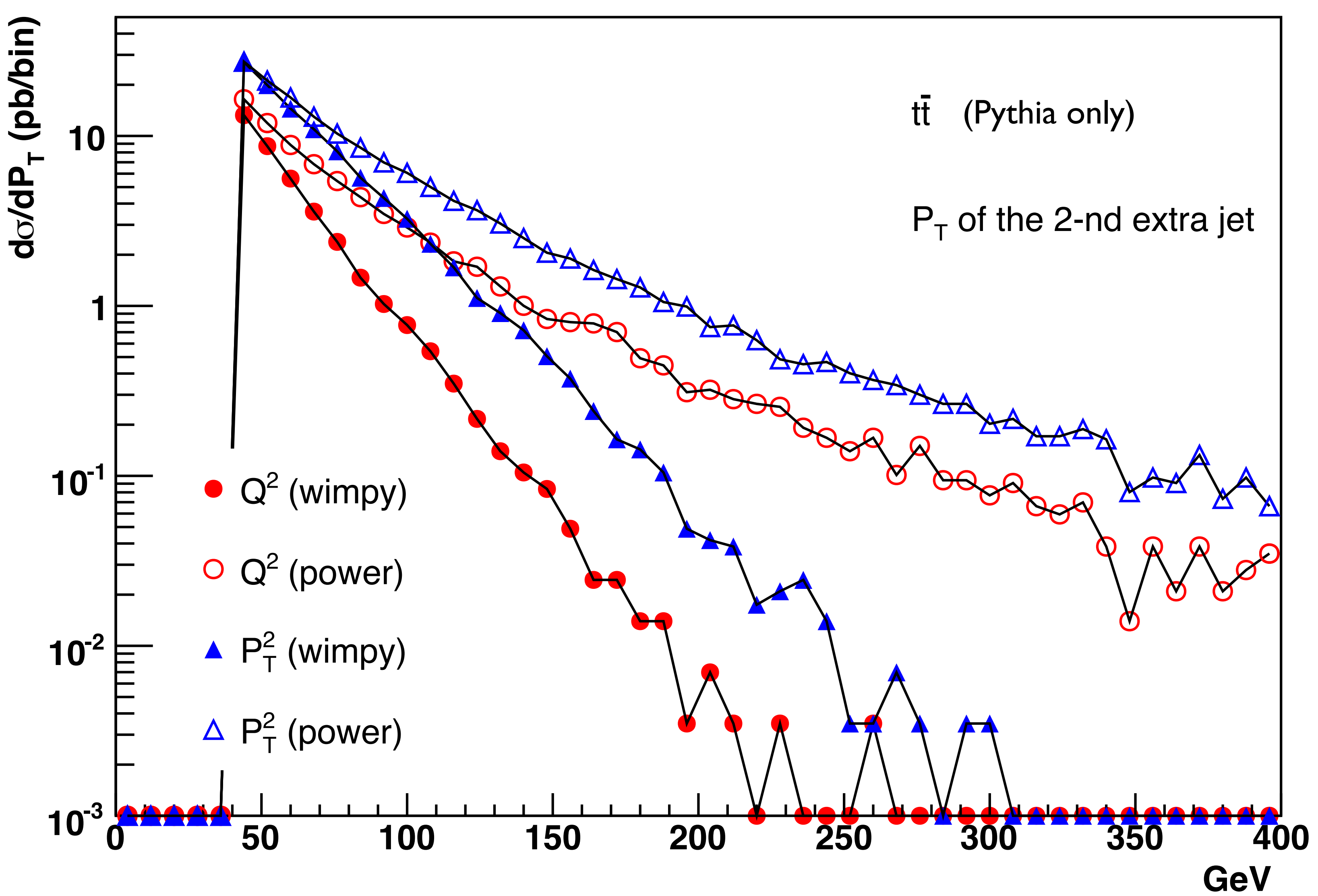
Jet distributions smooth, and stable when we vary the matching scale!

Demo

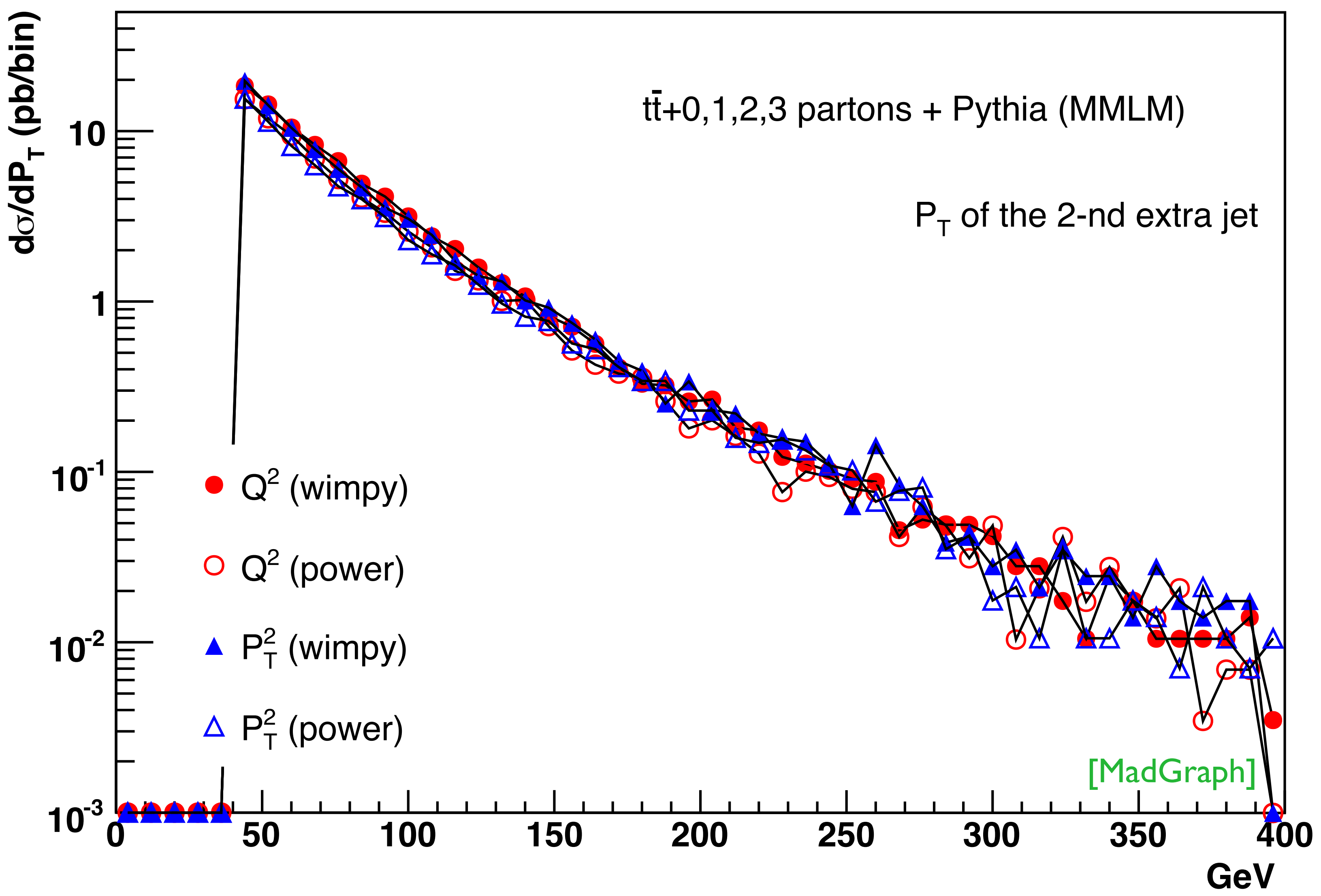
- Time for live Demonstration!

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)



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



Lecture 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