MadGraph 5 Developments and Plans

Johan Alwall
National Taiwan University

for the MadGraph team

ATLAS/CMS/LPCC MC workshop, CERN
2012/11/19

MadGraph 5 Developments and Plans

Olivier Mattelaer
University of Illinois at Urbana-Champaign

for the MadGraph team

(Most slide from Johan Alwall)

ATLAS/CMS/LPCC MC workshop, CERN
2012/11/19

Huge news!

Completely automated simulations at next-to-leading order in QCD, matched to shower, now public (aMC@NLO in MG5 v. 2.0 beta)!

- No-need-for-theorist NLO simulations that can be immediately interfaced to detector sim, for any 2→3 and many 2→4 and higher processes within the SM, at reasonable computer time
- You will hear (and have already heard!) much more about this from other speakers at this meeting!

- So I will focus on "LO" developments
- As per request, special focus on matching with b-quarks (public, but still under validation)
- Also present some exciting upcoming developments in the next few months!

Core news since MG5 release - Quick list

- Lots of speedups and improvements, including
  - ➡ Huge speedup of gridpacks
  - ➡ Vast speedup for long decay chains with multiparticle decays
  - ➡ Huge improvements in user interface
  - ➡ Multi cluster support
- New 4-flavor matching and VBF-type matching
- 4 fermion vertices in FR+MG5 (except Majorana)
- Spin 3/2 particles in FR+MG5
- Complex mass scheme
- Feynman gauge
- Handling of negative weights
- On-the-fly 2-body decay width calculations (“Auto width”)

News from our friends

- FeynRules (C. Duhr, B. Fuks et al)
  - ➡ SLagrangians in superspace formalism
  - ➡ Automatic 2-body width expressions
  - ➡ Automatic renormalization group equations (soon!)
  - ➡ Automatic mass matrix diagonalization (soon!)
- MadAnalysis 5 (B. Fuks et al)
  - ➡ Super flexible, fast, user friendly analysis suite
  - ➡ Arbitrary weights (also negative)
  - ➡ Automatic systematics uncertainty bands (soon!)
- MadGolem (Goncalves-Netto, Plehn et al)
  - ➡ Automated NLO SUSY production cross sections

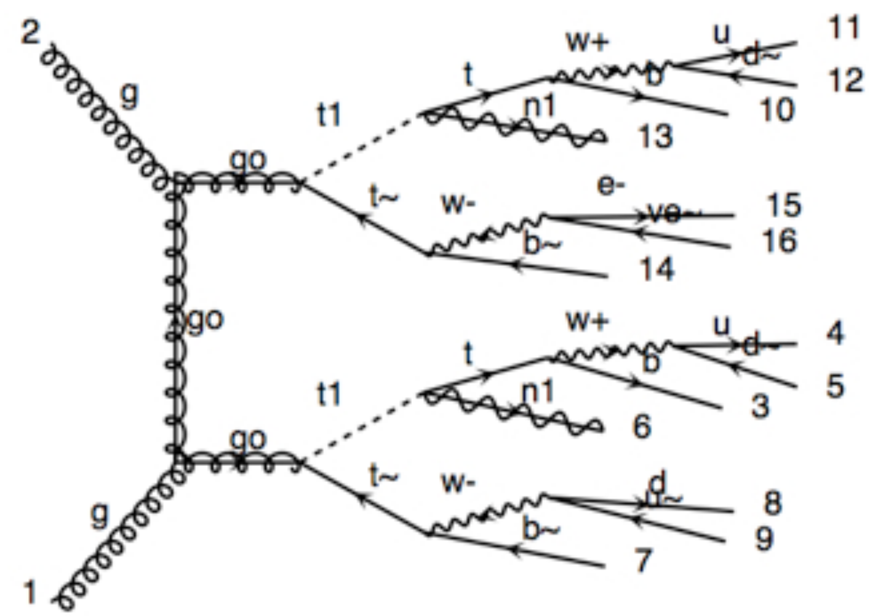
"LO" upcoming developments - Quick list

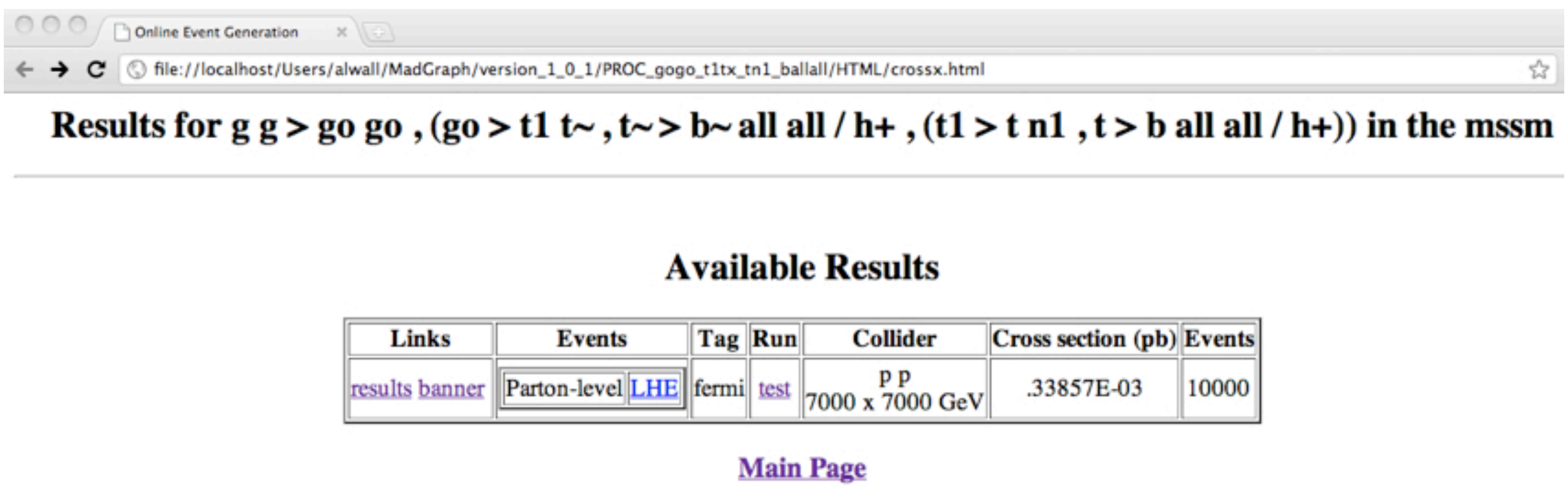
- Automatic matching to Pythia 8
  - ➡ CKKW-L matching (S. Pretzel et al)
- Automatic scale/PDF/matching systematics (A. Kalogeropoulos et al)
- Fast multiparton processes using color-ordered recursion
- MadSpin (full spin correlations for LO&NLO decays) (P. Artoisenet et al)
- MadDecay: Automatic BSM decay width suite (including needed 3- and 4-body widths and decay of event file) (C-H Shen et al)
  - ➡ Combined with MadSpin for full spin correlations
- MadDM: Relic density calculations and direct detection limits for any BSM model (K.C. Kong et al)
- MadWeight5: Matrix-Element Method (P. Artoisenet et al)

Recap: "What is MG5"

- MadGraph 5 is a completely new (released spring 2011) matrix element generator written in Python
- Can handle ANY model (that can be written as a Lagrangian), conveniently output by FeynRules
  - ➡ Any Lorentz structure for any spin (up to 2) and color (6tets, ε^{ijk})
  - ➡ Multiparticle vertices for any multiplicity, multifermion vertices
- Super fast process generation
- Unlimited-length decay chains with full BW and spin effects
- Event generation speedups by orders of magnitude
- Output in multiple languages and formats (including Pythia 8)
- Super-user-friendly command line interfaces

Decay chains




Online Event Generation

file://localhost/Users/alwall/MadGraph/version_1_0_1/PROC_gogo_t1tx_tn1_ballall/HTML/crossx.html

Results for g g > go go , (go > t1 t~ , t~ > b~ all all / h+ , (t1 > t n1 , t > b all all / h+)) in the mssm

Available Results

Links	Events	Tag	Run	Collider	Cross section (pb)	Events
results banner	Parton-level LHE	fermi	test	p p 7000 x 7000 GeV	.33857E-03	10000

Main Page

(See backup slides for discussion about physics validity)

User Interface

- Nice interactive session
  - Auto-completion
  - Tutorial
  - interactive help
- Simple command set
  - import model sm
  - generate p p > e+ e-
  - output FORMAT MY_DIR
  - launch

```
*************************************************************
*                                                           *
*          W E L C O M E  to  M A D G R A P H  5            *
*                                                           *
*                                                           *
*                 *                       *                 *
*                   *        * *        *                   *
*                     * * * * 5 * * * *                     *
*                   *        * *        *                   *
*                 *                       *                 *
*                                                           *
*        VERSION 1.3.16                2011-09-11           *
*                                                           *
*    The MadGraph Development Team - Please visit us at     *
*    https://server06.fynu.ucl.ac.be/projects/madgraph      *
*                                                           *
*               Type 'help' for in-line help.               *
*           Type 'tutorial' to learn how MG5 works          *
*                                                           *
*************************************************************
load MG5 configuration from /Users/omatt/.mg5_config
Loading default model: sm
models.import_ufo: Restrict model sm with file models/sm/rest
models.import_ufo: Run "set stdout_level DEBUG" before import
INFO: Change particles name to pass to MG5 convention
Defined multiparticle p = g u c d s u~ c~ d~ s~
Defined multiparticle j = g u c d s u~ c~ d~ s~
Defined multiparticle l+ = e+ mu+
Defined multiparticle l- = e- mu-
Defined multiparticle vl = ve vm vt
Defined multiparticle vl~ = ve~ vm~ vt~
mg5>help
```

User Interface

- Nice Interactive session
  - Auto-completion
  - Tutorial
  - interactive help

If You test it, you are going to like it !

- Simple command set
  - import model sm
  - generate p p > e+ e-
  - output FORMAT MY_DIR
  - launch

```
************************************************************
*                                                          *
*           W E L C O M E  to  M A D G R A P H  5          *
*                                                          *
*                                                          *
*                 *                       *                *
*                   *        * *        *                  *
*                     * * * * 5 * * * *                    *
*                   *        * *        *                  *
*                 *                       *                *
*                                                          *
*         VERSION 1.3.16                2011-09-11         *
*                                                          *
*    The MadGraph Development Team - Please visit us at    *
*                                        jects/madgraph    *
*                                                          *
*               Type 'help' for in-line help.              *
*           Type 'tutorial' to learn how MG5 works         *
*                                                          *
************************************************************
load MG5 configuration from /Users/omatt/.mg5_config
Loading default model: sm
models.import_ufo: Restrict model sm with file models/sm/rest
models.import_ufo: Run "set stdout_level DEBUG" before import
INFO: Change particles name to pass to MG5 convention
Defined multiparticle p = g u c d s u~ c~ d~ s~
Defined multiparticle j = g u c d s u~ c~ d~ s~
Defined multiparticle l+ = e+ mu+
Defined multiparticle l- = e- mu-
Defined multiparticle vl = ve vm vt
Defined multiparticle vl~ = ve~ vm~ vt~
mg5>help
```

Matching in MG + Pythia 6

J.A. et al. [arXiv:0706.2569],
J.A., de Visscher, Maltoni [arXiv:0810.5350],
MadGraph wiki

- MLM-style matching
  - ➡ Classic cone-jet matching (a la AlpGen), k_T-jet matching, and "shower-k_T" matching
- Easily adopted to different shower algorithms (e.g., virtuality- and p_T-ordered showers in Pythia 6)
- Official CMS SM background simulation
  - ➡ Excellent agreement with data across the line
- Fully supports matching for any NP process
- Restriction: Matching of jets from core process, not from decay products (let Pythia ME corr. take care of that)

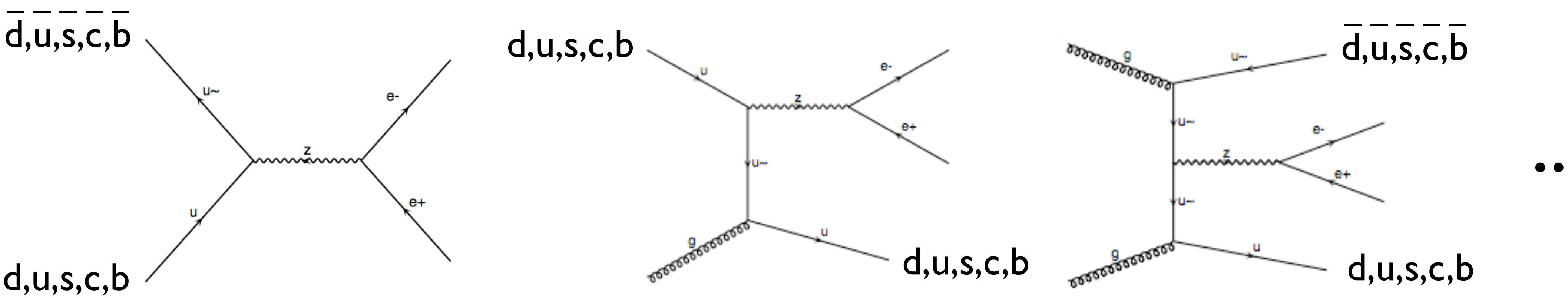
MLM algorithm in a nutshell

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 corresponding clustering vertices
3. Run the parton shower with starting scale $t_0 = m_T$.
4. Check that the number of jets after parton shower is the same as ME partons, and that all jets after parton shower are matched to the ME partons at a scale Q^{match}.
   If yes, keep the event. If no, reject the event. Q^{match} is called the *matching scale*.
5. For highest multiplicity, allow radiation < lowest ME scale

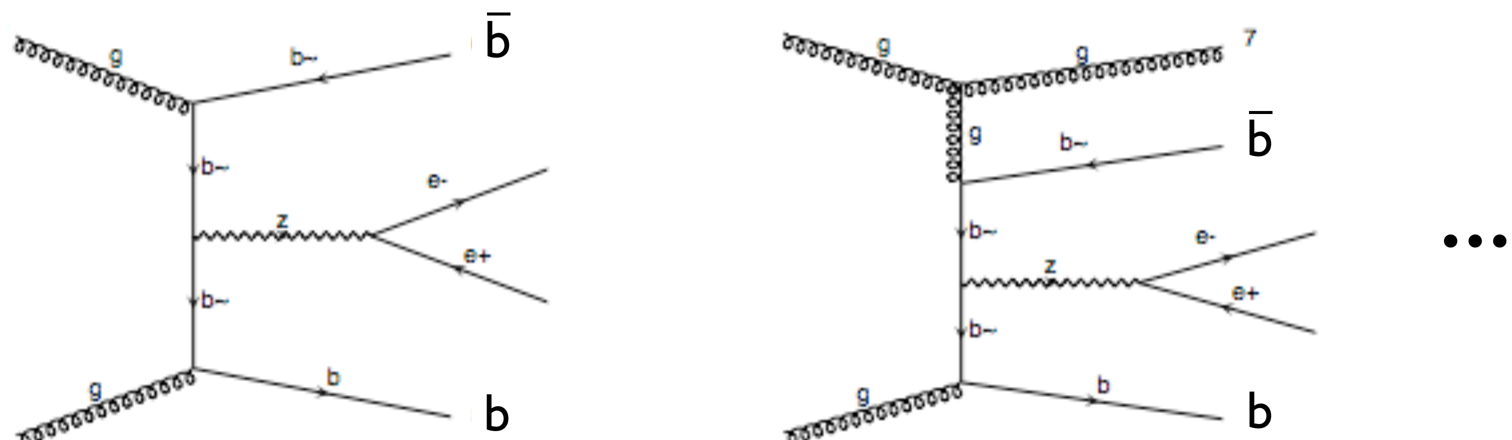
(See backup slides for pedagogic discussion)

Matching with b-quarks

- When matching with b-quarks, two options:
  1. b as any parton (5-flavor scheme)
     ➡ No special treatment needed, just use regular matching



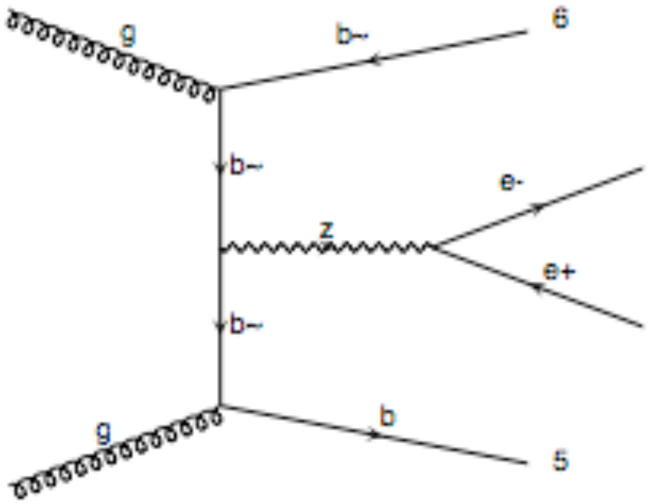
  2. b as massive final-state particle only (4-flavor scheme)



Matching with b-quarks

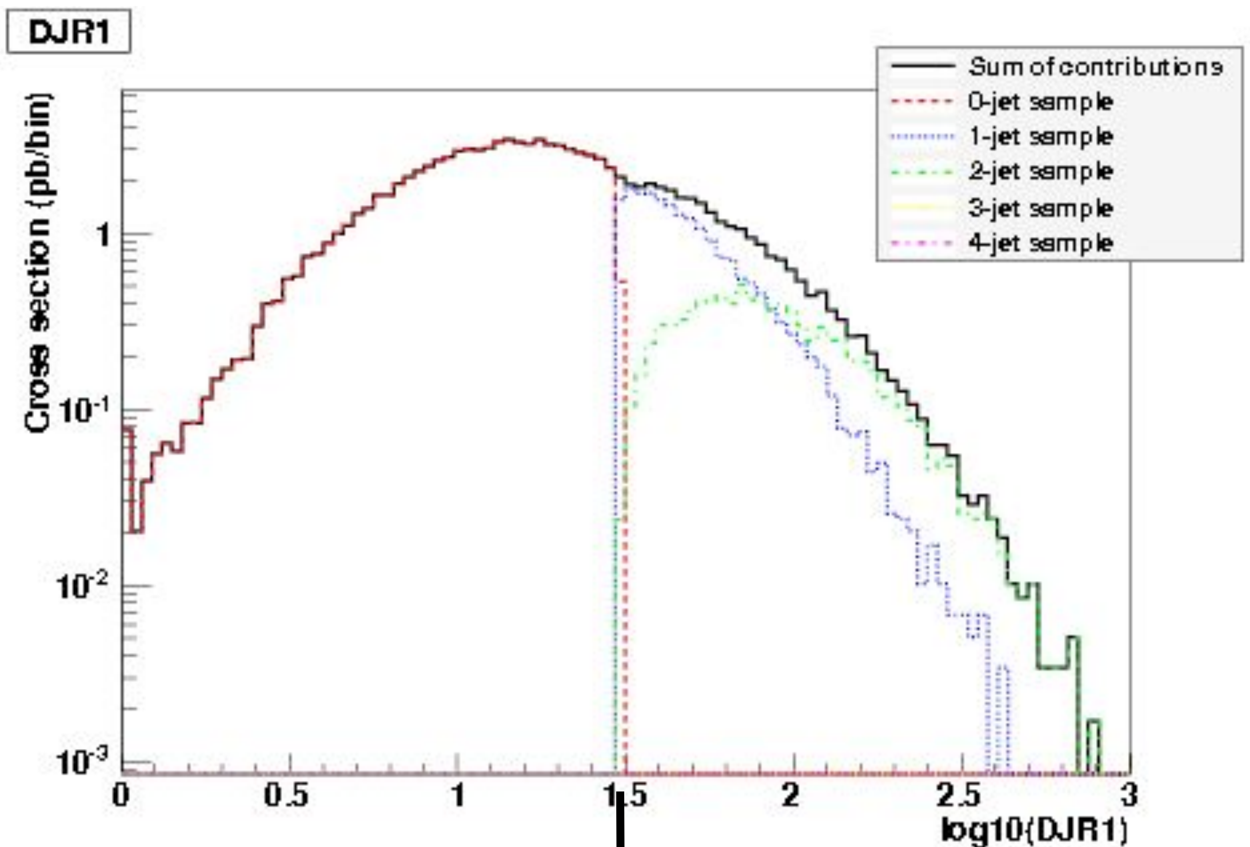
- In 5-flavor matching, just need to pick out events with b:s from all-inclusive sample
  - ➡ Below Q^{match}, b's given by shower $gb\bar{b}$ splittings
- In 4-flavor matching, no cut on b:s (allowing, 0, 1 or 2 energetic b's in event)
  - ➡ Need to remove b's from Sudakov treatment (Pythia clustering), but make sure to veto too-hard FSR from b
  - ➡ Question: How to deal with alpha_s reweighting of b vertices and factorization scales?

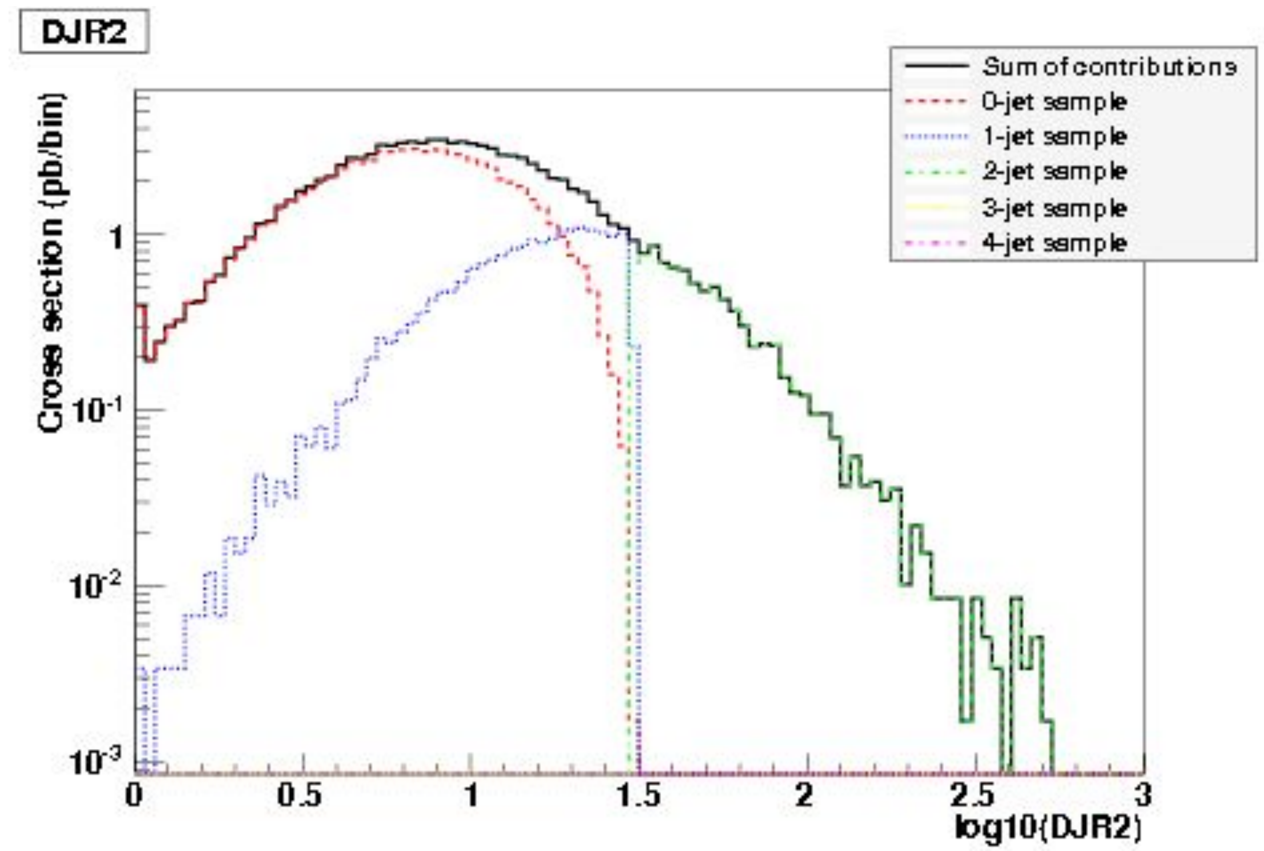
Matching with b-quarks



- If this configuration given by shower, scale for the b and b vertices would be given by m_{Tb} and $m_{T\bar{b}}$
- Shower prescription works well in 5-flavor matching, so let's use the same also for the 4-flavor scheme
- Special difficulty: Factorization scale. In principle, parton line stops at the $gb\bar{b}$ vertex, however, Z sets central scale for process
  - ➡ Use geometric average $m_{TZ}m_{Tb}$ for factorization scale
  - ➡ Gives smooth matching to Pythia PS

Matching with b-quarks





Shower | ME

➡ Gives smooth matching to Pythia PS

Summary

- MadGraph 5 is a heavily used matrix element generator and MC simulator for both SM and BSM
- Jet matching in SM and any BSM model
- News include 4-flavor b and VBF matching, new gauges and mass schemes, even more BSM functionality
- Biggest news (this month!): aMC@NLO
- Lots of news upcoming in the next few months, including: MadDecay, MadSpin, MadDM, Pythia 8 matching, automatic systematics bands, fast multipartons, ...
- Keep updated at http://launchpad.net/madgraph5!

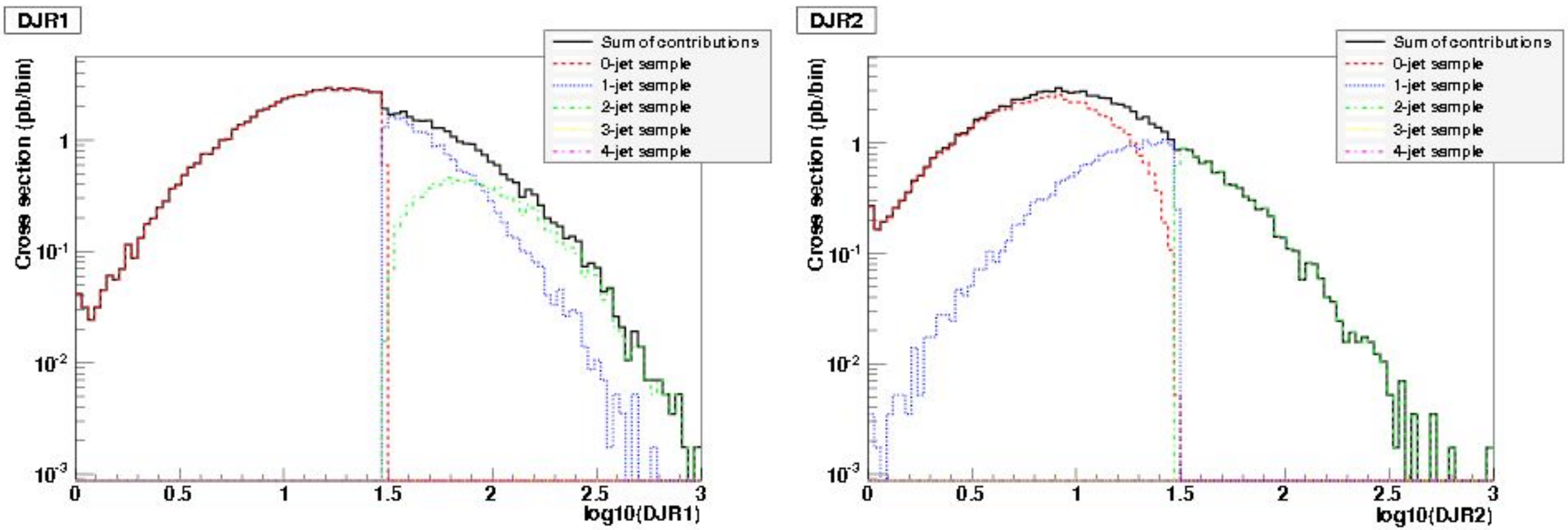
Backup slides

Decay chains

- Decay chains retain **full matrix element** for the diagrams compatible with the decay
- Full spin correlations (within and between decays)
- Full width effects
- However, no interference with non-resonant diagrams
  - ➡ Description only valid "near" pole mass
  - ➡ Cutoff at $|m \pm n\Gamma|$ where n is set in run_card.

B Matching

- bb~ W matching with scale mt(W)

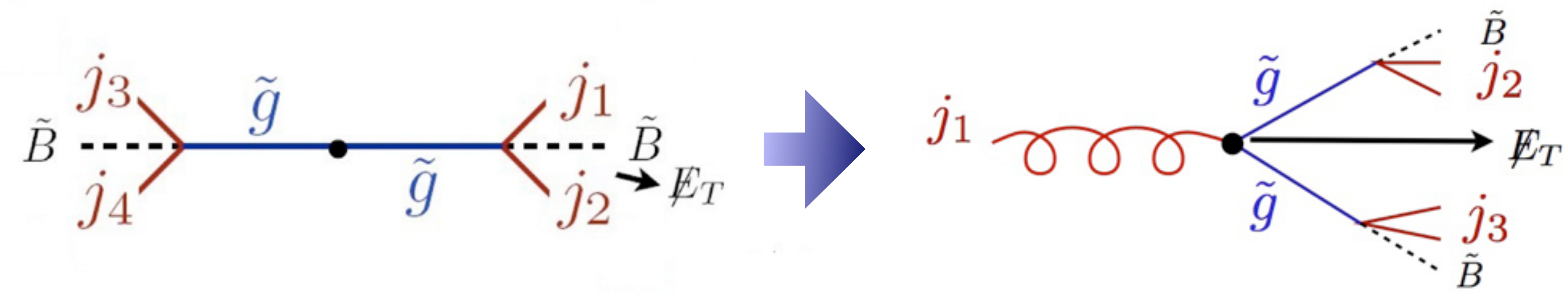


- Gives NOT smooth matching to Pythia PS

Matching in New Physics production

J.A., de Visscher, Maltoni [arXiv:0810.5350]

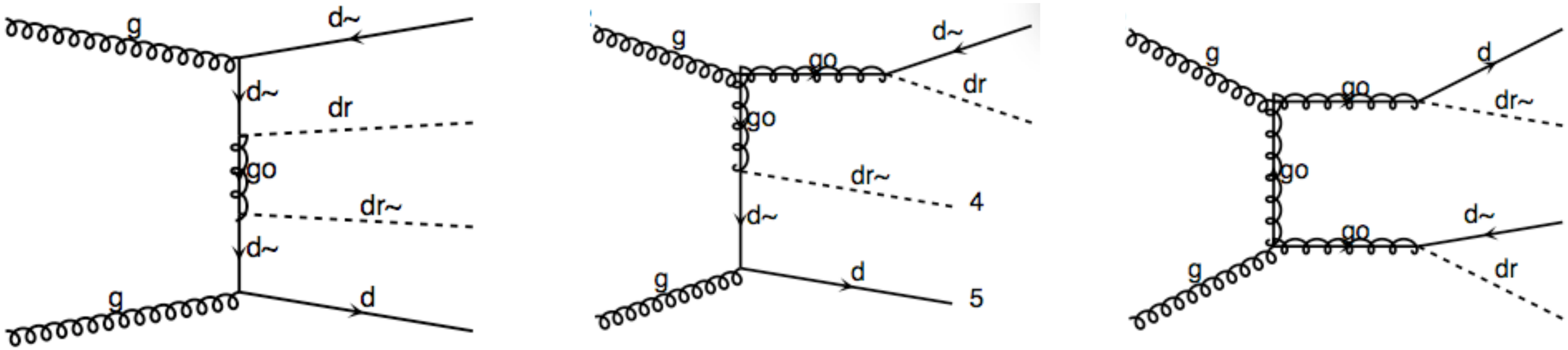
- Matching necessary also in NP production whenever QCD radiation jets are important
  - ➡ When small mass differences in hadronic decays make decay jets softer than radiated jets
  - ➡ When recoil of NP system against initial state radiation is important



- Special difficulty:
  Double-counting due to onshell decays to jets

Double counting of decays

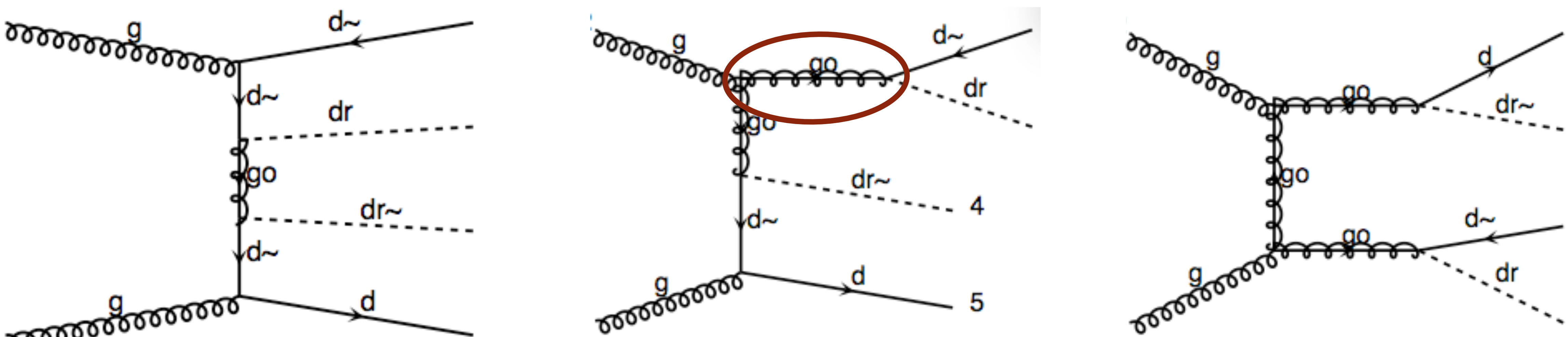
- Special difficulty in e.g. SUSY matching:
  Double counting due to on-shell decays to jets!



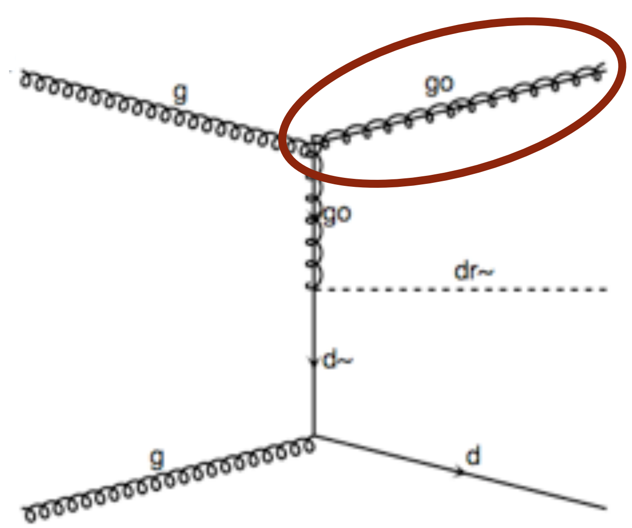
Decays double-counted with on-shell gluino production and subsequent decay

Double counting of decays

- Special difficulty in e.g. SUSY matching:
  Double counting due to on-shell decays to jets!

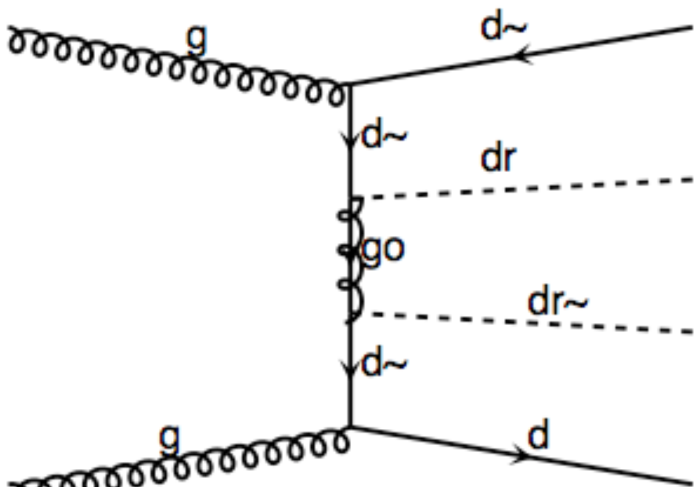


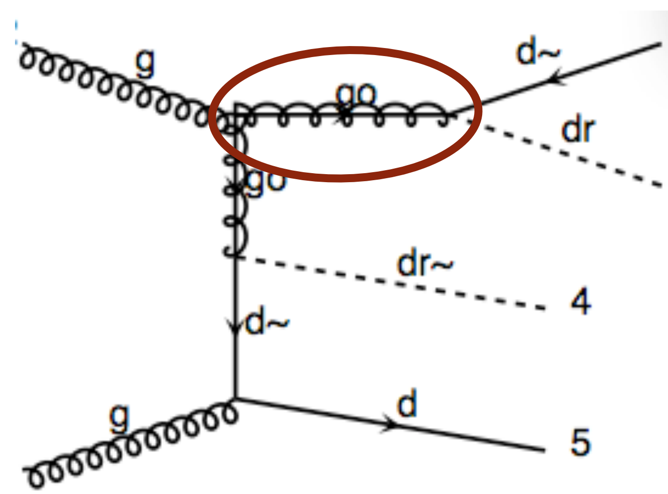
Decays double-counted with on-shell gluino production and subsequent decay

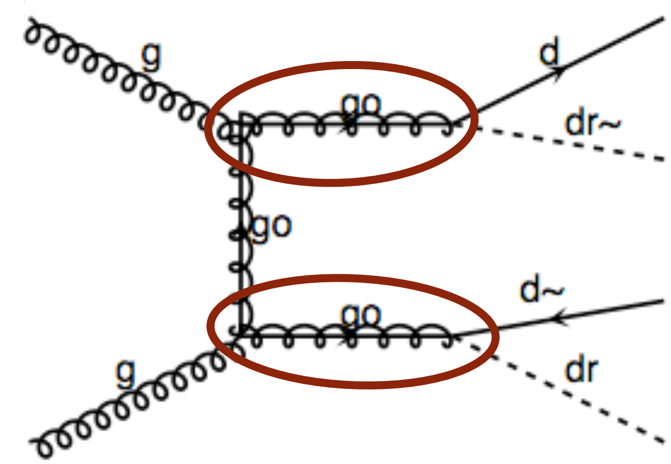


Double counting of decays

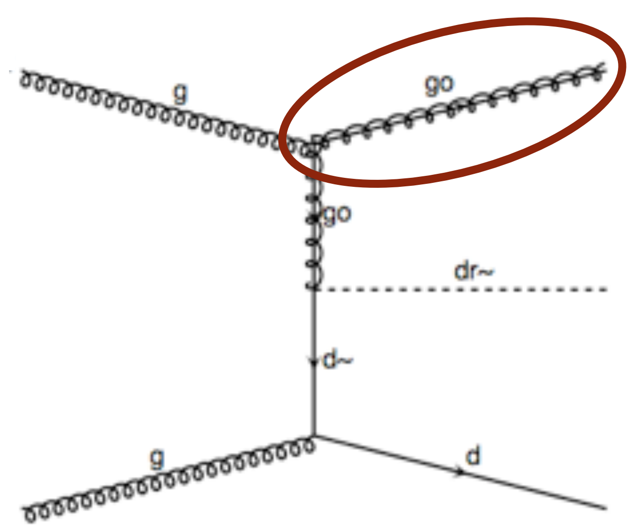
- Special difficulty in e.g. SUSY matching:
  Double counting due to on-shell decays to jets!

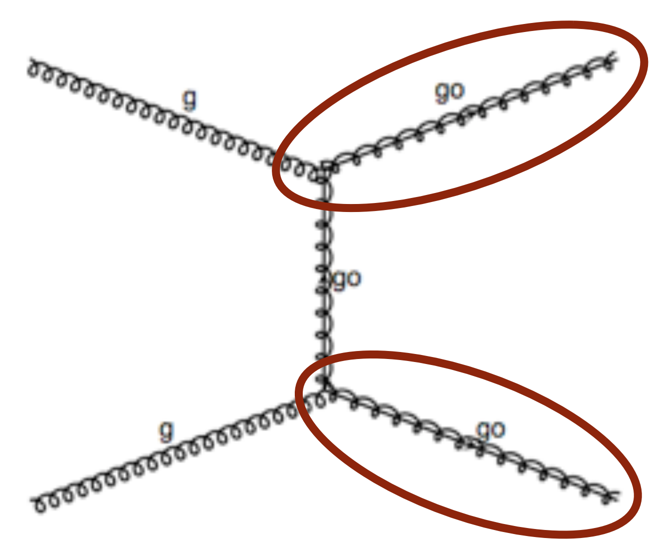






Decays double-counted with on-shell gluino production and subsequent decay





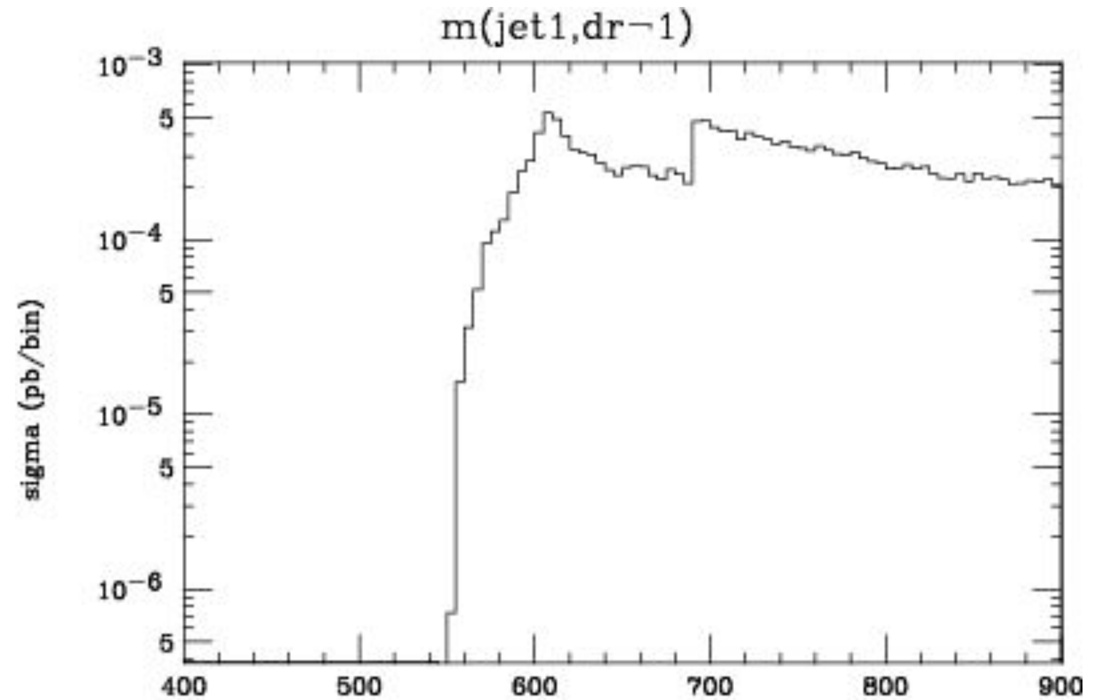
Double counting of decays

- This has been solved in MadGraph 5 by the "$" 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`).

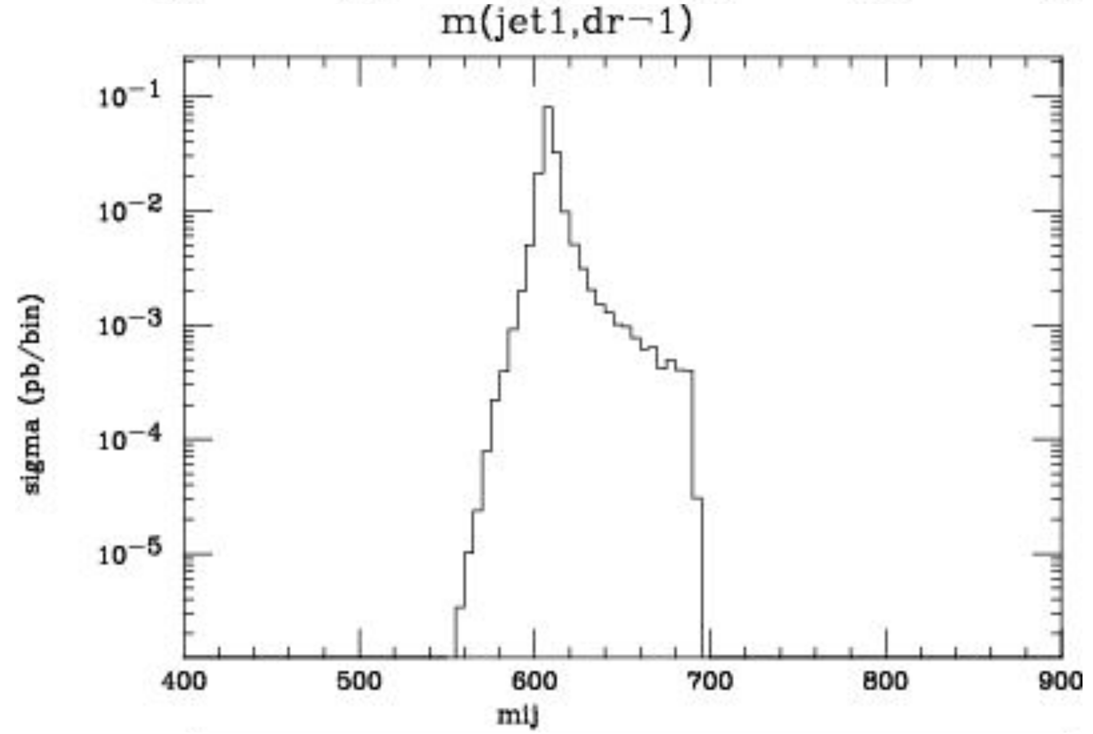
Double counting of decays

Invariant mass distributions
of d_r squark and d quark

p p > dr dr~ d $ go



p p > dr go, go > dr~ d



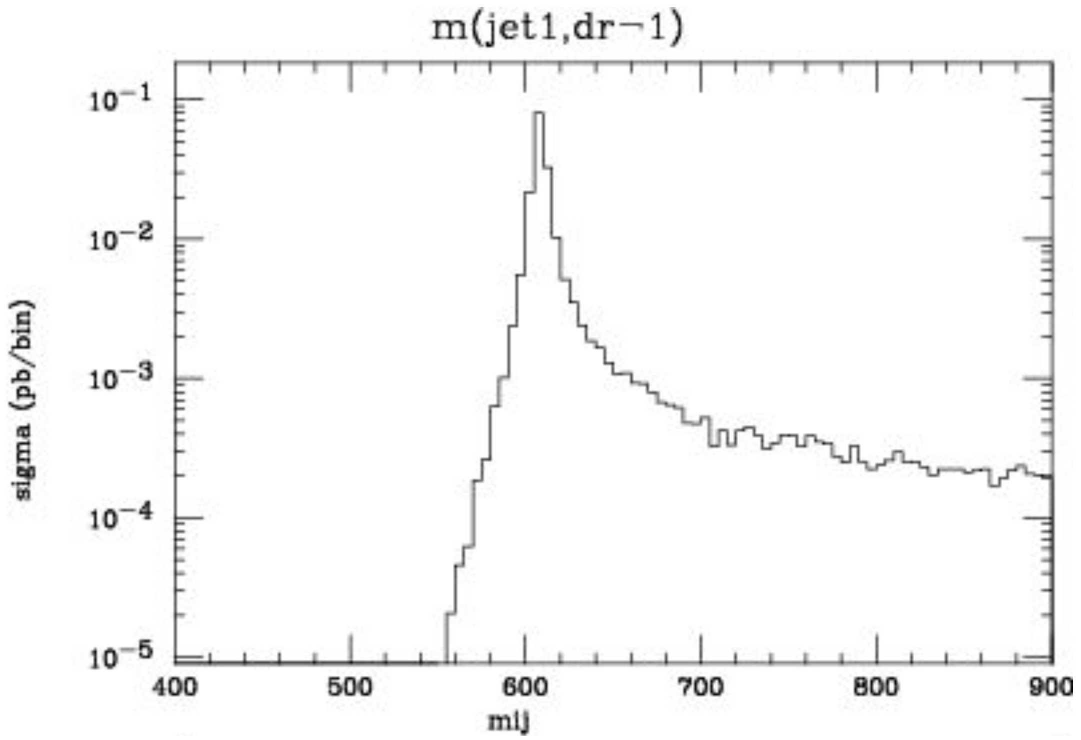
Double counting of decays

Invariant mass distributions
of d_r squark and d quark

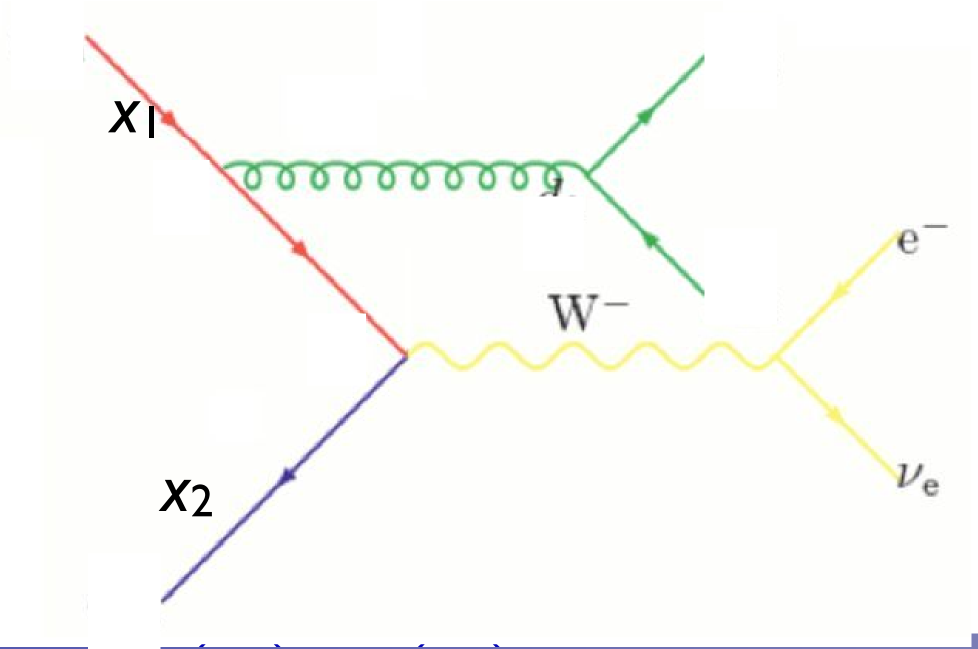
p p > dr dr~ d $ go

+

p p > dr go, go > dr~ d

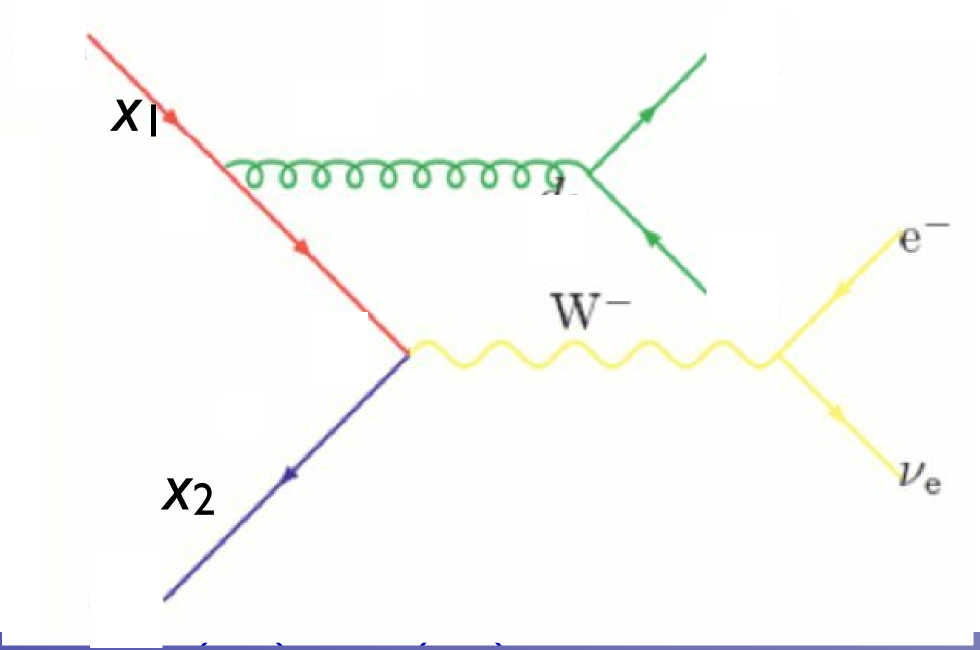


Matching for initial state radiation



Matching for initial state radiation

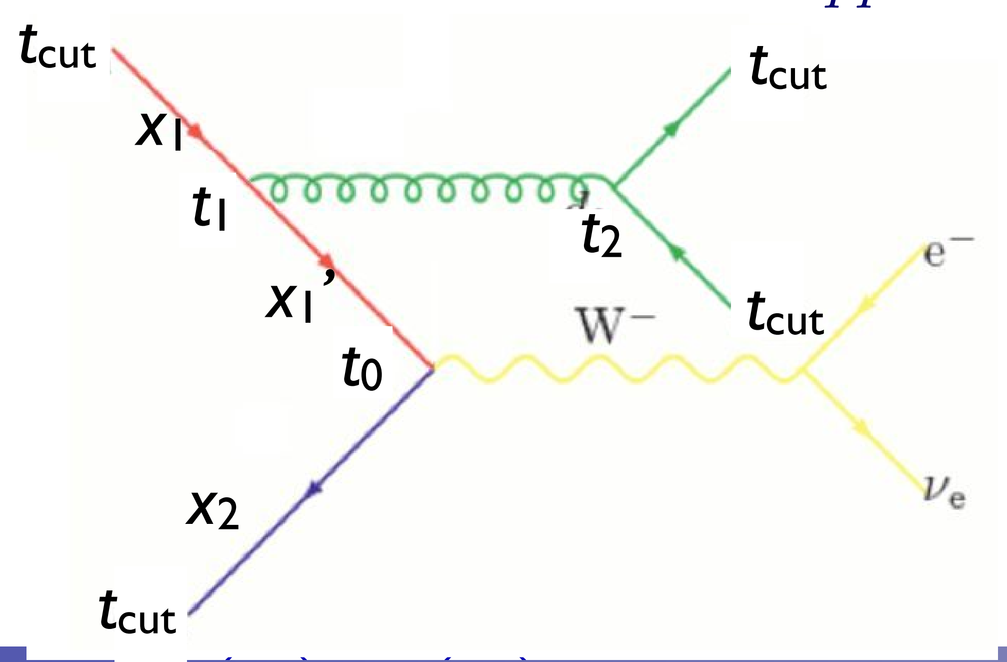
- Look at what Parton Shower gives us



Matching for initial state radiation

- Look at what Parton Shower gives us
- Modify ME to ensure smooth matching near Q^{match}

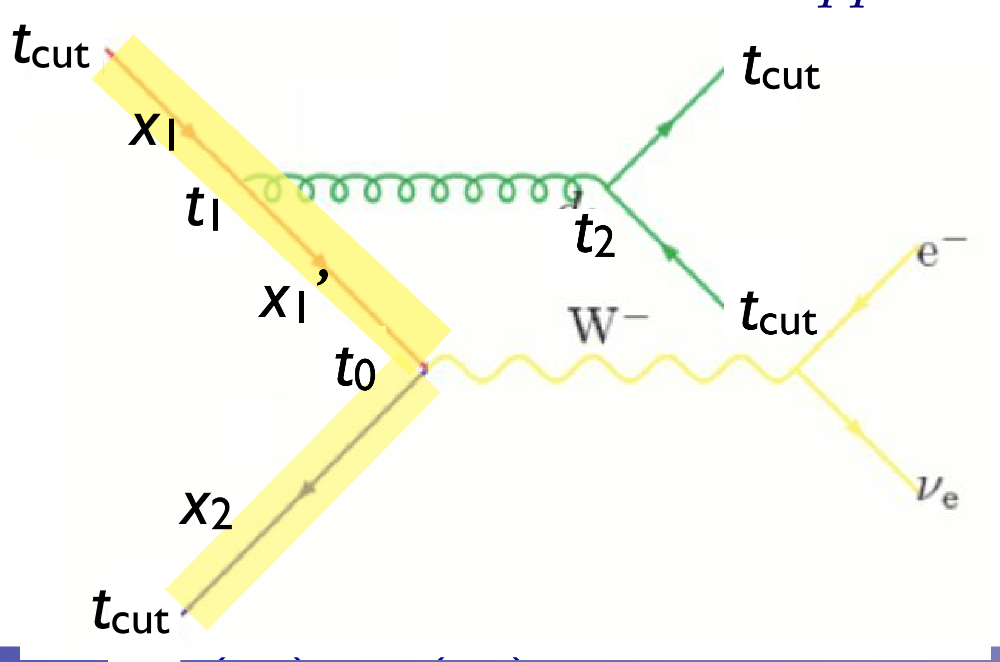
$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} \frac{P_{gq}(z)}{z} \frac{f_q(x_1, t_1)}{f_q(x_1', t_1)} \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z') \times \hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s}, \ldots) f_q(x_1', t_0) f_{\bar{q}}(x_2,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



Matching for initial state radiation

- Look at what Parton Shower gives us
- Modify ME to ensure smooth matching near Q^{match}

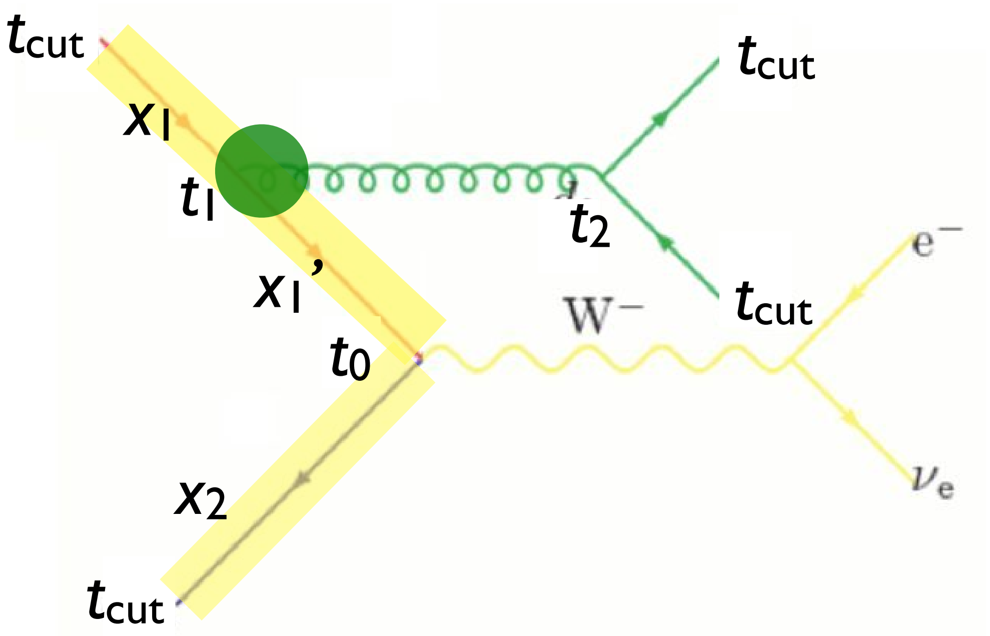
$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} \frac{P_{gq}(z)}{z} \frac{f_q(x_1, t_1)}{f_q(x'_1, t_1)} \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z') \times \hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s}, \ldots) f_q(x'_1, t_0) f_{\bar{q}}(x_2,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



Matching for initial state radiation

- Look at what Parton Shower gives us
- Modify ME to ensure smooth matching near Q^{match}

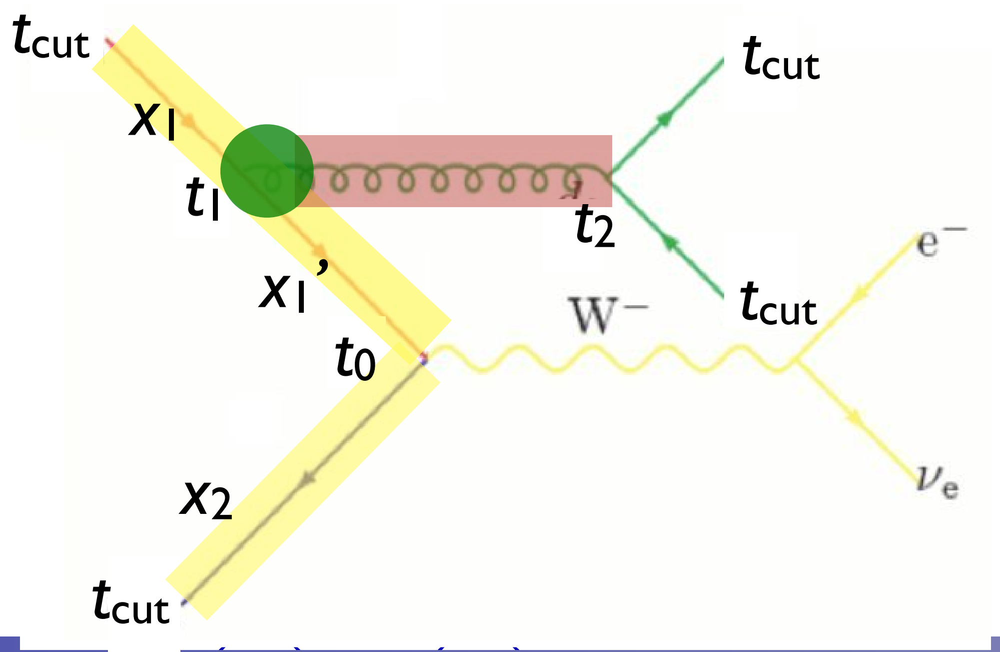
$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1, t_1)}{f_q(x'_1, t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z') \times \hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s}, \ldots) f_q(x'_1, t_0) f_{\bar{q}}(x_2,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



Matching for initial state radiation

- Look at what Parton Shower gives us
- Modify ME to ensure smooth matching near Q^{match}

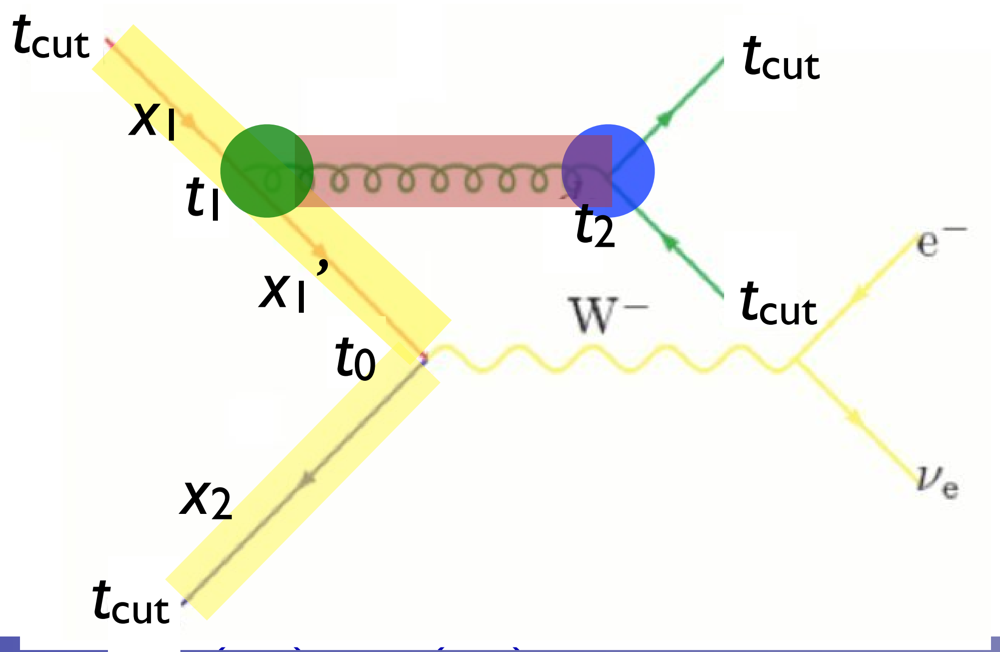
$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1, t_1)}{f_q(x'_1, t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$
$$\times \hat\sigma_{q\bar q\to e\nu}(\hat s, \ldots) f_q(x'_1, t_0) f_{\bar q}(x_2,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



Matching for initial state radiation

- Look at what Parton Shower gives us
- Modify ME to ensure smooth matching near Q^{match}

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

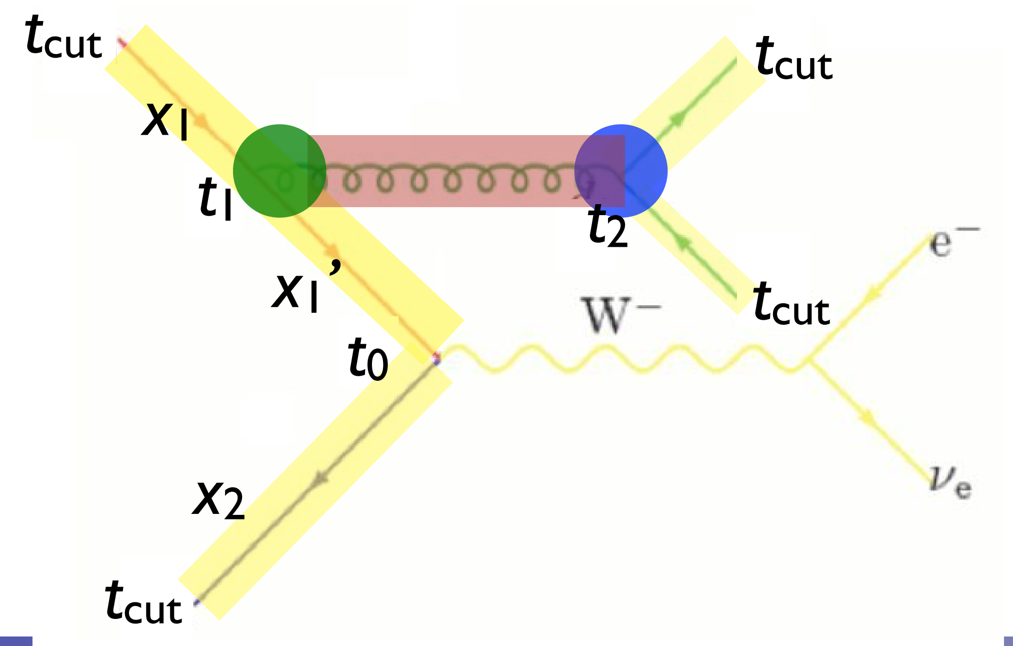


Matching for initial state radiation

- Look at what Parton Shower gives us
- Modify ME to ensure smooth matching near Q^{match}

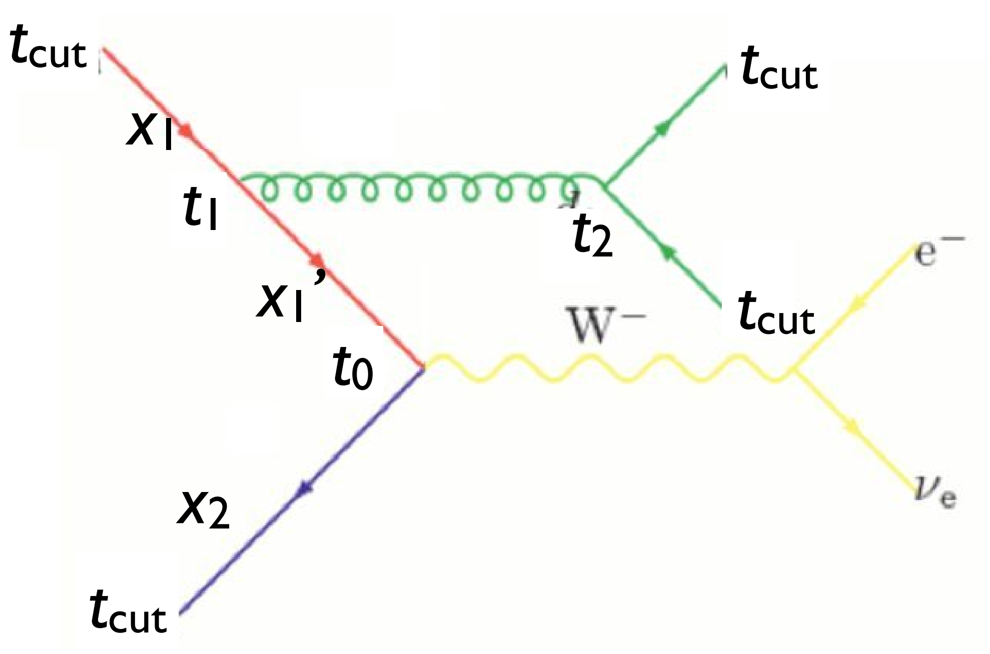
$$\mathcal{P} = (\Delta_{Iq}(t_{\text{cut}}, t_0))^2 \Delta_g(t_2, t_1) (\Delta_q(t_{\text{cut}}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} \frac{P_{gq}(z)}{z} \frac{f_q(x_1, t_1)}{f_q(x'_1, t_1)} \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$

$$\times \hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s}, \ldots) f_q(x'_1, t_0) f_{\bar{q}}(x_2, t_0)$$



Matching for initial state radiation

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

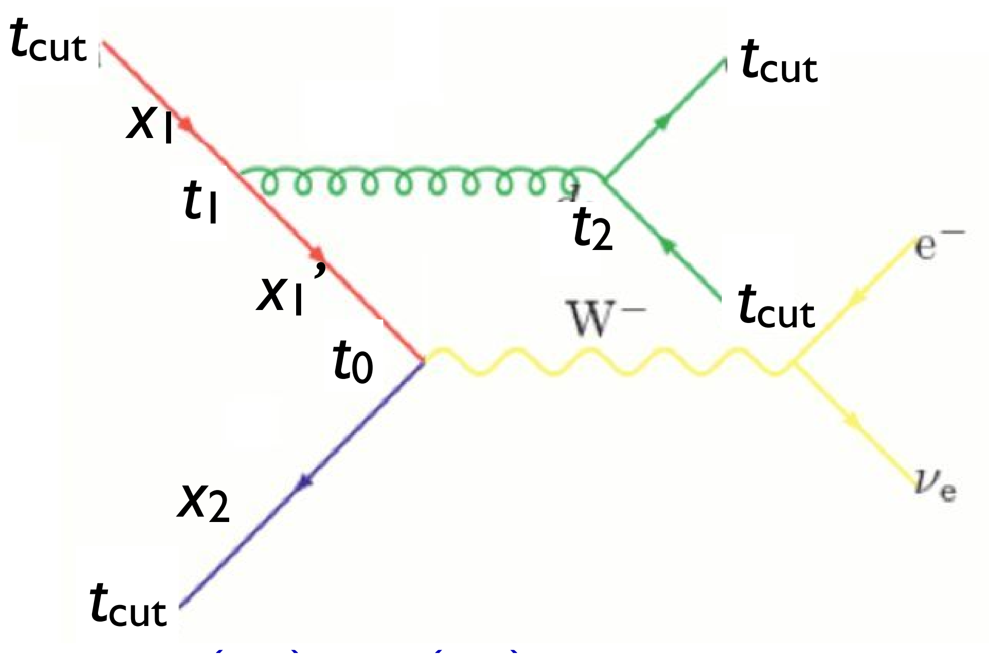


Matching for initial state radiation

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

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

ME with α_s evaluated at the scale of each splitting



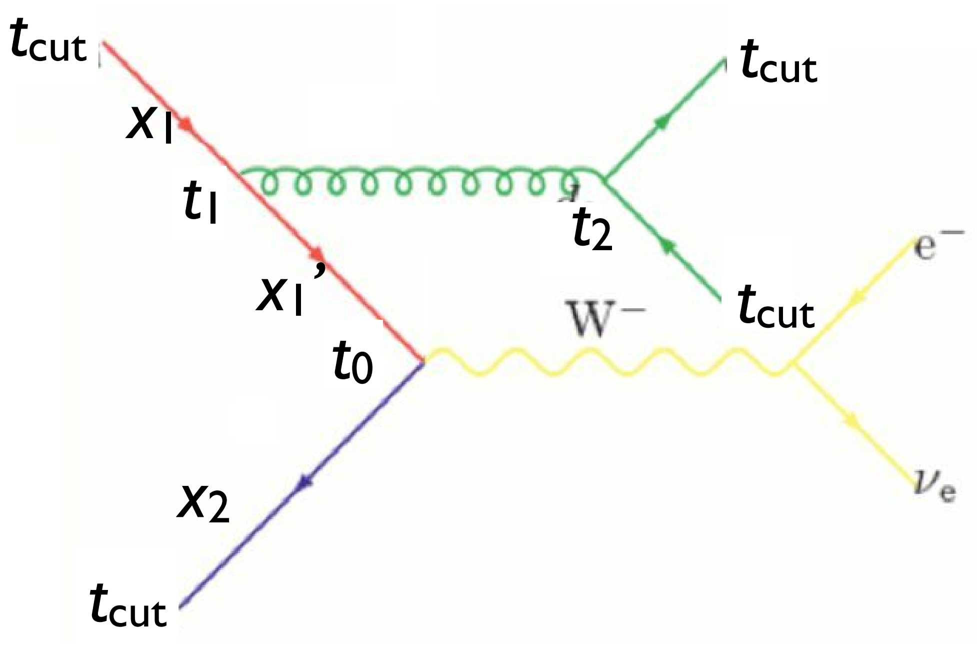
Matching for initial state radiation

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

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

ME with α_s evaluated at the scale of each splitting

PDF reweighting



Matching for initial state radiation

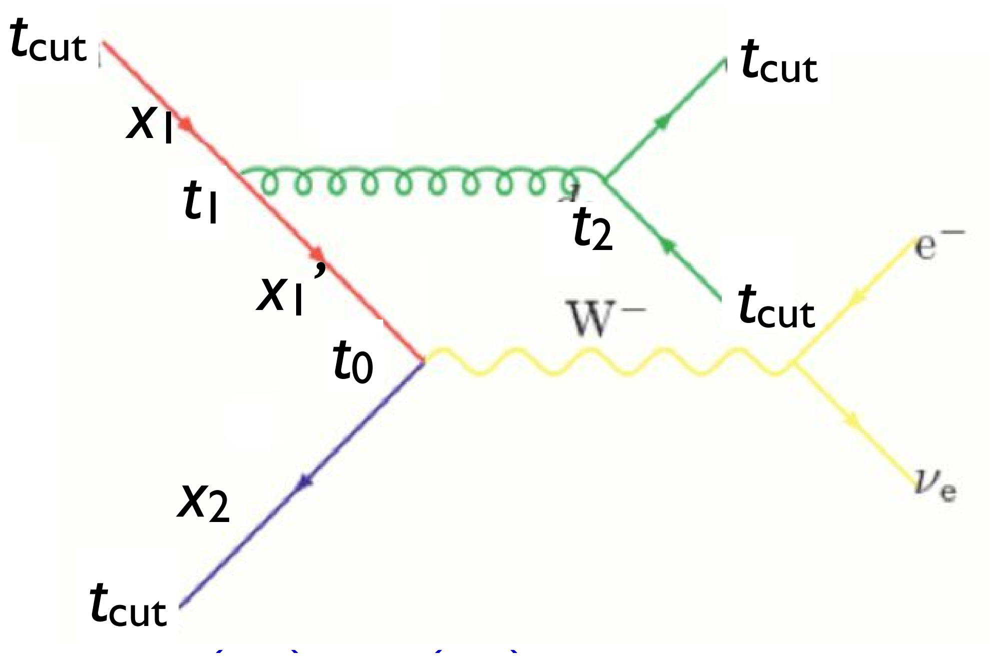
$$(\Delta_{Iq}(t_{\rm cut},t_0))^2\Delta_g(t_2,t_1)(\Delta_q(t_{\rm cut},t_2))^2\frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1,t_1)}{f_q(x_1',t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$

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

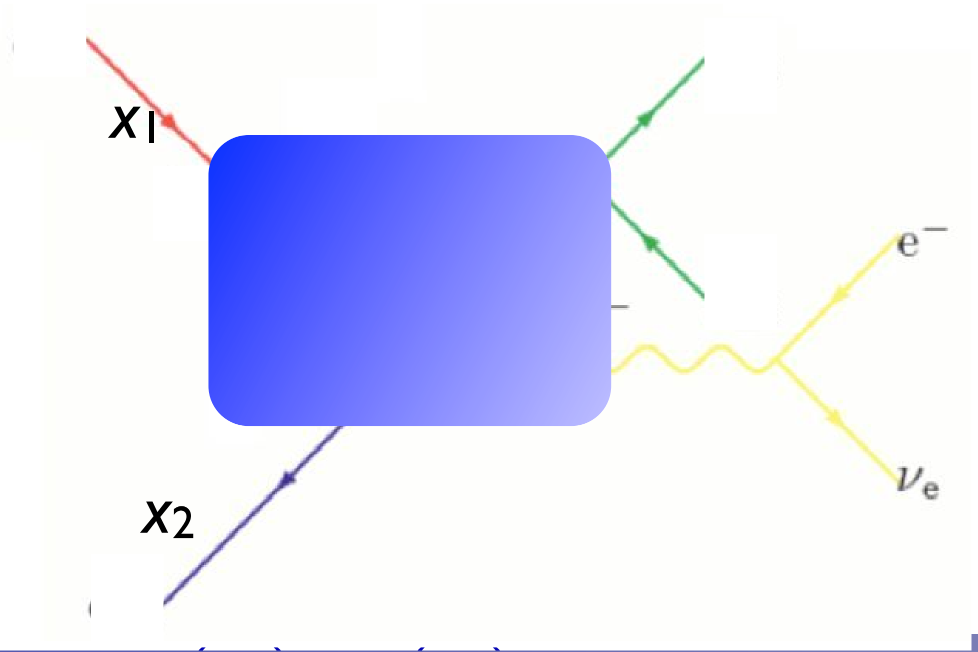
ME with α_s evaluated at the scale of each splitting

PDF reweighting

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

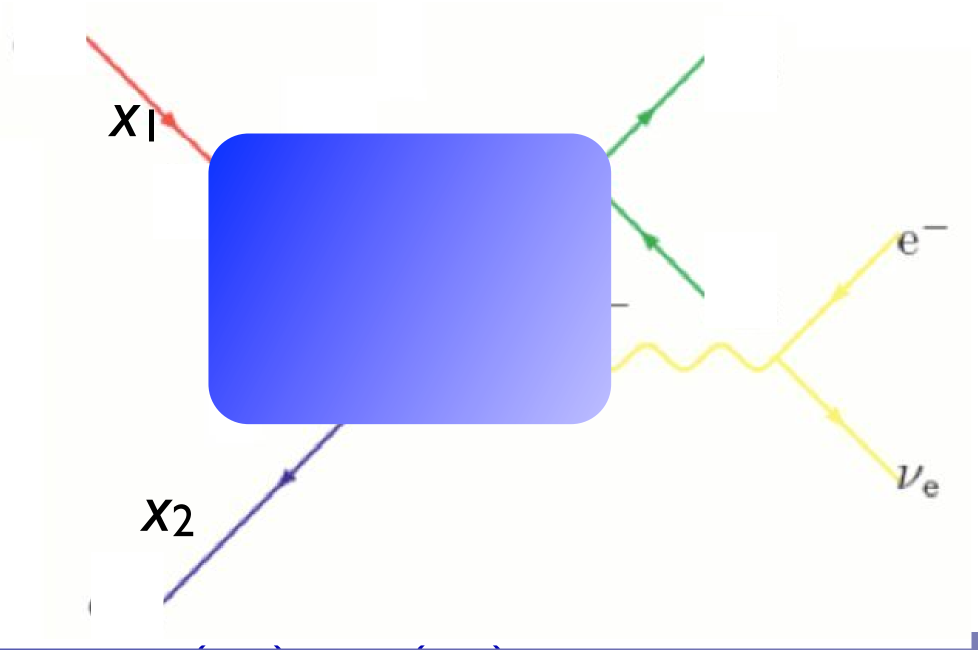


Matching for initial state radiation



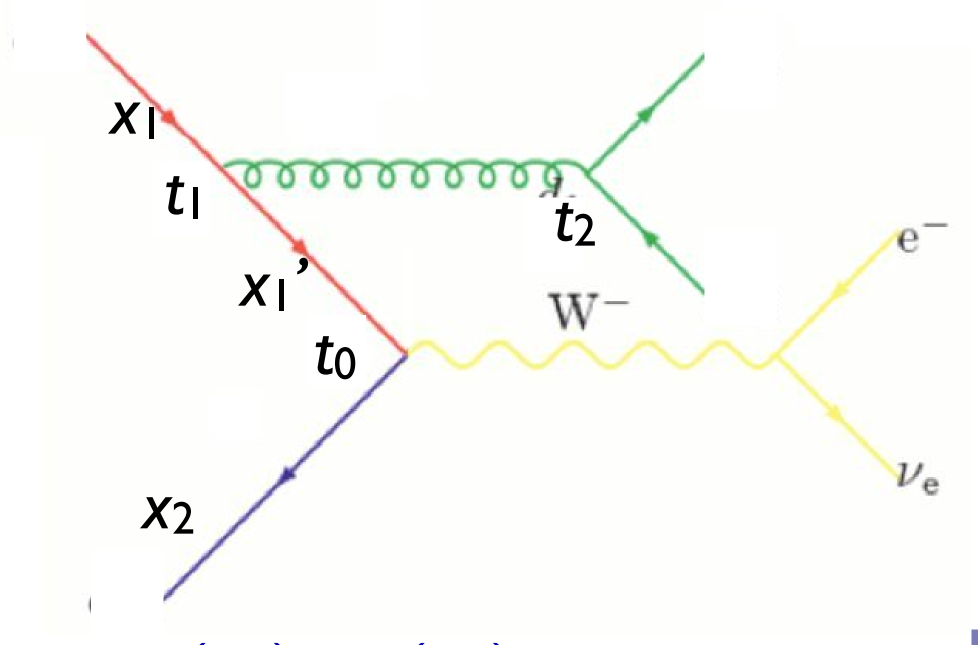
Matching for initial state radiation

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



Matching for initial state radiation

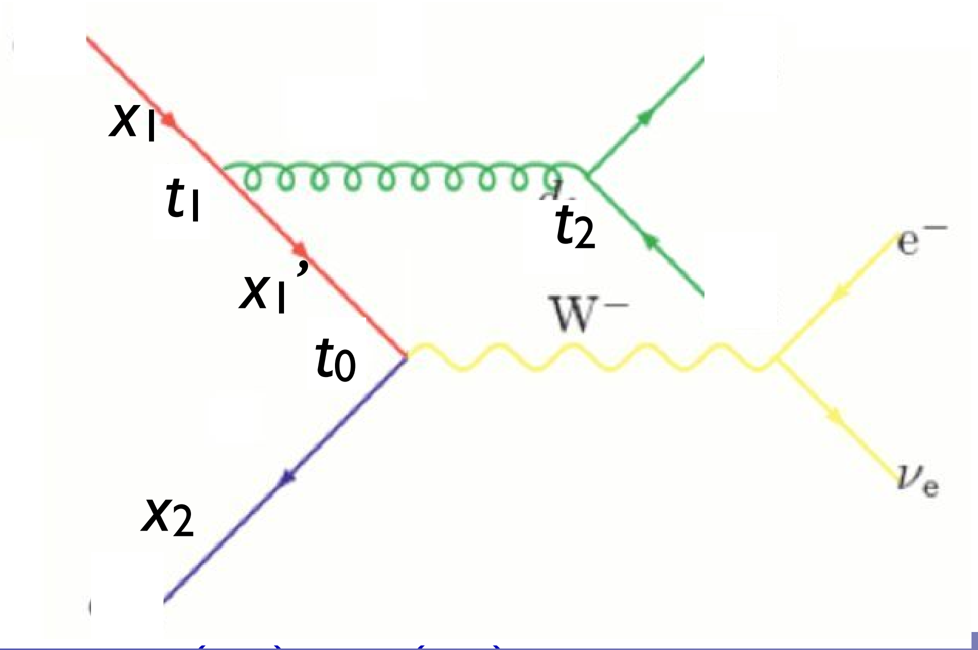
- Again, use a clustering scheme to get a parton shower history



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

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

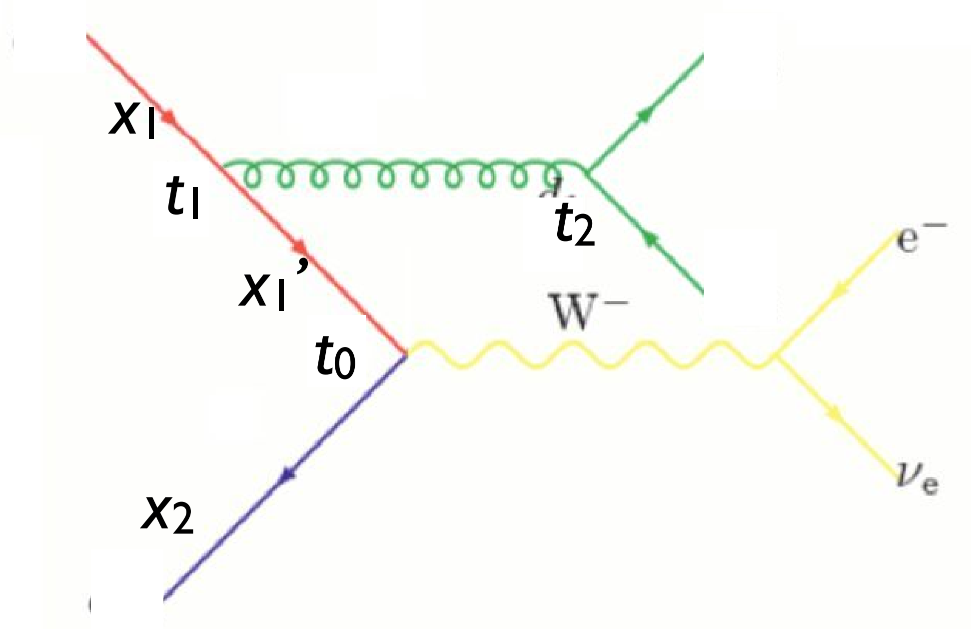


Matching for initial state radiation

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

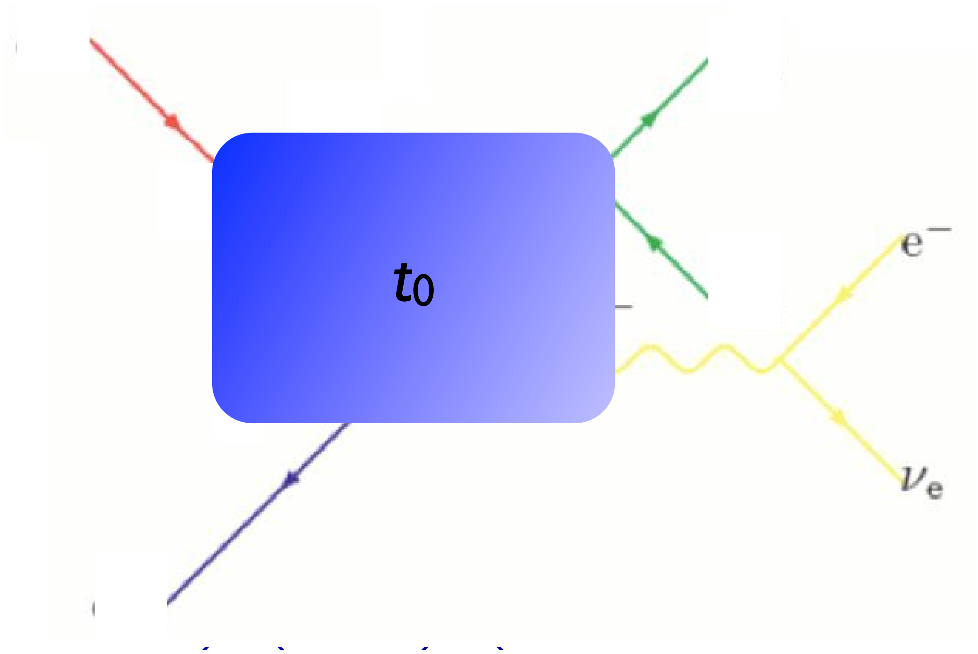
$$|\mathcal{M}|^2 \to |\mathcal{M}|^2 \frac{\alpha_s(t_1)}{\alpha_s(t_0)} \frac{\alpha_s(t_2)}{\alpha_s(t_0)} \frac{f_q(x_1', t_0)}{f_q(x_1', t_1)}$$

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



MLM matching

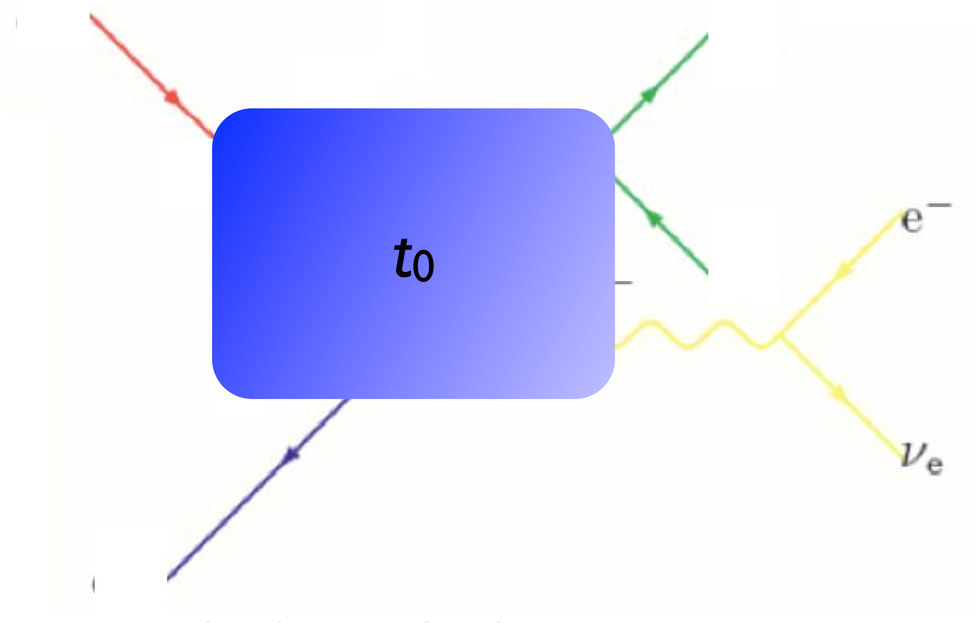
[M.L. Mangano, ~2002, 2007]
[J.A. et al 2007, 2008]



MLM matching

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

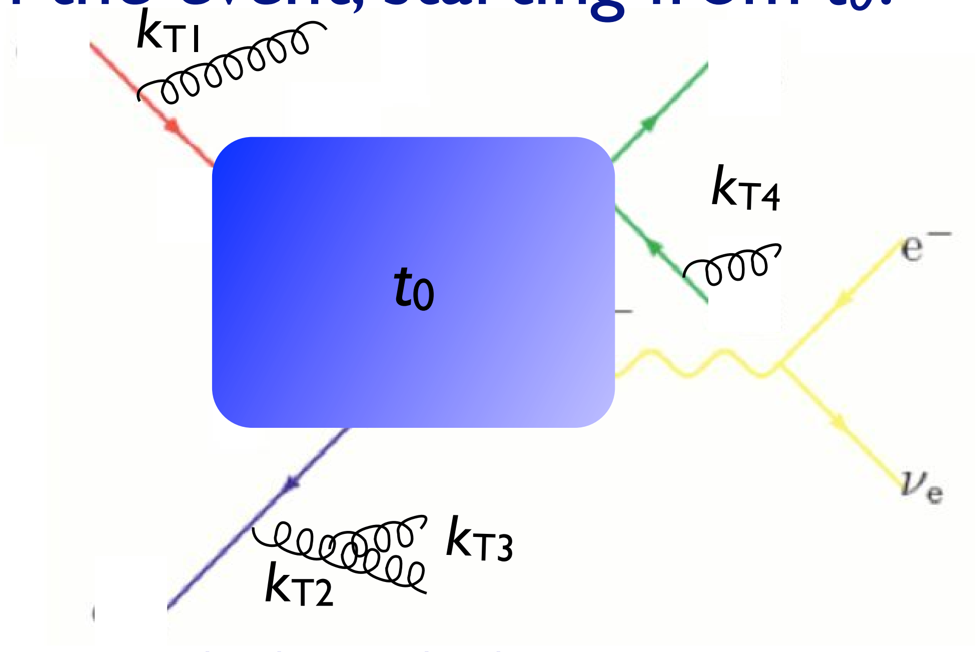
- The simplest way to do the Sudakov suppression is to run the shower on the event, starting from t_0!



MLM matching

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

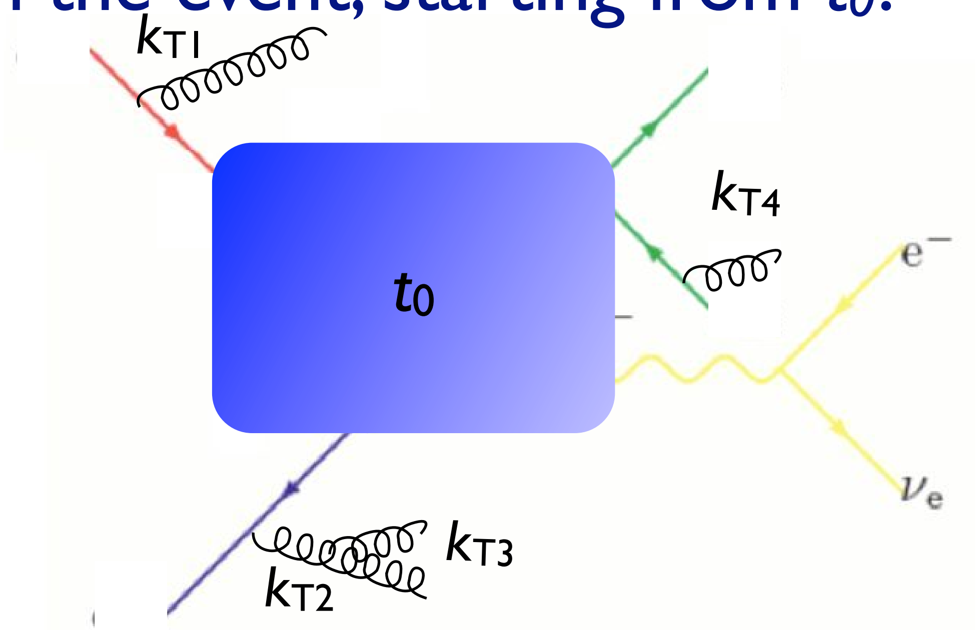
- The simplest way to do the Sudakov suppression is to run the shower on the event, starting from t_0!



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!

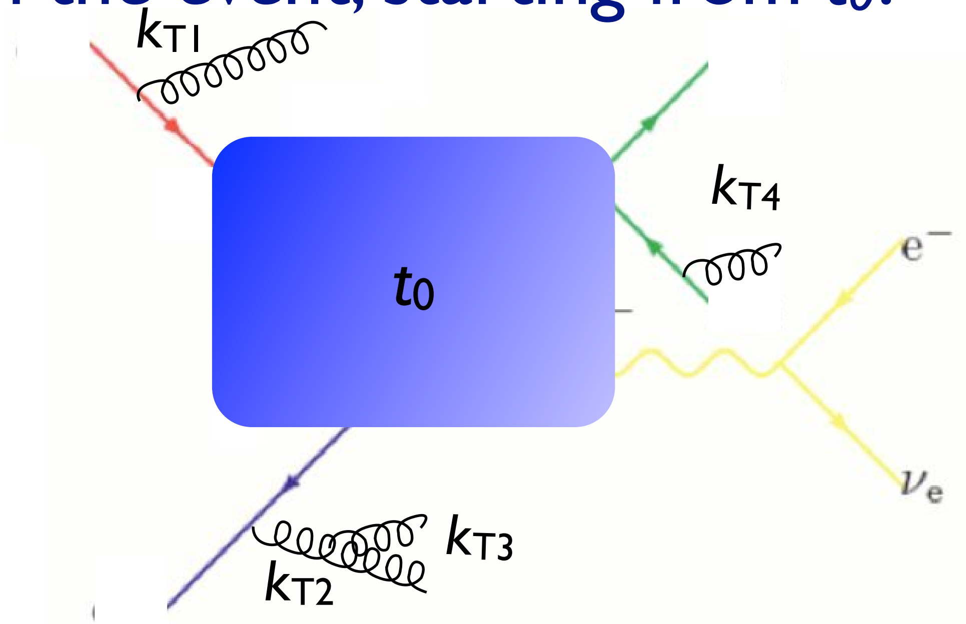


- Perform jet clustering after PS - if hardest jet $k_{T1} > t_{cut}$ or there are jets not matched to partons, reject the event

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!

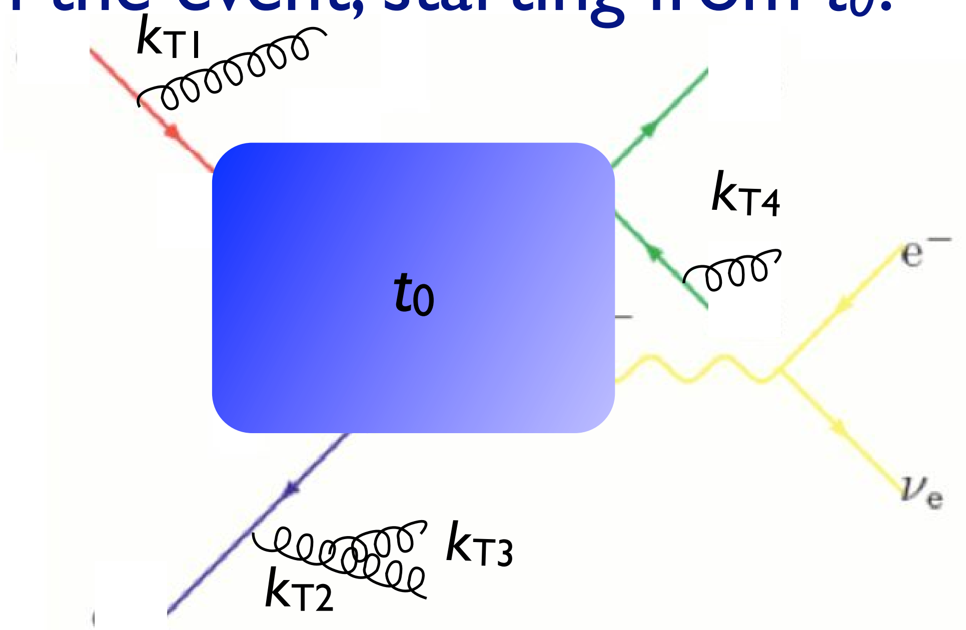


- Perform jet clustering after PS - if hardest jet $k_{T1} > t_{cut}$ or there are jets not matched to partons, reject the event
- The resulting Sudakov suppression from the procedure is
$$(\Delta_{Iq}(t_{\rm cut}, t_0))^2 (\Delta_q(t_{\rm cut}, t_0))^2$$
which turns out to be a good enough approximation 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!



- 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