MadGraph5_aMC@NLO tutorial

CERN

Marco Zaro
Hua-Sheng Shao
Benoit Hespel
Antony Martini
Ioannis Tsinikos

The people

Benoit Hespel

Antony Martini

Ioannis Tsinikos

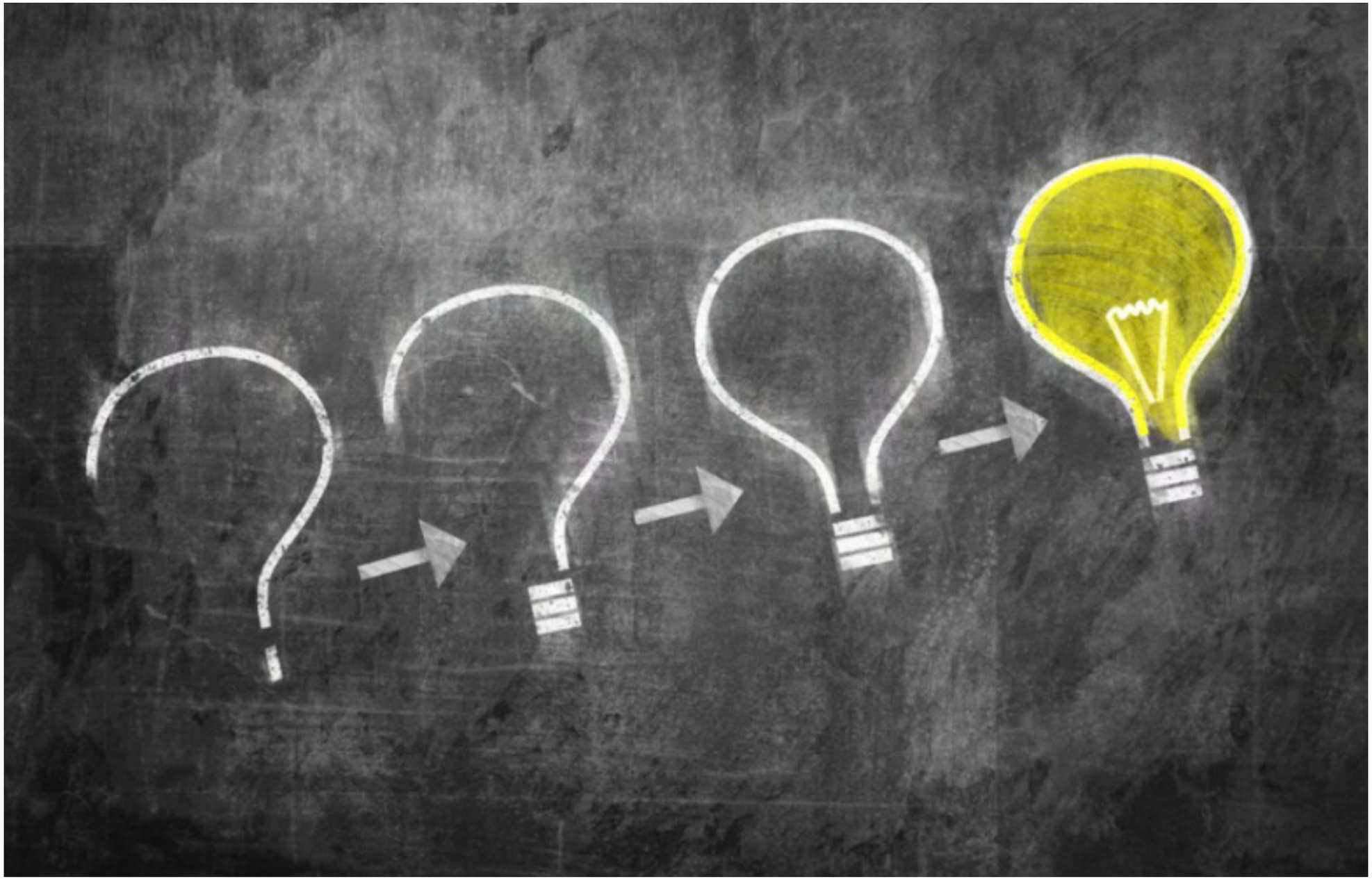
Hua-Sheng Shao

Marco Zaro

Please…

Please…

- Ask us as many questions as you like!



Please…

- Ask us as many questions as you like!
- Tell us what you think

Part 0:
Getting familiar with

What is MadGraph5_aMC@NLO?

- It is an automatic meta-code that write the code for computing the cross-section and generating events for any process at colliders
- All the details are in arXiv:1405.0301
- NLO QCD corrections can be included
- Matrix elements of different multiplicities can be combined
  - at LO (CKKW or MLM)
  - at NLO (FxFx or UNLOPS)

Software prerequisites:

- Python 2.6 or 2.7
- Fortran compiler supporting quadruple precision (needed for NLO)
  - gfortran v4.6+ OK
- Optional:
  - gnuplot
  - FastJet (FJcore is included in the tarball)
  - LHAPDF
  - Herwig++ / Pythia8 ← Pythia8.2XX may be needed in the tutorial
  - …

Where do I get it?

- On LaunchPad: https://launchpad.net/mg5amcnlo

MadGraph5_aMC@NLO Generator

marco zaro (marco-zaro) • Log Out

Overview | Code | Bugs | Blueprints | Translations | Answers

Registered 2009-09-15 by Michel Herquet

MadGraph5_aMC@NLO is a framework that aims at providing all the elements necessary for SM and BSM phenomenology, such as the computations of cross sections, the generation of hard events and their matching with event generators, and the use of a variety of tools relevant to event manipulation and analysis. Processes can be simulated to LO accuracy for any user-defined Lagrangian, and the NLO accuracy in the case of QCD corrections to SM processes. Matrix elements at the tree- and one-loop-level can also be obtained.

MadGraph5_aMC@NLO is the new version of both MadGraph5 and aMC@NLO that unifies the LO and NLO lines of development of automated tools within the MadGraph family. It therefore supersedes all the MadGraph5 1.5.x versions and all the beta versions of aMC@NLO.

The standard reference for the use of the code is: J. Alwall et al, "The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations", arXiv:1405.0301 [hep-ph]. A more complete list of references can be found here: http://amcatnlo.web.cern.ch/amcatnlo/list_refs.htm

Download:

The latest stable release can downloaded as a tar.gz package (see the right of this page), or through the Bazaar versioning system, using bzr branch lp:madgraph5

Installation:

MadGraph5_aMC@NLO needs Python version 2.6 or 2.7 ; gfortran/gcc 4.6 or higher is required for NLO calculations/simulations.

Getting started:

Run bin/mg5_aMC and type "help" to learn how to run MadGraph5_aMC@NLO using the command interface, or run the interactive quick-start tutorial by typing "tutorial". Some third-party packages can be installed using the MG5_aMC shell command "install". LO generation can also be done

Change details
Sharing
Subscribe to bug mail
Edit bug mail

Get Involved
Report a bug
Ask a question
Register a blueprint
Help translate

Configuration Progress

Configuration options

Downloads

Latest version is 2.2.0

MG5_aMC_v2....beta.tar.gz

MG5_aMC_v2.2.3.tar.gz

Where do I get it?

- On LaunchPad: https://launchpad.net/mg5amcnlo
- `tar -xzf MG5_aMC_v2.4.2.tar.gz`
- `cd MG5_aMC_v2_4_2`
- `./bin/mg5_aMC`

Let's start the tutorial

- On LaunchPad: https://launchpad.net/mg5amcnlo
- `tar -xzf MG5_aMC_v2.4.2.tar.gz`
- `cd MG5_aMC_v2_4_2`
- `./bin/mg5_aMC`
- `> tutorial`
- `> tutorial aMCatNLO`
- `> install pythia-pgs`
- `> install MadAnalysis`

Will be needed in the tutorial

Part 1:
LO event generation

Exercise 1:
Top pair production at LO

- Basic questions:
  - Generate the process (following the tutorial)
  - Which partonic subprocesses contribute?
  - How many Feynman diagrams has each subprocess?
  - Output the code
  - Compute the cross-section at the LHC (8 TeV) for m_t=170 GeV
- Extra questions:
  - Are b-quarks included in the initial state? If not, how can I include them?
  - Are diagrams with photons/z included? If not, how can I include them? How much does the cross-section change? What is that 'WEIGHTED'?
  - Recompute the $t\bar{t}$ cross-section for m_t=170, 172, 174 … 180 GeV

Steering MG5_aMC

- Once the process is generated, the parameters for the run are stored in various cards (inside the `Cards` dir):
  - `param_card` : all model parameters (masses, couplings, …)
  - `run_card` : all parameters relevant to the collider and run settings (energy, N_{events}, cuts, scales, …)
  - `pythia_card` (MadEvent) / `shower_card` (aMC) : parameters needed to shower the partonic events
  - `plot_card` (MadEvent) : which variables to plot
  - `delphes/pgs_card` : detector simulation
  - …

Exercise 1: solution

- Generate the process (following the tutorial)
  - > `generate p p > t t~`

```
INFO: Checking for minimal orders which gives processes.
INFO: Please specify coupling orders to bypass this step.
INFO: Trying coupling order WEIGHTED=2
INFO: Trying process: g g > t t~ WEIGHTED=2
INFO: Process has 3 diagrams
INFO: Trying process: u u~ > t t~ WEIGHTED=2
INFO: Process has 1 diagrams
INFO: Trying process: u c~ > t t~ WEIGHTED=2
INFO: Trying process: c u~ > t t~ WEIGHTED=2
INFO: Trying process: c c~ > t t~ WEIGHTED=2
INFO: Process has 1 diagrams
INFO: Trying process: d d~ > t t~ WEIGHTED=2
INFO: Process has 1 diagrams
INFO: Trying process: d s~ > t t~ WEIGHTED=2
INFO: Trying process: s d~ > t t~ WEIGHTED=2
INFO: Trying process: s s~ > t t~ WEIGHTED=2
INFO: Process has 1 diagrams
INFO: Process u~ u > t t~ added to mirror process u u~ > t t~
INFO: Process c~ c > t t~ added to mirror process c c~ > t t~
INFO: Process d~ d > t t~ added to mirror process d d~ > t t~
INFO: Process s~ s > t t~ added to mirror process s s~ > t t~
5 processes with 7 diagrams generated in 0.075 s
Total: 5 processes with 7 diagrams
```

Exercise 1: solution

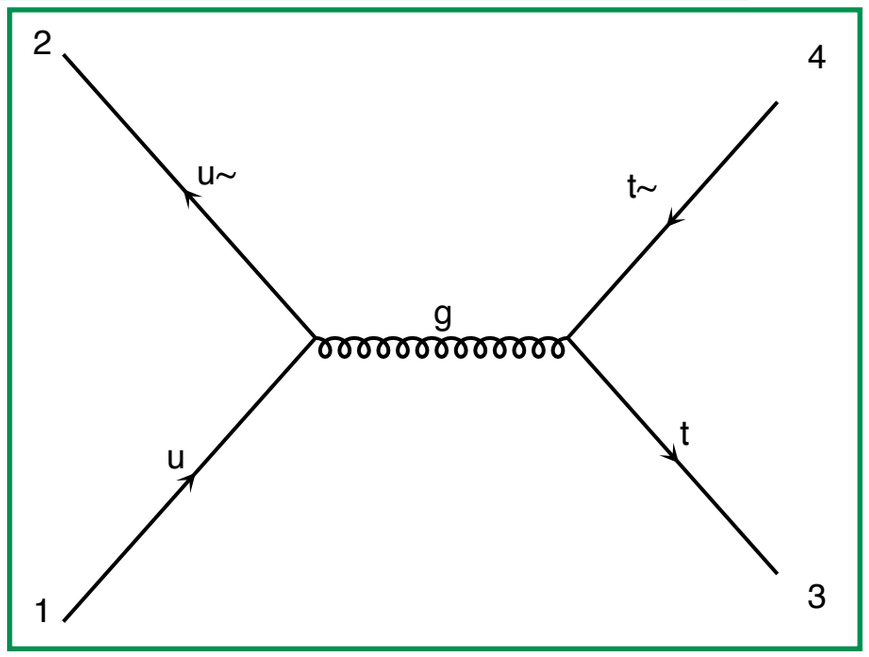
- Which partonic subprocesses contribute?
  - > `display processes`

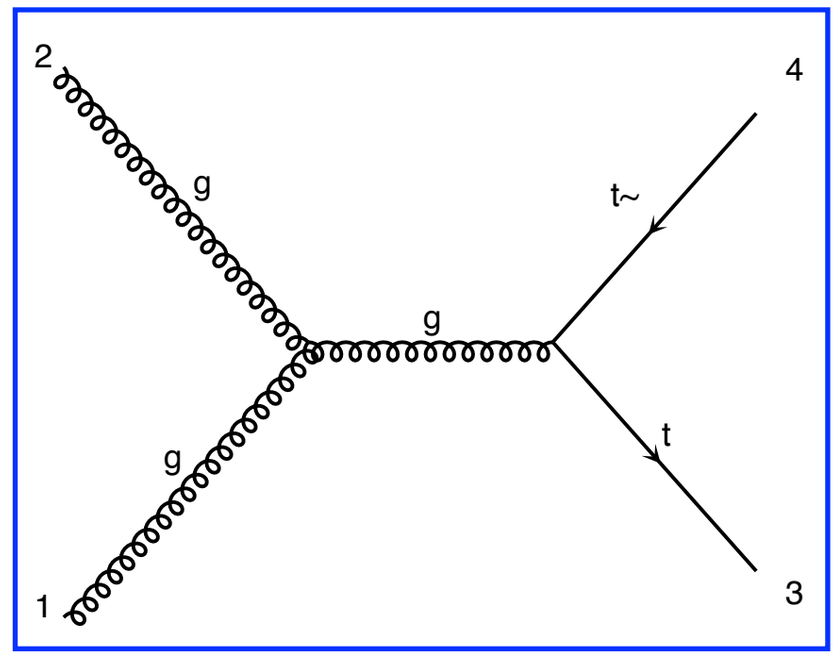
```
Process: g g > t t~ WEIGHTED=2
Process: u u~ > t t~ WEIGHTED=2
Process: c c~ > t t~ WEIGHTED=2
Process: d d~ > t t~ WEIGHTED=2
Process: s s~ > t t~ WEIGHTED=2
```

Exercise 1: solution

- Which partonic subprocesses contribute?
  - > `display processes`

```
Process: g g > t t~ WEIGHTED=2
Process: u u~ > t t~ WEIGHTED=2
Process: c c~ > t t~ WEIGHTED=2
Process: d d~ > t t~ WEIGHTED=2
Process: s s~ > t t~ WEIGHTED=2
```





Exercise 1: solution

- Which partonic subprocesses contribute?
  - `> display processes`

```
Process: g g > t t~ WEIGHTED=2
Process: u u~ > t t~ WEIGHTED=2
Process: c c~ > t t~ WEIGHTED=2
Process: d d~ > t t~ WEIGHTED=2
Process: s s~ > t t~ WEIGHTED=2
```

QCD master formula:

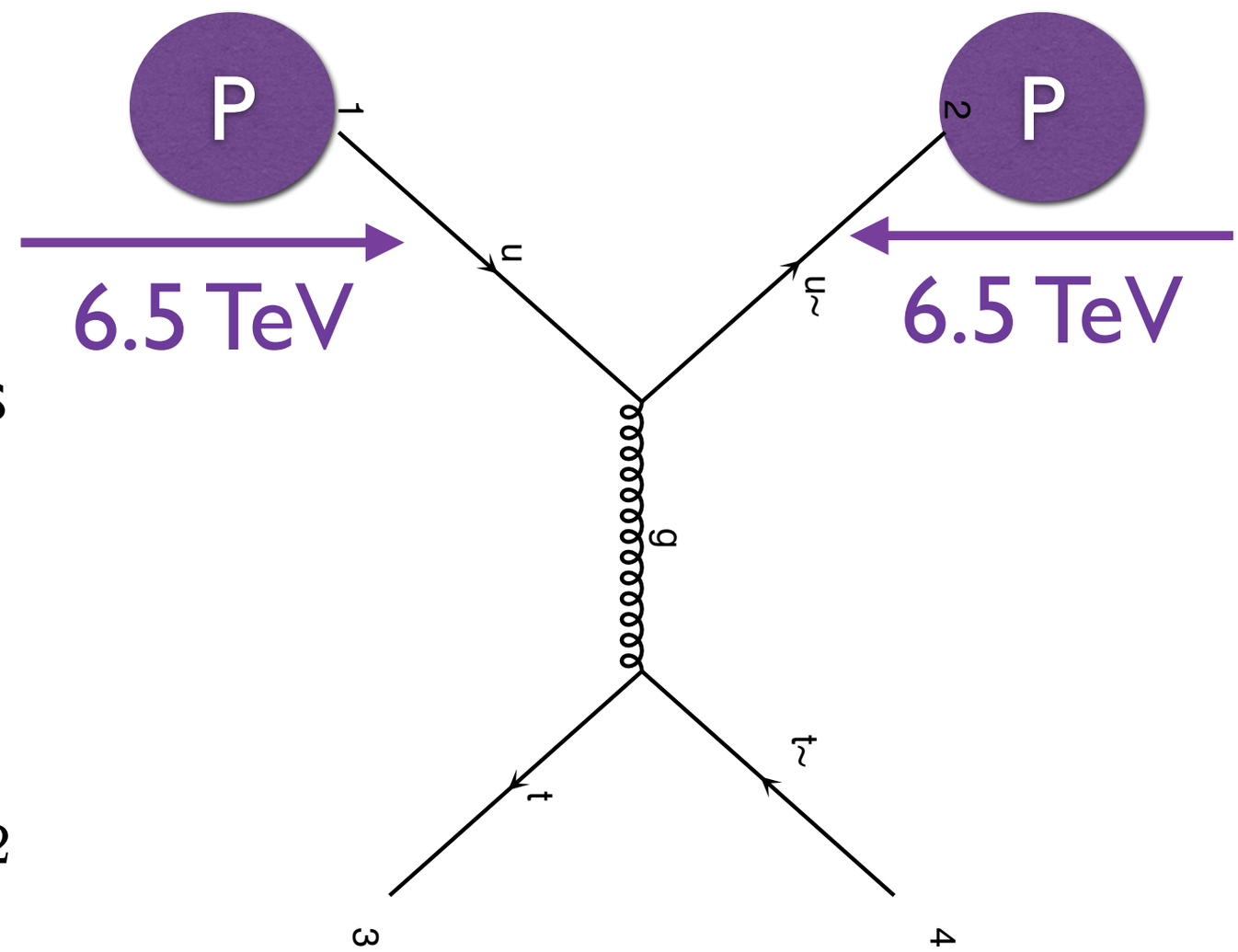
$$\sigma(pp \to t\bar{t}) = \sum_{ab} \int dx_1 dx_2 f_a(x_1, \mu_F) f_b(x_2, \mu_F) \times \hat{\sigma}(ab \to t\bar{t})$$

What does it mean?

$$\sigma(pp \to t\bar{t}) = \sum_{ab} \int dx_1 dx_2 f_a(x_1, \mu_F) f_b(x_2, \mu_F) \times \hat{\sigma}(ab \to t\bar{t})$$

- What is the probability to find parton a inside the proton with momentum fraction x? $f_a(x)$
- μ_F is a scale which separates low energy from high energy dynamics
- The partonic scattering occurs at a reduced energy:

$$\hat{s} = x_1 x_2 S = x_1 x_2 (13\mathrm{TeV})^2$$



Exercise 1: solution

- How many Feynman diagrams has each subprocess?
  - > `display diagrams`

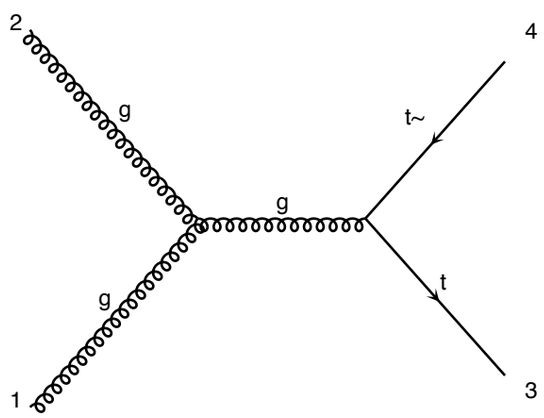


diagram 1 QCD=2, QED=0

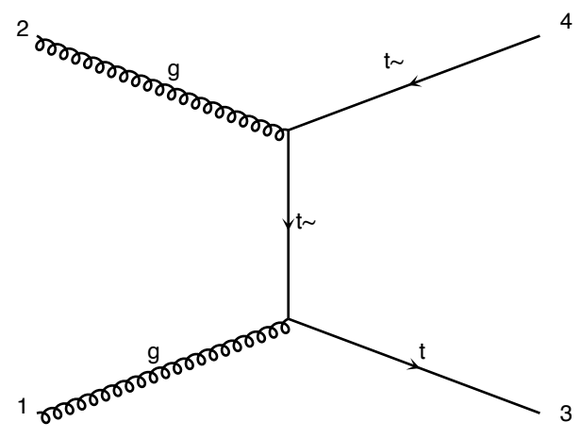


diagram 2 QCD=2, QED=0

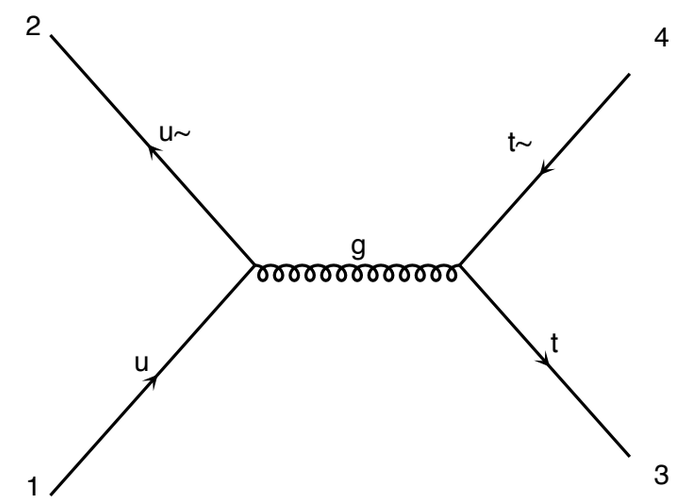


diagram 1 QCD=2, QED=0

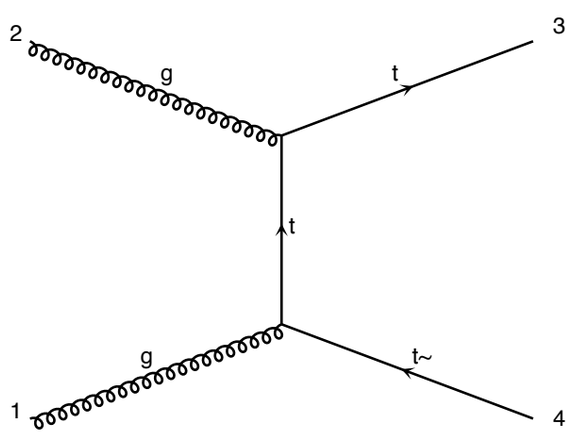


diagram 3 QCD=2, QED=0

gg: 3 diagrams

$q\bar{q}$: 1 diagram

Exercise 1: solution

- Output the code
  - > output mytestdir

```
INFO: initialize a new directory: mytestdir
INFO: remove old information in mytestdir
INFO: Creating files in directory P0_gg_ttx
INFO: Generating Feynman diagrams for Process: g g > t t~ WEIGHTED=2
INFO: Finding symmetric diagrams for subprocess group gg_ttx
INFO: Creating files in directory P0_qq_ttx
INFO: Generating Feynman diagrams for Process: u u~ > t t~ WEIGHTED=2
INFO: Finding symmetric diagrams for subprocess group qq_ttx
History written to /Users/marcozaro/Physics/MadGraph/MG5_aMC_v2_2_2/mytestdir/Cards/proc_card_mg5.dat
Generated helas calls for 2 subprocesses (0 diagrams) in 0.000 s
Wrote files for 16 helas calls in 0.102 s
Export UFO model to MG4 format
ALOHA: aloha creates FFV1 routines
ALOHA: aloha creates VVV1 set of routines with options: P0
save configuration file to /Users/marcozaro/Physics/MadGraph/MG5_aMC_v2_2_2/mytestdir/Cards/me5_configuration.txt
INFO: Use Fortran compiler gfortran
INFO: Generate jpeg diagrams
INFO: Generate web pages
Output to directory /Users/marcozaro/Physics/MadGraph/MG5_aMC_v2_2_2/mytestdir done.
```

Exercise 1: solution

- Compute the cross-section at the LHC (8 TeV) for m_t=170 GeV
  - > launch

```
The following switches determine which programs are run:
 1 Run the pythia shower/hadronization:                    pythia=NOT INSTALLED
 2 Run PGS as detector simulator:                             pgs=NOT INSTALLED
 3 Run Delphes as detector simulator:                      delphes=NOT INSTALLED
 4 Decay particles with the MadSpin module:                madspin=OFF
 5 Add weight to events based on coupling parameters:    reweight=OFF
  Either type the switch number (1 to 5) to change its default setting,
  or set any switch explicitly (e.g. type 'madspin=ON' at the prompt)
  Type '0', 'auto', 'done' or just press enter when you are done.
 [0, 4, 5, auto, done, madspin=ON, madspin=OFF, madspin, reweight=ON, ... ][60s to answer]
```

  - > 0 (let's keep it simple ;-)

```
Do you want to edit a card (press enter to bypass editing)?
  1 / param      : param_card.dat
  2 / run        : run_card.dat
 you can also
   - enter the path to a valid card or banner.
   - use the 'set' command to modify a parameter directly.
     The set option works only for param_card and run_card.
     Type 'help set' for more information on this command.
   - call an external program (ASperGE/MadWidth/...).
     Type 'help' for the list of available command
 [0, done, 1, param, 2, run, enter path][60s to answer]
```

  - edit the cards

Exercise 1: solution

- Compute the cross-section at the LHC (8 TeV) for m_t=170 GeV

run_card

```
#*********************
# Running parameters
#*********************
#
#*********************************************************************
# Tag name for the run (one word)                                    *
#*********************************************************************
  tag_1     = run_tag ! name of the run
#*********************************************************************
# Run to generate the grid pack                                      *
#*********************************************************************
  .false.     = gridpack  !True = setting up the grid pack
#*********************************************************************
# Number of events and rnd seed                                      *
# Warning: Do not generate more than 1M events in a single run       *
# If you want to run Pythia, avoid more than 50k events in a run.    *
#*********************************************************************
  10000 = nevents ! Number of unweighted events requested
      0       = iseed   ! rnd seed (0=assigned automatically=default))
#*********************************************************************
# Collider type and energy                                           *
# lpp: 0=No PDF, 1=proton, -1=antiproton, 2=photon from proton,      *
#                                         3=photon from electron     *
#*********************************************************************
        1     = lpp1    ! beam 1 type
        1     = lpp2    ! beam 2 type
     6500     = ebeam1  ! beam 1 total energy in GeV
     6500     = ebeam2  ! beam 2 total energy in GeV
#*********************************************************************
# Beam polarization from -100 (left-handed) to 100 (right-handed)    *
#*********************************************************************
        0     = polbeam1 ! beam polarization for beam 1
        0     = polbeam2 ! beam polarization for beam 2
#*********************************************************************
# PDF CHOICE: this automatically fixes also alpha_s and its evol.    *
#*********************************************************************
                                                    47,1          5%
```

param_card

```
#####################################
## INFORMATION FOR MASS
#####################################
Block mass
    5 4.700000e+00 # MB
    6 1.730000e+02 # MT
   15 1.777000e+00 # MTA
   23 9.118800e+01 # MZ
   25 1.250000e+02 # MH
## Dependent parameters, given by model restrictions.
## Those values should be edited following the
## analytical expression. MG5 ignores those values
## but they are important for interfacing the output of MG5
## to external program such as Pythia.
  1 0.000000 # d : 0.0
  2 0.000000 # u : 0.0
  3 0.000000 # s : 0.0
  4 0.000000 # c : 0.0
  11 0.000000 # e- : 0.0
  12 0.000000 # ve : 0.0
  13 0.000000 # mu- : 0.0
  14 0.000000 # vm : 0.0
  16 0.000000 # vt : 0.0
  21 0.000000 # g : 0.0
  22 0.000000 # a : 0.0
  24 80.419002 # w+ : cmath.sqrt(MZ__exp__2/2. + cmath.sqrt(MZ__exp__4
/4. - (aEW*cmath.pi*MZ__exp__2)/(Gf*sqrt__2)))
<_v2_2_2/mytestdir/Cards/param_card.dat" 78L, 2770C 1,1          Top
```

Exercise 1: solution

- Compute the cross-section at the LHC (8 TeV) for m_t=172 GeV
  - One can also set the parameters without editing the cards (useful for scripting)
  - `> set ebeam1 4000`
  - `> set ebeam2 4000`
  - `> set MT 172.`
  - `> done`

Exercise 1: solution

- Compute the cross-section at the LHC (8 TeV) for m_t=172 GeV
  - One can also set the parameters without editing the cards (useful for scripting)
    - `> set ebeam1 4000`
    - `> set ebeam2 4000`
    - `> set MT 172.`
    - `> done`

```
. . .
Working on SubProcesses
    P0_gg_ttx
    P0_qq_ttx
INFO:  Idle: 0,  Running: 1,  Completed: 1 [ current time: 15h13 ]
INFO: End survey
refine 10000
Creating Jobs
INFO: Refine results to 10000
    P0_gg_ttx
    P0_qq_ttx
INFO:  Idle: 6,  Running: 4,  Completed: 3 [  3.2s  ]
INFO:  Idle: 2,  Running: 4,  Completed: 7 [  6.6s  ]
INFO:  Idle: 0,  Running: 1,  Completed: 12 [  9.7s  ]
INFO: Combining runs
INFO: finish refine
refine 10000
Creating Jobs
INFO: Refine results to 10000
    P0_gg_ttx
    P0_qq_ttx
INFO: Combining runs
INFO: finish refine
combine_events
INFO: Combining Events
  === Results Summary for run: run_01 tag: tag_1 ===

     Cross-section :   160.1 +- 0.2302 pb
     Nb of events :  10000
```

Monitor via the web interface

Results in the sm for p p > t t~

Currently Running

Run Name	Tag Name	Cards	Results	Status/Jobs		
				Queued	Running	Done
run_01	tag_1	param_card run_card plot_card	160.1 ± 0.2302 (pb)	Combining Events		

Available Results

Run	Collider	Banner	Cross section (pb)	Events	Data	Output	Action
run_01	p p 4000 x 4000 GeV	tag_1	160.1 ± 0.23	No events yet		banner only	remove run / re-run from the banner

Main Page

Exercise 1: solution

- Script it:
  - open a text file (`mymg5amc.txt`) and put the commands inside:

```
generate p p > t t~
output mytestdir
launch
set ebeam1 4000
set ebeam2 4000
set MT 172
```

  - launch MG5_aMC@NLO with that file
  - `./bin/mg5_amc mymg5amc.txt`

Exercise 1:
Extra questions:

- Are b-quarks included in the initial state? If not, how can I include them?
- `> display processes`

```
Process: g g > t t~ WEIGHTED=2
Process: u u~ > t t~ WEIGHTED=2
Process: c c~ > t t~ WEIGHTED=2
Process: d d~ > t t~ WEIGHTED=2
Process: s s~ > t t~ WEIGHTED=2
```

- No b-quark appears. Note that at the startup you have

```
Defined multiparticle p = g u c d s u~ c~ d~ s~
Defined multiparticle j = g u c d s u~ c~ d~ s~
```

- You can add the b/$\bar{b}$ to the multiparticle labels
- `> define p = p b b~`

```
Defined multiparticle p = g u c d s u~ c~ d~ s~ b b~
```

- `> display multiparticles`
- For consistency one should use a model with m_b=0
- `> import model sm-no_b_mass`

Exercise 1:
Extra questions:

Exercise 1:
Extra questions:

- Are b-quarks included in the initial state? If not, how can I include them?

Exercise 1:
Extra questions:

- Are b-quarks included in the initial state? If not, how can I include them?
- Regenerate the process
  - `> generate p p > t t~`
  - `> display processes`

```
Process: g g > t t~ WEIGHTED=2
Process: u u~ > t t~ WEIGHTED=2
Process: c c~ > t t~ WEIGHTED=2
Process: d d~ > t t~ WEIGHTED=2
Process: s s~ > t t~ WEIGHTED=2
Process: b b~ > t t~ WEIGHTED=2
```

Exercise 1:
Extra questions:

- Are b-quarks included in the initial state? If not, how can I include them?
- Regenerate the process
  - `> generate p p > t t~`
  - `> display processes`

```
Process: g g > t t~ WEIGHTED=2
Process: u u~ > t t~ WEIGHTED=2
Process: c c~ > t t~ WEIGHTED=2
Process: d d~ > t t~ WEIGHTED=2
Process: s s~ > t t~ WEIGHTED=2
Process: b b~ > t t~ WEIGHTED=2
```

- Does it make a big difference?
  - `> output`
  - `> launch`
  - `> set ebeam1 4000`
  - `> set ebeam2 4000`
  - `> set MT 172`

Exercise 1:
Extra questions:

- Are b-quarks included in the initial state? If not, how can I include them?
- Regenerate the process
  - `> generate p p > t t~`
  - `> display processes`

```
Process: g g > t t~ WEIGHTED=2
Process: u u~ > t t~ WEIGHTED=2
Process: c c~ > t t~ WEIGHTED=2
Process: d d~ > t t~ WEIGHTED=2
Process: s s~ > t t~ WEIGHTED=2
Process: b b~ > t t~ WEIGHTED=2
```

- Does it make a big difference?
  - `> output`
  - `> launch`
  - `> set ebeam1 4000`
  - `> set ebeam2 4000`
  - `> set MT 172`

```
Cross-section :   160.4 +- 0.231 pb
Nb of events :  10000
```

Without b

```
Cross-section :   160.1 +- 0.2302 pb
Nb of events :  10000
```

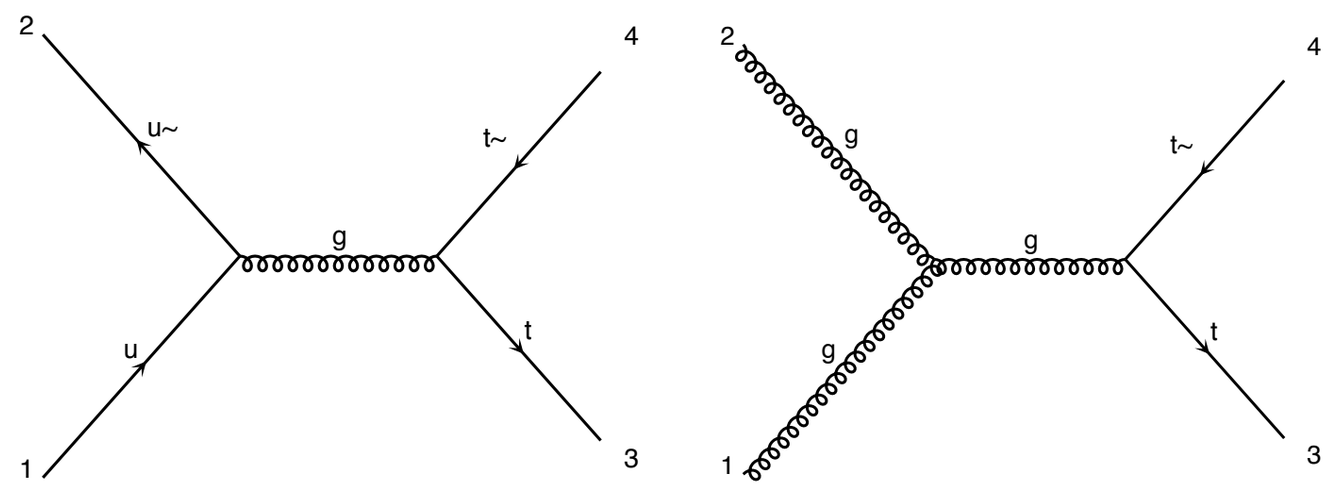
Exercise 1:
Extra questions:

Exercise 1:
Extra questions:

- Are diagrams with photons/z included? If not, how can I include them? How much does the cross-section change? What is that 'WEIGHTED'?

Exercise 1:
Extra questions:

- Are diagrams with photons/z included? If not, how can I include them? How much does the cross-section change? What is that 'WEIGHTED'?
  - `> display diagrams`



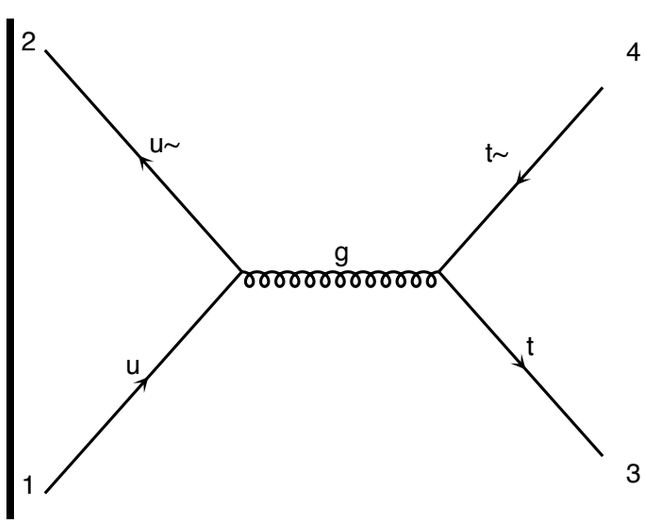
  - No photon/z appear.
  - Are we missing anything important?
  -

Exercise 1:
Extra questions:

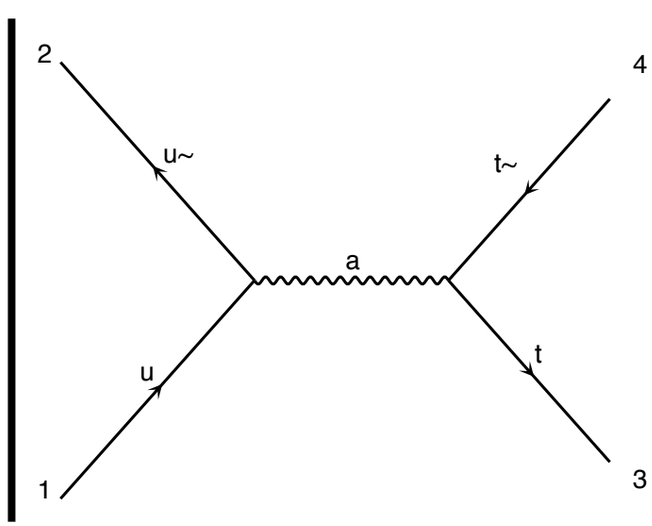
- Are diagrams with photons/z included? If not, how can I include them? How much does the cross-section change? What is that 'WEIGHTED'?
  - `> display diagrams`
  - No photon/z appear.
  - Are we missing anything important?

Exercise 1:
Extra questions:

- Are diagrams with photons/z included? If not, how can I include them? How much does the cross-section change? What is that 'WEIGHTED'?
  - > `display diagrams`
  - No photon/z appear.
  - Are we missing anything important?



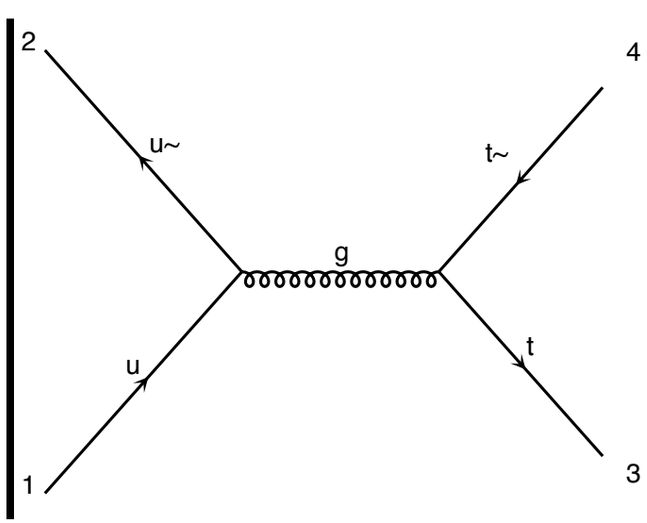
$$\sim \alpha_s^2$$



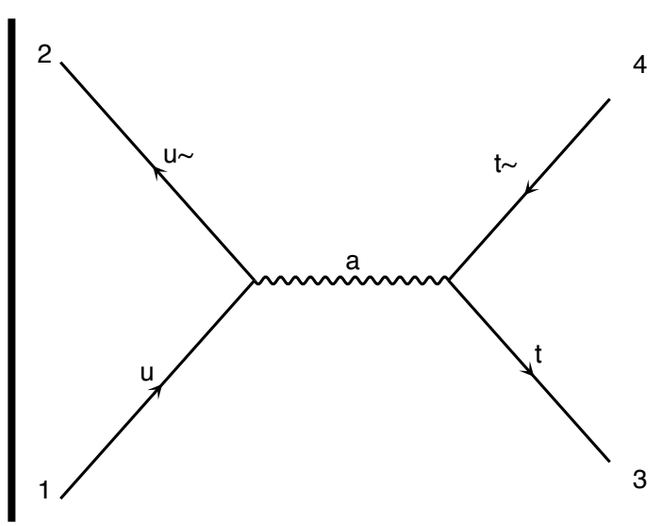
$$\sim \alpha_e^2$$

Exercise 1:
Extra questions:

- Are diagrams with photons/z included? If not, how can I include them? How much does the cross-section change? What is that 'WEIGHTED'?
  - > `display diagrams`
  - No photon/z appear.
  - Are we missing anything important?



$$\sim \alpha_s^2$$



$$\sim \alpha_e^2 \, \frac{\alpha_e}{\alpha_s} \lesssim 0.1$$

Exercise 1:
Extra questions:

- Are diagrams with photons/z included? If not, how can I include them? How much does the cross-section change?
  What is that 'WEIGHTED'?
  - > `display diagrams`
  - No photon/z appear.
  - Are we missing anything important? Does not seem the case
  - How to have them anyway?
  - MG5 exploits the hierarchy between QCD and QED couplings in order to give the leading (i.e. with most QCD) contribution to the cross-section by default
  - It assign WEIGHTED order =1(=2) to QCD (QED) vertices and generates the process with minimum WEIGHTED order

Exercise 1:
Extra questions:

- Are diagrams with photons/z included? If not, how can I include them? How much does the cross-section change?
  What is that 'WEIGHTED'?
  - `> display diagrams`
  - No photon/z appear.
  - Are we missing anything important? Does not seem the case
  - How to have them anyway?
  - MG5 exploits the hierarchy between QCD and QED couplings in order to give the leading (i.e. with most QCD) contribution to the cross-section by default
  - It assign WEIGHTED order =1(=2) to QCD (QED) vertices and generates the process with minimum WEIGHTED order

Exercise 1:
Extra questions:

Exercise 1:
Extra questions:

- Are diagrams with photons/z included? If not, how can I include them? How much does the cross-section change?
  - `> generate p p > t t~ WEIGHTED=4`
  - `> display diagrams`

Exercise 1:
Extra questions:

- Are diagrams with photons/z included? If not, how can I include them? How much does the cross-section change?
  - `> generate p p > t t~ WEIGHTED=4`
  - `> display diagrams`

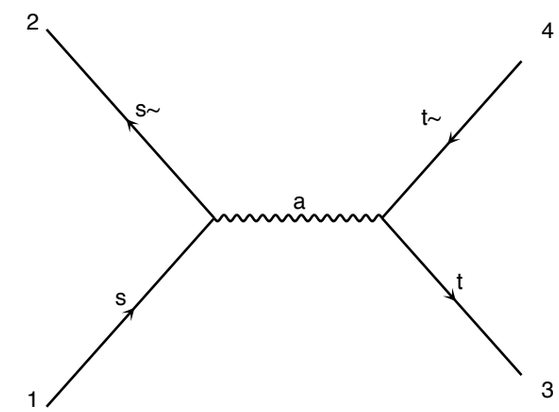


diagram 1 QCD=0, QED=2

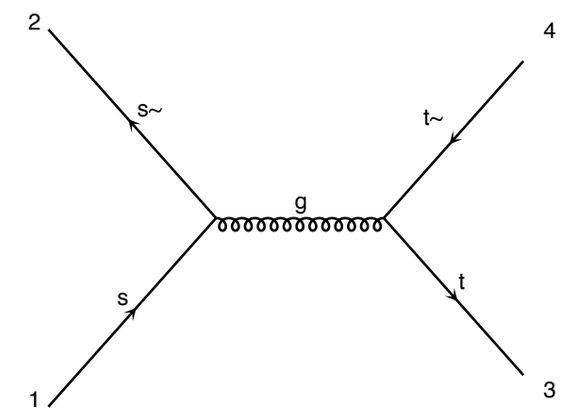


diagram 2 QCD=2, QED=0

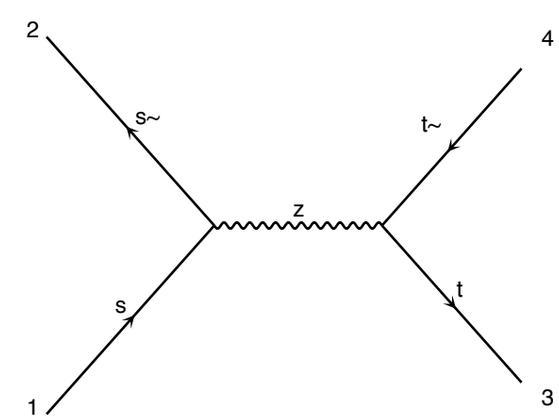


diagram 3 QCD=0, QED=2

Exercise 1:
Extra questions:

- Are diagrams with photons/z included? If not, how can I include them? How much does the cross-section change?
  - `> generate p p > t t~ WEIGHTED=4`
  - `> display diagrams`
  - `> output …`
  - `> launch`
  - `> …`

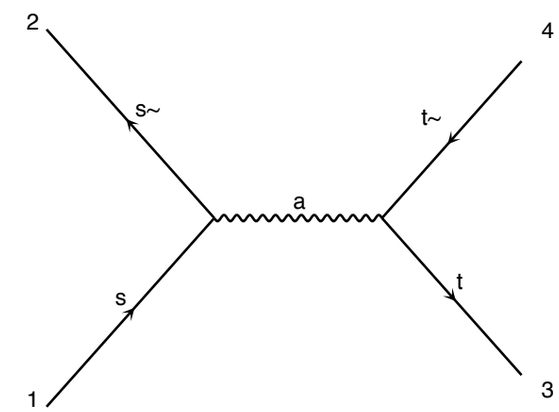


diagram 1 QCD=0, QED=2

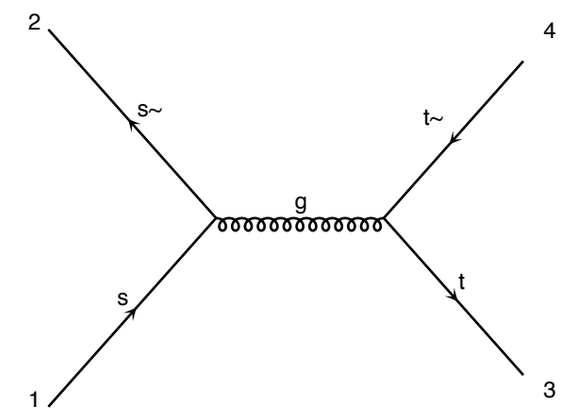


diagram 2 QCD=2, QED=0

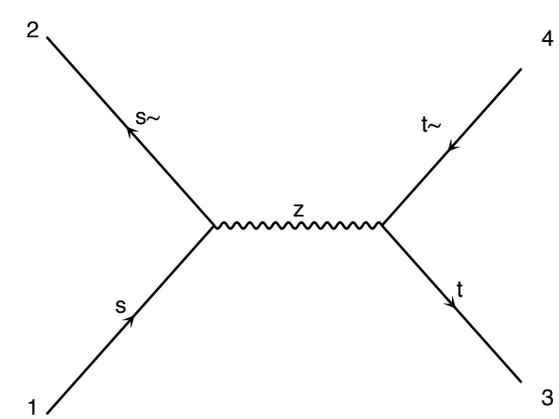


diagram 3 QCD=0, QED=2

Exercise 1:
Extra questions:

- Are diagrams with photons/z included? If not, how can I include them? How much does the cross-section change?
  - `> generate p p > t t~ WEIGHTED=4`
  - `> display diagrams`
  - `> output …`
  - `> launch`
  - `> …`

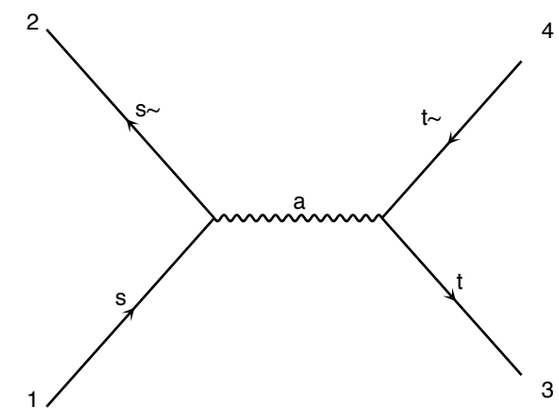


diagram 1 QCD=0, QED=2

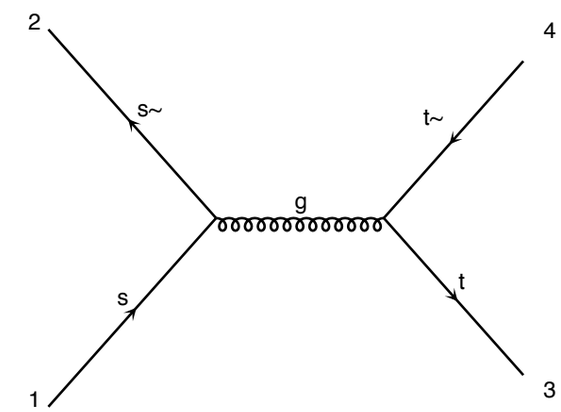


diagram 2 QCD=2, QED=0

```
Cross-section :   160.8 +- 0.1999 pb
Nb of events :  10000
```

WEIGHTED=2

```
Cross-section :   160.4 +- 0.231 pb
Nb of events :  10000
```

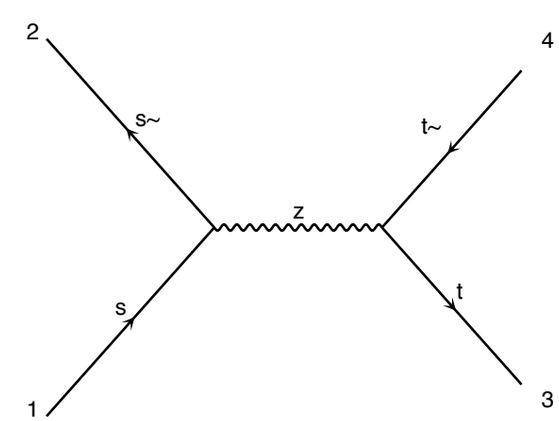


diagram 3 QCD=0, QED=2

Exercise 1:
Extra questions:

Exercise 1:
Extra questions:

- Alternatively, one can specify the coupling powers
  - `> generate p p > t t~ QED=2`
  - orders which are not specified are unconstrained

Exercise 1:
Extra questions:

- Alternatively, one can specify the coupling powers
  - `> generate p p > t t~ QED=2`
  - orders which are not specified are unconstrained
- In order to have only the QED contribution
  - `> generate p p > t t~ QED=2 QCD=0`

Exercise 1:
Extra questions:

- Recompute the $t\bar{t}$ cross-section for m_t=170, 172, 174 … 180 GeV
- Be smart! Script it!
- Create a txt file `myttbar_scan.txt`

```
generate p p > t t~
output mytestdir2
launch
set ebeam1 4000
set ebeam2 4000
set MT 170
launch
set MT 172
launch
set MT 174
launch
set MT 176
launch
set MT 178
launch
set MT 180
```

- `./bin/mg5_aMC myttbar_scan.txt`

Exercise 1:
Extra questions:

- Recompute the $t\bar{t}$ cross-section for m_t=170, 172, 174 … 180 GeV
- Be smart! Script it!
- You can also launch an existing folder, without regenerating the code

```
launch mytestdir2
set ebeam1 4000
set ebeam2 4000
set MT 170
launch
set MT 172
launch
set MT 174
launch
set MT 176
launch
set MT 178
launch
set MT 180
```

Exercise 1:
Extra questions:

- Recompute the $t\bar{t}$ cross-section for m_t=170, 172, 174 … 180 GeV

Results in the sm for p p > t t~

Available Results

Run	Collider	Banner	Cross section (pb)	Events	Data	Output	Action
run_01	p p 4000 x 4000 GeV	tag_1	169.8 ± 0.24	10000	parton madevent	LHE	remove run launch detector simulation
run_02	p p 4000 x 4000 GeV	tag_1	160.1 ± 0.28	10000	parton madevent	LHE	remove run launch detector simulation
run_03	p p 4000 x 4000 GeV	tag_1	151.1 ± 0.2	10000	parton madevent	LHE	remove run launch detector simulation
run_04	p p 4000 x 4000 GeV	tag_1	142.9 ± 0.18	10000	parton madevent	LHE	remove run launch detector simulation
run_05	p p 4000 x 4000 GeV	tag_1	134.7 ± 0.19	10000	parton madevent	LHE	remove run launch detector simulation
run_06	p p 4000 x 4000 GeV	tag_1	127.3 ± 0.16	10000	parton madevent	LHE	remove run launch detector simulation

Main Page

Exercise 1:
Extra questions:

- Recompute the $t\bar{t}$ cross-section for m_t=170, 172, 174 … 180 GeV

Results in the sm for p p > t t~

Available Results

Run	Collider	Banner	Cross section (pb)	Events	Data	Output	Action
run_01	p p 4000 x 4000 GeV	tag_1	169.8 ± 0.24	10000	parton madevent	LHE	remove run launch detector simulation
run_02	p p 4000 x 4000 GeV	tag_1	160.1 ± 0.28	10000	parton madevent	LHE	remove run launch detector simulation
run_03	p p 4000 x 4000 GeV	tag_1	151.1 ± 0.2	10000	parton madevent	LHE	remove run launch detector simulation
run_04	p p 4000 x 4000 GeV	tag_1	142.9 ± 0.18	10000	parton madevent	LHE	remove run launch detector simulation
run_05	p p 4000 x 4000 GeV	tag_1	134.7 ± 0.19	10000	parton madevent	LHE	remove run launch detector simulation
run_06	p p 4000 x 4000 GeV	tag_1	127.3 ± 0.16	10000	parton madevent	LHE	remove run launch detector simulation

Main Page

which folder is what?

Exercise 1:
Extra questions:

- Recompute the $t\bar{t}$ cross-section for m_t=170, 172, 174 … 180 GeV
- Be smart! Script it!
- You can specify the name (instead of `run_01…`) with `-n NAME`

```
launch mytestdir2 -n run_MT170
set ebeam1 4000
set ebeam2 4000
set MT 170
launch -n run_MT172
set MT 172
launch -n run_MT174
set MT 174
launch -n run_MT176
set MT 176
launch -n run_MT178
set MT 178
launch -n run_MT180
set MT 180
```

Exercise 1:
Extra questions:

- Since recently, multiple values can be specified for parameters. Just set in the `param_card`, instead of the top mass
  `6 scan:[170,172,174,176,178]`

Exercise 1:
Extra questions:

- Recompute the $t\bar{t}$ cross-section for m_t=170, 172, 174 … 180 GeV

Run	Collider	Banner	Cross section (pb)	Events	Data	Output	Action
run_01	p p 4000 x 4000 GeV	tag_1	169.8 ± 0.24	10000	parton madevent	LHE	remove run launch detector simulation
run_02	p p 4000 x 4000 GeV	tag_1	160.1 ± 0.28	10000	parton madevent	LHE	remove run launch detector simulation
run_03	p p 4000 x 4000 GeV	tag_1	151.1 ± 0.2	10000	parton madevent	LHE	remove run launch detector simulation
run_04	p p 4000 x 4000 GeV	tag_1	142.9 ± 0.18	10000	parton madevent	LHE	remove run launch detector simulation
run_05	p p 4000 x 4000 GeV	tag_1	134.7 ± 0.19	10000	parton madevent	LHE	remove run launch detector simulation
run_06	p p 4000 x 4000 GeV	tag_1	127.3 ± 0.16	10000	parton madevent	LHE	remove run launch detector simulation
run_MT170	p p 4000 x 4000 GeV	tag_1	170 ± 0.22	10000	parton madevent	LHE	remove run launch detector simulation
run_MT172	p p 4000 x 4000 GeV	tag_1	159.6 ± 0.22	10000	parton madevent	LHE	remove run launch detector simulation
run_MT174	p p 4000 x 4000 GeV	tag_1	151.1 ± 0.22	10000	parton madevent	LHE	remove run launch detector simulation
run_MT176	p p 4000 x 4000 GeV	tag_1	142.6 ± 0.19	10000	parton madevent	LHE	remove run launch detector simulation
run_MT178	p p 4000 x 4000 GeV	tag_1	134.7 ± 0.18	10000	parton madevent	LHE	remove run launch detector simulation
run_MT180	p p 4000 x 4000 GeV	tag_1	127.2 ± 0.24	10000	parton madevent	LHE	remove run launch detector simulation

Main Page

Part 2:
NLO

NLO run mode(s)

- While the LO (MadEvent) code will always compute the cross section and generate events, this is not the case for NLO runs
- The NLO code can be run in two ways:
  - Fixed-order (*à la* MCFM): no events will be generated, just compute the cross section (possibly within cuts). Optionally fill histograms on-the-fly
  - NLO+PS: generate a LHE event sample and compute the cross section (both possibly within cuts). The event file must be showered in order to obtain physical distributions.
  - Note that Fixed-order ≠ LHE!

NLO exercise
$t\bar{t}$ production at NLO

Part I

- Learn the syntax:
  - > `tutorial aMCatNLO`
- Generate the code for $t\bar{t}$ production at NLO
- Compute the LO and NLO cross-section (run at fixed order)
- Select the analysis `analysis_HwU_pp_ttx.o` in the `FO_analyse_card` to generate histograms (need `GnuPlot` installed)
- In the NLO histograms, which of these variables are described at the NLO? $p_T(t)$, $p_T(t\bar{t})$, $y(t)$, $M(t\bar{t})$, $\Delta\phi(t\bar{t})$
- What are the histograms with `muR=...` `muF=...` for?

NLO exercise
Solution

Part I

- Learn the syntax:
  - > `tutorial aMCatNLO`
- Generate the code for $t\bar{t}$ production at NLO
  - > `generate p p > t t~ [QCD]`

```
The current model sm does not allow to generate loop corrections of type QCD.
MG5_aMC now loads 'loop_sm'.
 import model loop_sm
. . .
INFO: Generating FKS-subtracted matrix elements for born process: g g > t t~  [ QCD ] (1 / 9)
. . .
```

  - > `output my_ttbar_nlo`
- Compute the LO and NLO cross-section
  - > `launch`

NLO exercise
Solution

Part I

- Learn the synt
  - > tutorial
- Generate the c
  - > generate

```
The current model sm does n
MG5_aMC now loads 'loop_sm'
 import model loop_sm
. . .
INFO: Generating FKS-subtra
. . .
```

  - > output m
- Compute the l
  - > launch

```
INFO: ************************************************************
*                                                          *
*           W E L C O M E  to  M A D G R A P H 5           *
*                      a M C @ N L O                       *
*                                                          *
*                *                       *                 *
*                  *        * *        *                   *
*                    * * * * 5 * * * *                     *
*                  *        * *        *                   *
*                *                       *                 *
*                                                          *
*         VERSION 2.2.1                 2014-09-25         *
*                                                          *
*    The MadGraph5_aMC@NLO Development Team - Find us at   *
*                  http://amcatnlo.cern.ch                 *
*                                                          *
*               Type 'help' for in-line help.              *
*                                                          *
************************************************************
launch auto
The following switches determine which operations are executed:
 1 Perturbative order of the calculation:                          order=NLO
 2 Fixed order (no event generation and no MC@[N]LO matching):   fixed_order=OFF
 3 Shower the generated events:                                     shower=ON
 4 Decay particles with the MadSpin module:                       madspin=OFF
  Either type the switch number (1 to 4) to change its default setting,
  or set any switch explicitly (e.g. type 'order=LO' at the prompt)
  Type '0', 'auto', 'done' or just press enter when you are done.
 [0, 1, 2, 3, 4, auto, done, order=LO, order=NLO, ... ][60s to answer]
> fixed_order=ON
> order=LO (for LO run)
```

NLO exercise
Solution

Part I

- Learn the sy
  - > tutori
- Generate th
  - > genera

```
The current model sm do
MG5_aMC now loads 'loop
 import model loop_sm
. . .
INFO: Generating FKS-su
. . .
```

  - > output
- Compute th
  - > launch

```
INFO: **********************************************************
*                                                               *
*            W E L C O M E  to  M A D G R A P H 5               *
                                                                *
                                                                *
                                                                *
                                                                *
                                                                *
                                                                *
                                                                *
                                                                *
                                                                *
                                                                *
                                                                *
                                                                *
                                                                *
                                                                *
                                                                *
                                                                *
                                                                ted:
                                                                         order=NLO
                                                                g):   fixed_order=OFF
                                                                          shower=ON
                                                                         madspin=OFF
                                                                ult setting,
                                                                 prompt)
  Type '0', 'auto', 'done' or just press enter when you are done.
 [0, 1, 2, 3, 4, auto, done, order=LO, order=NLO, ... ][60s to answer]
> fixed_order=ON
> order=LO (for LO run)
```

```
INFO:
      Final results and run summary:
      Process p p > t t~ [QCD]
      Run at p-p collider (6500 + 6500 GeV)
      Total cross-section:      6.871e+02 +- 5.9e+00 pb
      Ren. and fac. scale uncertainty: +9.7% -11.7%

INFO: The results of this run and the HwU and GnuPlot
files with the plots have been saved in /Users/marcozaro/
Physics/MadGraph/2.2.3new/my_tt_nlo_qcd/Events/run_01

INFO:
      Final results and run summary:
      Process p p > t t~ [QCD]
      Run at p-p collider (6500 + 6500 GeV)
      Total cross-section:      4.622e+02 +- 2.2e+00 pb
      Ren. and fac. scale uncertainty: +29.8% -22.3%

INFO: The results of this run and the HwU and GnuPlot
files with the plots have been saved in /Users/marcozaro/
Physics/MadGraph/2.2.3new/my_tt_nlo_qcd/Events/run_02_LO
```

NLO exercise
Solution

Part I

- Select the analysis `analysis_HwU_pp_ttx` in the `FO_analyse_card` to generate histograms
  - `> launch my_ttbar_nlo`

```
The following switches determine which operations are executed:
 1 Perturbative order of the calculation:                          order=NLO
 2 Fixed order (no event generation and no MC@[N]LO matching):   fixed_order=ON
 3 Shower the generated events:                                       shower=OFF
 4 Decay particles with the MadSpin module:                         madspin=OFF
  Either type the switch number (1 to 4) to change its default setting,
  or set any switch explicitly (e.g. type 'order=LO' at the prompt)
  Type '0', 'auto', 'done' or just press enter when you are done.
 [0, 1, 2, 3, 4, auto, done, order=LO, order=NLO, ... ][60s to answer]
>
INFO: will run in mode: NLO
Do you want to edit a card (press enter to bypass editing)?
  1 / param      : param_card.dat
  2 / run        : run_card.dat
  3 / FO_analyse : FO_analyse_card.dat
 you can also
   - enter the path to a valid card or banner.
   - use the 'set' command to modify a parameter directly.
     The set option works only for param_card and run_card.
     Type 'help set' for more information on this command.
   - call an external program (ASperGE/MadWidth/...).
     Type 'help' for the list of available command
 [0, done, 1, param, 2, run, 3, FO_analyse, enter path][60s to answer]
```

Part I

- Select t

FO_ana

- > la

```
The following
 1 Perturbati
 2 Fixed orde
 3 Shower the
 4 Decay part
  Either type
  or set any s
  Type '0', 'a
 [0, 1, 2, 3,
>
INFO: will run
Do you want to
  1 / param
  2 / run
  3 / FO_analy
 you can also
   - enter the
   - use the
     The set o
     Type 'he
   - call an e
     Type 'he
 [0, done, 1,
```

```
#################################################################
#
# This file contains the settings for analyses to be linked to aMC@NLO
# fixed order runs. Analyse files are meant to be put (or linked)
# inside <PROCDIR>/FixedOrderAnalysis/ (<PROCDIR> is the name of the
# exported process directory). See the
# <PROCDIR>/FixedOrderAnalysis/analysis_template.f file for details on
# how to write your own analysis.
#
#################################################################
#
# Analysis format. Can either be 'topdrawer', 'root', 'HwU' or 'none'.
# When choosing HwU, it comes with a GnuPlot wrapper. When choosing
# topdrawer, the histogramming package 'dbook.f' is included in the
# code, while when choosing root the 'rbook_fe8.f' and 'rbook_be8.cc'
# are included. If 'none' is chosen, all the other entries below have
# to be set empty.
FO_ANALYSIS_FORMAT = HwU
#
# Needed extra-libraries (FastJet is already linked):
FO_EXTRALIBS =
#
# (Absolute) path to the extra libraries. Directory names should be
# separated by white spaces.
FO_EXTRAPATHS =
#
# (Absolute) path to the dirs containing header files needed by the
# libraries (e.g. C++ header files):
FO_INCLUDEPATHS =
#
# User's analysis (to be put in the <PROCDIR>/FixedOrderAnalysis/
# directory). Please use .o as extension and white spaces to separate
# files.
FO_ANALYSE = analysis_td_template.o
#
#
## When linking with root, the following settings are a working
## example on lxplus (CERN). When using this, comment out the lines
## above and replace <PATH_TO_ROOT> with the physical path to root,
## e.g. /afs/cern.ch/sw/lcg/app/releases/ROOT/5.34.11/x86_64-slc6-gcc46-dbg/root/
#FO_ANALYSIS_FORMAT = root
#FO_EXTRALIBS = Core Cint Hist Matrix MathCore RIO dl Thread
#FO_EXTRAPATHS = <PATH_TO_ROOT>/lib
#FO_INCLUDEPATHS = <PATH_TO_ROOT>/include
#FO_ANALYSE = analysis_root_template.o
```

Part I

- Select t

FO_ana

- > la

```
The following
 1 Perturbati
 2 Fixed orde
 3 Shower the
 4 Decay part
  Either type
  or set any
  Type '0', '
 [0, 1, 2, 3,
>
INFO: will ru
Do you want t
  1 / param
  2 / run
  3 / FO_analy
 you can also
   - enter the
   - use the
     The set
     Type 'he
   - call an
     Type 'he
 [0, done, 1,
```

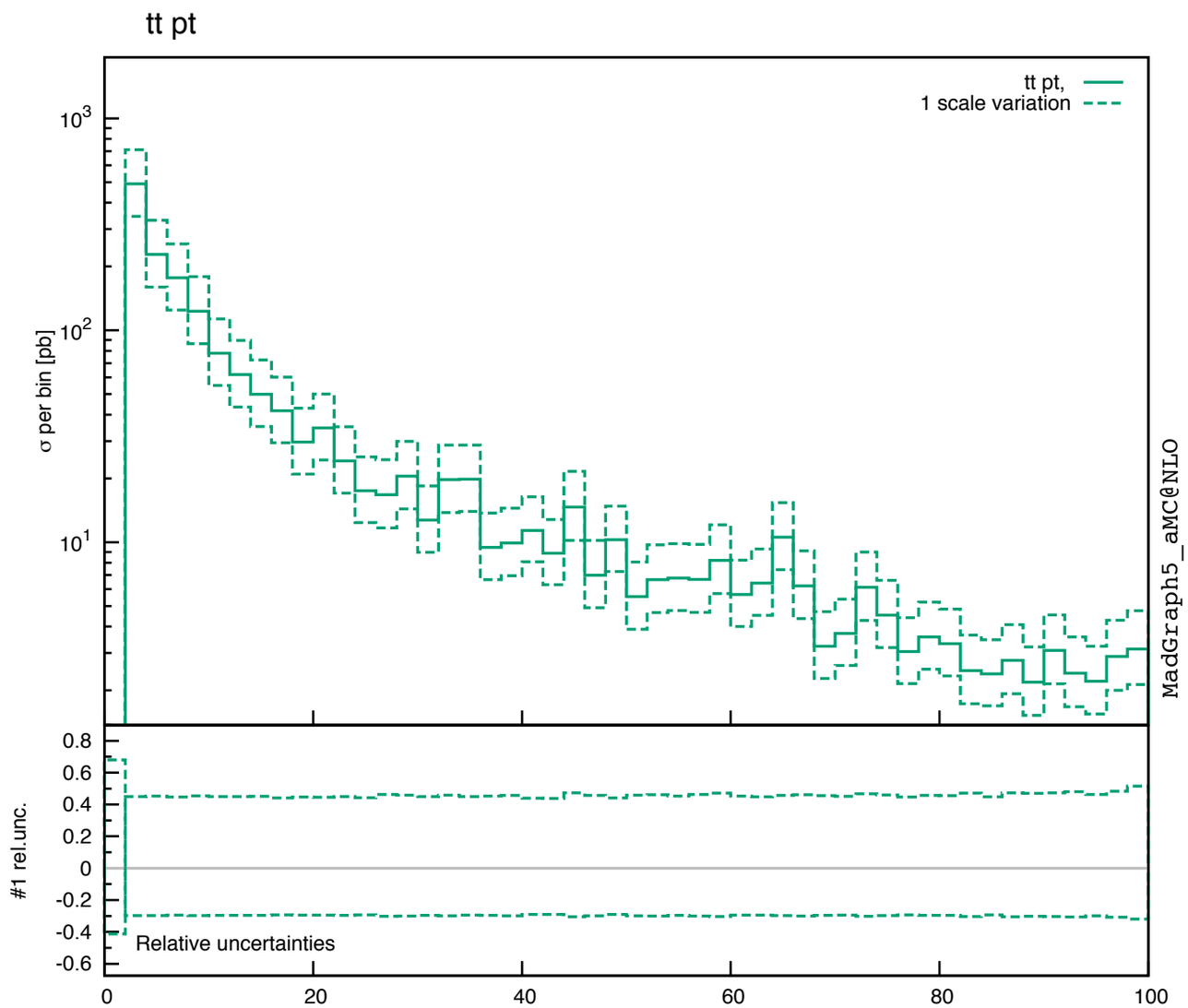
```
#################################################################
#
# This file contains the settings for analyses to be linked to aMC@NLO
# fixed order runs. Analyse files are meant to be put (or linked)
# inside <PROCDIR>/FixedOrderAnalysis/ (<PROCDIR> is the name of the
# exported process directory). See the
# <PROCDIR>/FixedOrderAnalysis/analysis_template.f file for details on
# how to write your own analysis.
#
#################################################################
#
# Analysis format. Can either be 'topdrawer', 'root', 'HwU' or 'none'.
# When choosing HwU, it comes with a GnuPlot wrapper. When choosing
# topdrawer, the histogramming package 'dbook.f' is included in the
# code, while when choosing root the 'rbook_fe8.f' and 'rbook_be8.cc'
# are included. If 'none' is chosen, all the other entries below have
# to be set empty.
FO_ANALYSIS_FORMAT = HwU
#
# Needed extra-libraries (FastJet is already linked):
FO_EXTRALIBS =
#
# (Absolute) path to the extra libraries. Directory names should be
# separated by white spaces.
FO_EXTRAPATHS =
#
# (Absolute) path to the dirs containing header files needed by the
# libraries (e.g. C++ header files):
FO_INCLUDEPATHS =
#
# User's analysis (to be put in the <PROCDIR>/FixedOrderAnalysis/
# directory). Please use .o as extension and white spaces to separate
# files.
FO_ANALYSE = analysis_HwU_pp_ttx.o
#
#
## When linking with root, the following settings are a working
## example on lxplus (CERN). When using this, comment out the lines
## above and replace <PATH_TO_ROOT> with the physical path to root,
## e.g. /afs/cern.ch/sw/lcg/app/releases/ROOT/5.34.11/x86_64-slc6-gcc46-dbg/root/
#FO_ANALYSIS_FORMAT = root
#FO_EXTRALIBS = Core Cint Hist Matrix MathCore RIO dl Thread
#FO_EXTRAPATHS = <PATH_TO_ROOT>/lib
#FO_INCLUDEPATHS = <PATH_TO_ROOT>/include
#FO_ANALYSE = analysis_root_template.o
```

NLO exercise: solution

- The HwU (Histogram with Uncertainties) format

```
 ##& xmin & xmax & central value & dy & delta_mu_min @aux & delta_mu_max @aux & muR=1.00 muF=1.00 & muR=1.00 muF=2.00 &
muR=1.00 muF=0.50 & muR=2.00 muF=1.00 & muR=2.00 muF=2.00 & muR=2.00 muF=0.50 & muR=0.50 muF=1.00 & muR=0.50 muF=2.00 &
muR=0.50 muF=0.50

<histogram> 50 "tt pt |X_AXIS@LIN |Y_AXIS@LOG"
  +0.0000000e+00   +2.0000000e+00   -1.0242367e+03   +2.5047252e+01  -1.7206530e+03   -6.0160203e+02   -1.0242367e+03
-9.0715087e+02   -1.1432407e+03   -6.8421704e+02   -6.0160203e+02   -7.6882229e+02   -1.5496422e+03   -1.3802509e+03
-1.7206530e+03
  +2.0000000e+00   +4.0000000e+00   +4.9088904e+02   +2.0297264e+01  +3.4493531e+02   +7.1188196e+02   +4.9088904e+02
+4.5019210e+02   +5.3086979e+02   +3.7613186e+02   +3.4493531e+02   +4.0679297e+02   +6.5832080e+02   +6.0377117e+02
+7.1188196e+02
  +4.0000000e+00   +6.0000000e+00   +2.2787754e+02   +2.3122314e+01  +1.5999659e+02   +3.3086836e+02   +2.2787754e+02
+2.0857157e+02   +2.4714205e+02   +1.7482611e+02   +1.5999659e+02   +1.8963760e+02   +3.0513912e+02   +2.7932554e+02
+3.3086836e+02
  +6.0000000e+00   +8.0000000e+00   +1.7671803e+02   +9.5392210e+00  +1.2453269e+02   +2.5575724e+02   +1.7671803e+02
+1.6227348e+02   +1.9111959e+02   +1.3562893e+02   +1.2453269e+02   +1.4669918e+02   +2.3651862e+02   +2.1720764e+02
+2.5575724e+02
  +8.0000000e+00   +1.0000000e+01   +1.2311654e+02   +7.1903869e+00  +8.6399100e+01   +1.7898773e+02   +1.2311654e+02
+1.1261446e+02   +1.3369767e+02   +9.4461506e+01   +8.6399100e+01   +1.0258866e+02   +1.6483914e+02   +1.5078780e+02
+1.7898773e+02
  +1.0000000e+01   +1.2000000e+01   +7.8022445e+01   +1.0748137e+01  +5.4873577e+01   +1.1315020e+02   +7.8022445e+01
+7.1570742e+01   +8.4452355e+01   +5.9823787e+01   +5.4873577e+01   +6.4760050e+01   +1.0454718e+02   +9.5909144e+01
+1.1315020e+02
  +1.2000000e+01   +1.4000000e+01   +6.1770611e+01   +3.2903213e+00  +4.3437593e+01   +8.9537046e+01   +6.1770611e+01
```

NLO exercise
Solution



NLO exercise
Solution

Part I

- In the NLO histograms, which of these variables are described at the NLO? $p_T(t)$, $p_T(t\bar{t})$, $y(t)$, $M(t\bar{t})$, $\Delta\phi(t\bar{t})$
  - Some of these variables are trivial at LO, because of 2→2 kinematics
    - t and $\bar{t}$ are always back to back:
      $d\sigma/d\Delta\Phi(t\bar{t}) = \delta(\Delta\Phi - \pi)$
      $d\sigma/dp_T(t\bar{t}) = \delta(p_T - 0)$
  - $p_T(t\bar{t})$ and $\Delta\phi(t\bar{t})$ are non-trivial if the cross-section is at least at NLO: they are effectively described with LO accuracy
  - The other variables are described at NLO

NLO exercise
Solution

Part 1

- What are the histograms with `muR=... muF=...` for?
  - QCD master formula

$$\sigma(pp \to t\bar{t}) = \sum_{ab} \int dx_1 dx_2 f_a(x_1, \mu_F) f_b(x_2, \mu_F) \times \hat{\sigma}(ab \to t\bar{t})$$

    or better

$$\sigma(pp \to t\bar{t}) = \sum_{ab} \int dx_1 dx_2 f_a(x_1, \mu_F) f_b(x_2, \mu_F) \times \hat{\sigma}(ab \to t\bar{t}; \mu_F, \mu_R, \alpha_S(\mu_R))$$

  - What are $\mu_{F/R}$?
    - They are arbitrary scales needed to renormalise the strong coupling and to reabsorb initial state IR-divergences in PDFs, chosen to be of the order of the hard scattering scales (sum of masses, p_T, …)
    - The all-order cross-section is independent of the choice of $\mu_{F/R}$
    - At N^kLO, the dependence is of $N^{k+1}LO$
    - Computing the cross-section with different scales can be a way to estimate uncertainties due to missing higher orders
    - How much scales are varied is arbitrary, usually in the range [0.5, 2]

Scale uncertainties

Scale uncertainties

- Look at the LO and NLO cross-section we have just computed
  - Values with different scales are computed on the fly and the envelope is taken

```
INFO:
      Final results and run summary:
      Process p p > t t~ [QCD]
      Run at p-p collider (6500 + 6500 GeV)
      Total cross-section:      6.871e+02 +- 5.9e+00 pb
      Ren. and fac. scale uncertainty: +9.7% -11.7%

INFO: The results of this run and the TopDrawer file with
the plots have been saved in /Users/marcozaro/Physics/
MadGraph/2.2.3new/my_tt_nlo_qcd/Events/run_01

INFO:
      Final results and run summary:
      Process p p > t t~ [QCD]
      Run at p-p collider (6500 + 6500 GeV)
      Total cross-section:      4.622e+02 +- 2.2e+00 pb
      Ren. and fac. scale uncertainty: +29.8% -22.3%

INFO: The results of this run and the TopDrawer file with
the plots have been saved in /Users/marcozaro/Physics/
MadGraph/2.2.3new/my_tt_nlo_qcd/Events/run_02_LO
```

Scale uncertainties

- Look at the LO and NLO cross-section we have just computed
  - Values with different scales are computed on the fly and the envelope is taken
- Typically LO has larger scale uncertainties

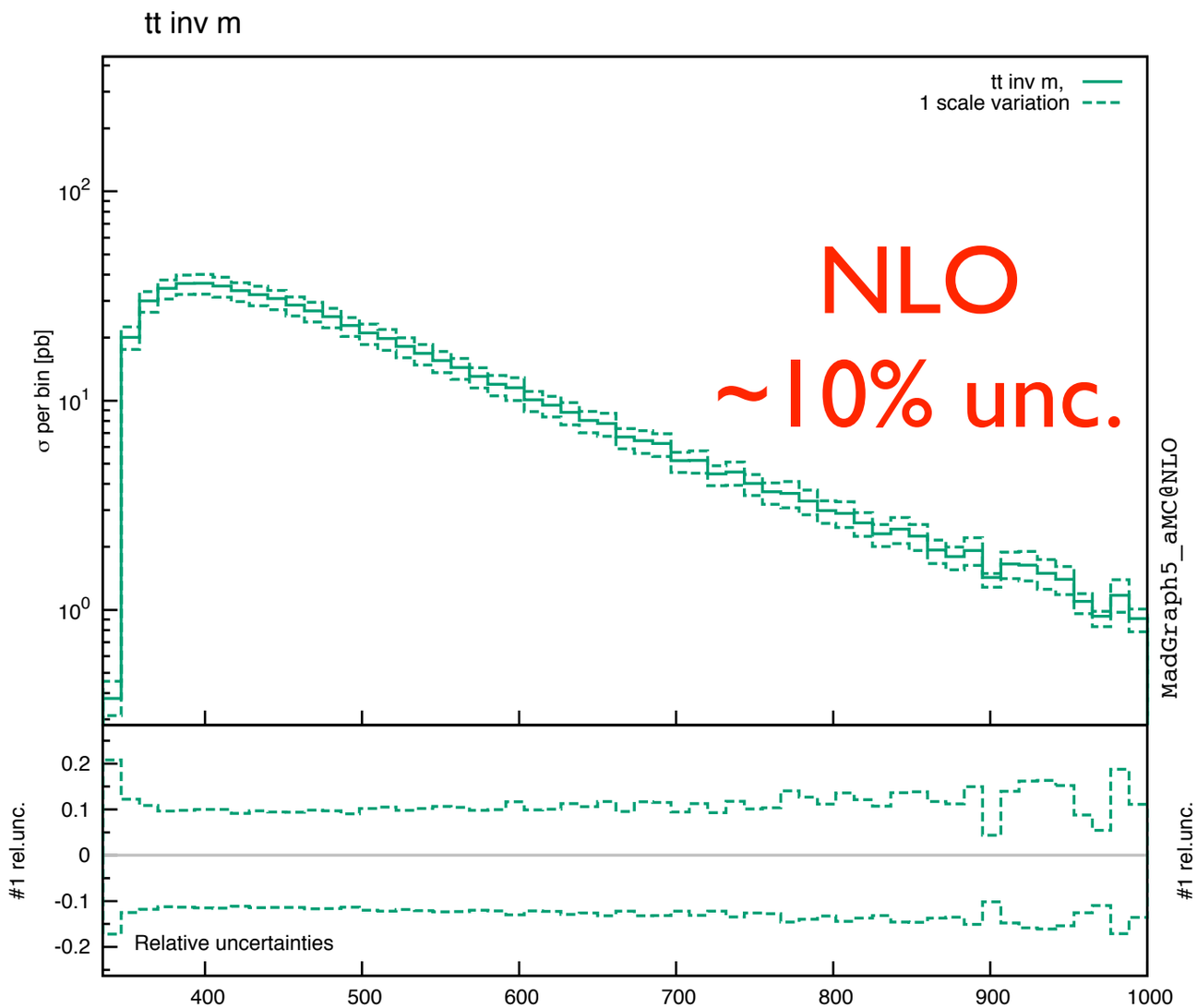
Scale uncertainties

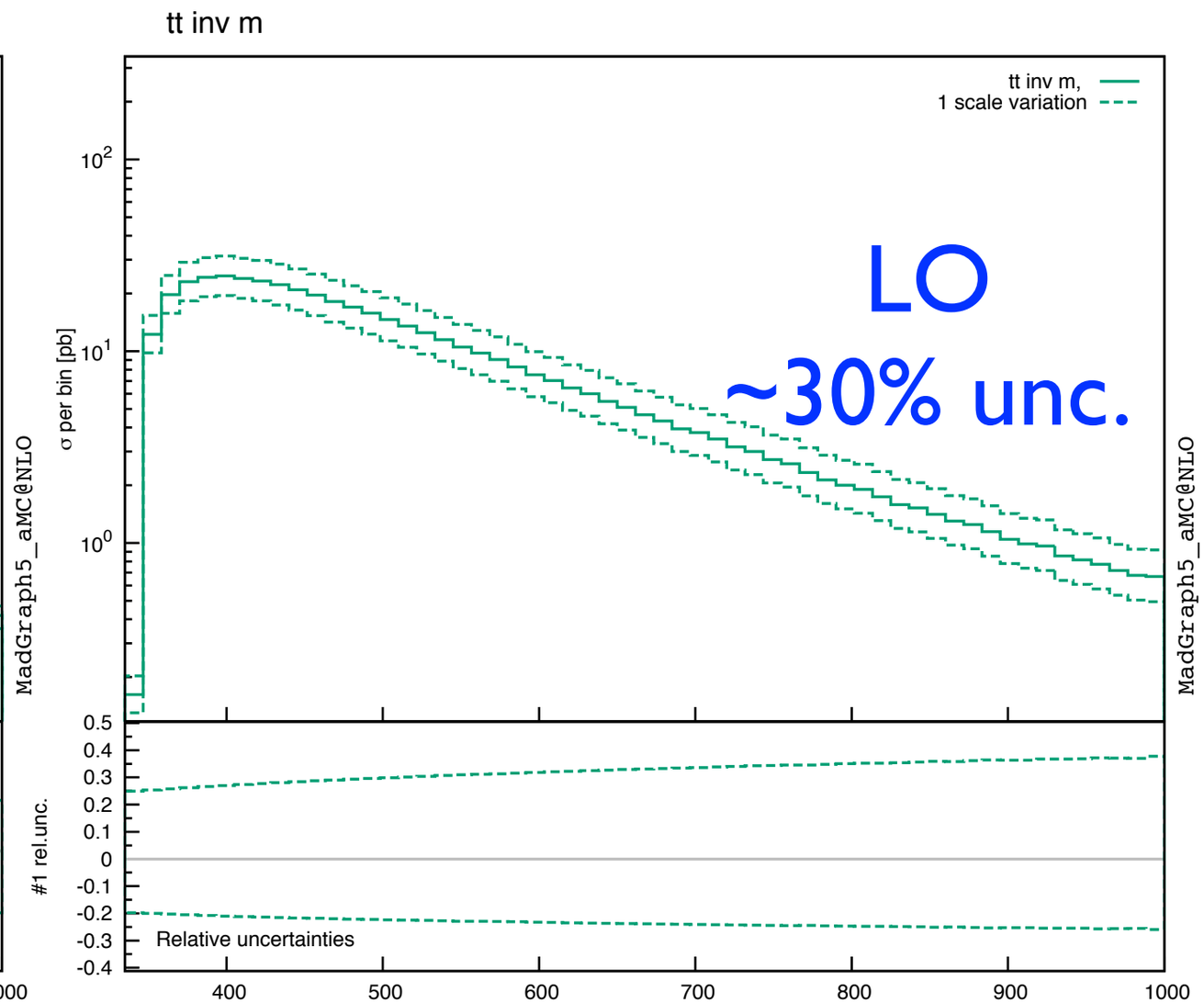
- Look at the LO and NLO cross-section we have just computed
  - Values with different scales are computed on the fly and the envelope is taken
- Typically LO has larger scale uncertainties
- To have scale uncertainties for distributions, one must fill one histogram per scale choice, and then take the envelope

Scale uncertainties

- Look at the LO and NLO cross-section we have just computed
  - Values with different scales are computed on the fly and the envelope is taken
- Typically LO has larger scale uncertainties
- To have scale uncertainties for distributions, one must fill one histogram per scale choice, and then take the envelope
- The same is possible for PDF uncertainties

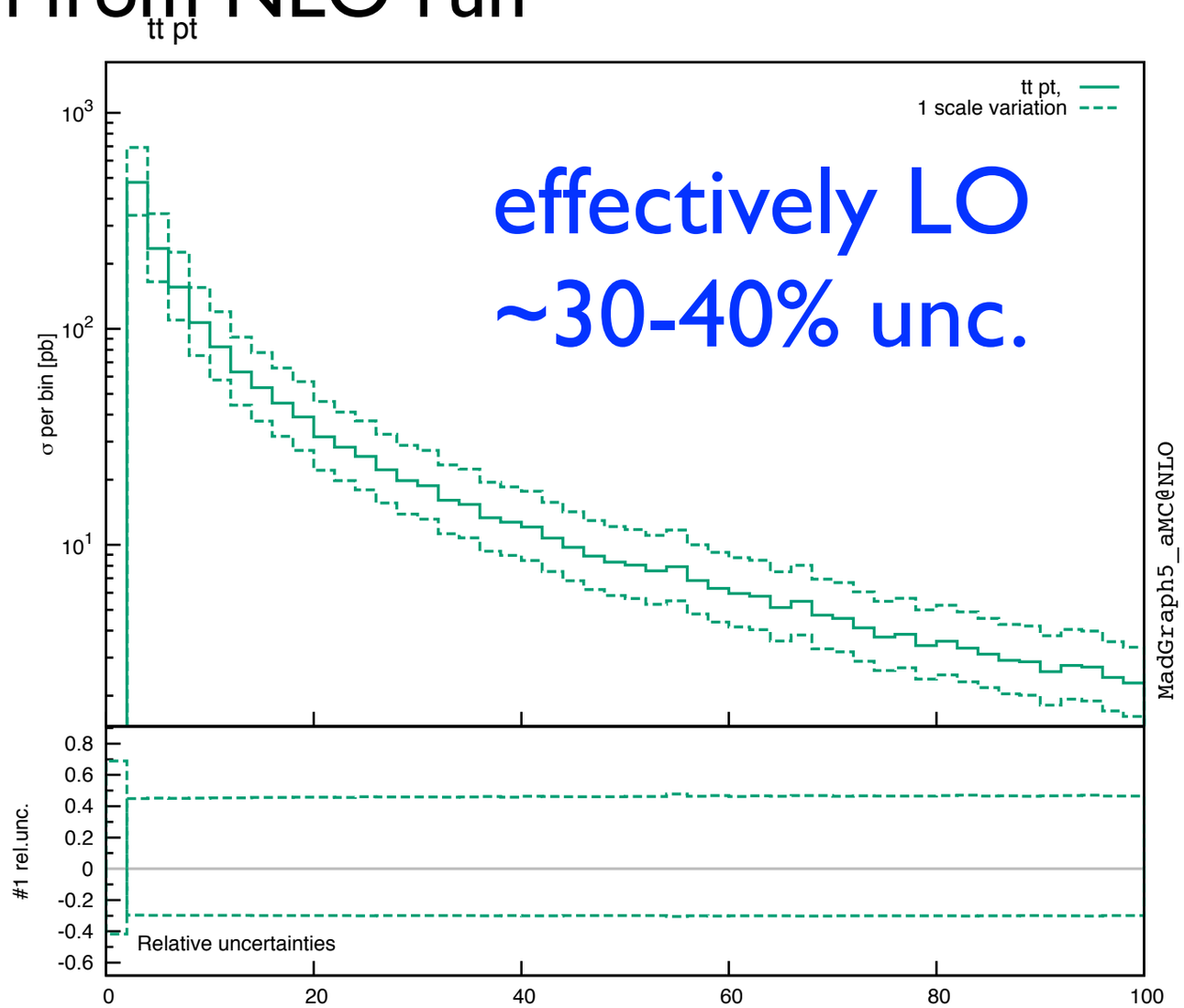
Scale uncertainties





Scale uncertainties

$p_T(t\bar{t})$ histogram from NLO run



NLO exercise
$t\bar{t}$ production at NLO

Part 2

- Generate a NLO event sample to be showered by Pythia6Q or Pythia8
- Shower and analyse it with the `py68an_HwU_pp_ttx.o` analysis (to be specified in the `shower_card`)
- The histogramming routine (`HwU.o`) must also be added to the analysis files in the `shower_card` (Hint: you can shower an existing run with `./bin/shower run_xx`)
- Use MadSpin to generate a di-leptonic (into muons) decayed sample
- Re-analyse the decayed and un-decayed sample with the `py68an_HwU_pp_lplm.o` analysis and check the lepton pair p_T
  - If using Pythia6Q, the analysis (in `MCatNLO/PYAnalyzer/py6an_HwU_pp_lplm.f`) has to be slightly modified:
    - `IORI.LE.10` → `IORI.LE.20` at lines 188, 192
  - To tell Pythia to perform di-leptonic decays, add these lines in the `shower_card` ('Decay channels' block; antiparticles are decayed as particles)
    - ```
      DM_1 = 6 > 24 5 @1d0 @100
      DM_2 = 24 > 14 -13 @1d0 @100
      ```
    - ```
      DM_1 = 24:onMode = off
      DM_2 = 24:onIfAny = 13 14
      ```

Using Pythia8 for NLO runs

- Before generating the process, set the path via the MG5_aMC interface
  `> set pythia8_path /path/to/pythia8`
- If the process has already been generated, set the path inside `Cards/amcatnlo_configuration.txt`
- Remember to set the parton shower in the `run_card`!

NLO exercise
Solution

Part 2

- Generate a NLO event sample to be showered by Pythia6Q
  - Shower it with the `mcatnlo_pyan_pp_ttx` analysis (to be specified in the `shower_card`)
    - `cd my_ttbar_nlo`
    - `./bin/aMCatNLO`
    - `> launch`
    - `> fixed_order=OFF`
    - `> shower=ON`
    - Edit `run_card`
    - Edit `shower_card`

NLO exercise
Solution

Part 2

- Generate a NLO event sa
  - Shower it with the mca cified in the shower_card)
    - cd my_ttbar_nlo
    - ./bin/aMCatNLO
    - > launch
    - > fixed_order=OF
    - > shower=ON
  - Edit run_card
  - Edit shower_card

```
# PDF choice: this automatically fixes also alpha_s(MZ) and its evol.
*
#*********************************************************************
*
 nn23nlo    = pdlabel    ! PDF set
 244600     = lhaid      ! if pdlabel=lhapdf, this is the lhapdf number
#*********************************************************************
*
# Include the NLO Monte Carlo subtr. terms for the following parton
*
# shower (HERWIG6 | HERWIGPP | PYTHIA6Q | PYTHIA6PT | PYTHIA8)
*
# WARNING: PYTHIA6PT works only for processes without FSR!!!!
*
#*********************************************************************
*
  PYTHIA6Q    = parton_shower
#*********************************************************************
*
# Renormalization and factorization scales
*
# (Default functional form for the non-fixed scales is the sum of
*
# the transverse masses of all final state particles and partons. This
*
# can be changed in SubProcesses/set_scales.f)
*
#*********************************************************************
*
 F         = fixed_ren_scale  ! if .true. use fixed ren scale
 F         = fixed_fac_scale  ! if .true. use fixed fac scale
 91.188    = muR_ref_fixed    ! fixed ren reference scale
 91.188    = muF1_ref_fixed   ! fixed fact reference scale for pdf1
 91.188    = muF2_ref_fixed   ! fixed fact reference scale for pdf2
#*********************************************************************
*
# Renormalization and factorization scales (advanced and NLO options)
*
#*********************************************************************
*
```

NLO exercise
Solution

Part 2

- Generate a NLO event sa
  - Shower it with the mca cified in the shower_card)
    - cd my_ttbar_nlo
    - ./bin/aMCatNLO
    - > launch
    - > fixed_order=OF
    - > shower=ON
  - Edit run_card
  - Edit shower_card

```
# PDF choice: this automatically fixes also alpha_s(MZ) and its evol.
*
#*********************************************************************
*
 nn23nlo    = pdlabel    ! PDF set
 244600     = lhaid      ! if pdlabel=lhapdf, this is the lhapdf number
#*********************************************************************
*
# Include the NLO Monte Carlo subtr. terms for the following parton
*
# shower (HERWIG6 | HERWIGPP | PYTHIA6Q | PYTHIA6PT | PYTHIA8)
*
# WARNING: PYTHIA6PT works only for processes without FSR!!!!
*
#*********************************************************************
*
  PYTHIA6Q    = parton_shower
#*********************************************************************
*
# Renormalization and factorization scales
*
# (
*
# t
*
# c
*
#**
*
 F
 F
 91
 91
 91
#**
*
# R
*
#*********************************************************************
*
```

```
#*********************************************************************
# Extra Libraries/analyses                                            *
# The following lines need to be changed if the user does not want to *
# create a StdHEP/HepMC file, but to directly run an own analysis (to *
# be placed in HWAnalyzer or analogous MCatNLO subfolders).           *
# Please use files in those folders as examples.                      *
#*********************************************************************
EXTRALIBS     =      # Extra-libraries (not LHAPDF)
                                  # Default: "stdhep Fmcfio"
                                  # PYTHIA > 8.200 may require library dl
EXTRAPATHS    = ../lib            # Path to the extra-libraries
                                  # Default: "../lib"
INCLUDEPATHS =                    # Path to header files needed by c++
                                  # Dir names separated by white spaces
ANALYSE       = pyan_HwU_pp_ttx.o HwU.o
                                  # routines (please use .o as extension
                                  # and use spaces to separate files)
```

NLO exercise
Solution

Part 2

- Generate a NLO event s
  - Shower it with the mca ne shower_card)
    - cd my_ttbar_nlo
    - ./bin/aMCatNLO
    - > launch
    - > fixed_order=O
    - > shower=ON
  - Edit run_card
  - Edit shower_card

```
# PDF choice: this automatically fixes also alpha_s(MZ) and its evol.
```

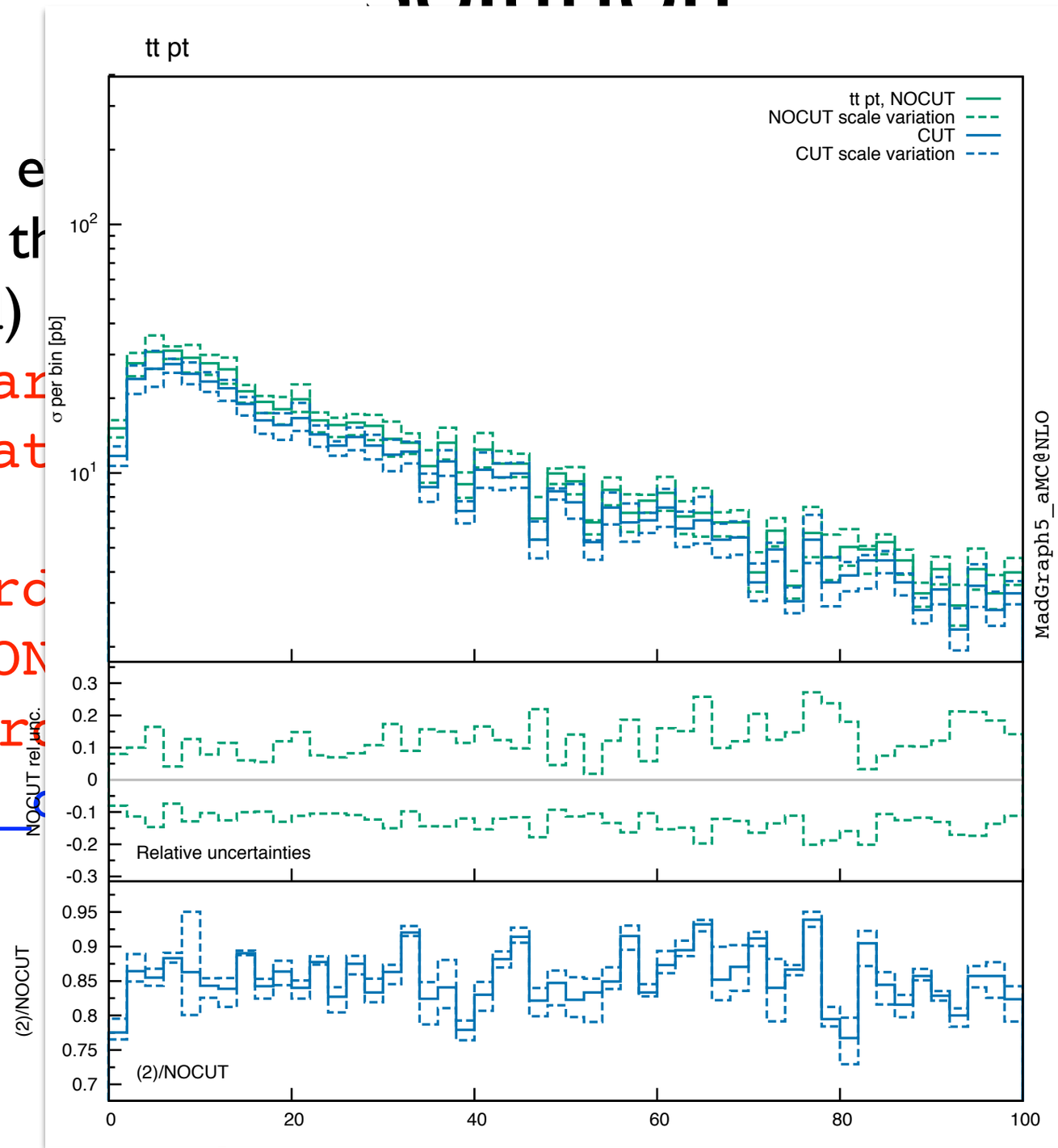
```
Summary:
Process p p > t t~ [QCD]
Run at p-p collider (6500 + 6500 GeV)
Total cross-section: 6.772e+02 +- 2.1e+00 pb
Ren. and fac. scale uncertainty: +11.5% -13.0%
Number of events generated: 100000
Parton shower to be used: PYTHIA6Q
Fraction of negative weights: 0.20
Total running time :  6m 58s

INFO: The /Users/marcozaro/Physics/MadGraph/2.2.3new/my_tt_nlo_qcd/Events/
run_12/events.lhe.gz file has been generated.
. . .
INFO: Preparing MCatNLO run
INFO: Compiling MCatNLO for PYTHIA6Q...
INFO:                          ... done
INFO: Showering events...
INFO: (Running in /Users/marcozaro/Physics/MadGraph/2.2.3new/my_tt_nlo_qcd/
MCatNLO/RUN_PYTHIA6Q_3)
INFO:  Idle: 0,  Running: 1,  Completed: 0 [ current time: 12h32 ]
INFO:  Idle: 0,  Running: 0,  Completed: 1 [  2m 35s  ]
INFO:  Idle: 0,  Running: 0,  Completed: 0 [ current time: 12h34 ]
INFO: The file /Users/marcozaro/Physics/MadGraph/2.3.1/ttbar/Events/run_01/
plot_PYTHIA6Q_1_0.HwU has been generated, with histograms in the HwU and
GnuPlot formats, obtained by showering the parton-level file /Users/
marcozaro/Physics/MadGraph/2.3.1/ttbar/Events/run_01/events.lhe.gz with
PYTHIA6Q.
INFO: Run complete
```

```
# R                                  # routines (please use .o as extension
*                                    # and use spaces to separate files)
#*********************************************************************
*
```

NLO exercise
Solution

Part 2

- Generate a NLO e
  - Shower it with th ne shower_card)
    - `cd my_ttbar`
    - `./bin/aMCat`
    - `> launch`
    - `> fixed_orc`
    - `> shower=ON`
  - Edit `run_car`
  - Edit `shower_c`




```
ts evol.
y_tt_nlo_qcd/Events/
2.3new/my_tt_nlo_qcd/
ime: 12h32 ]
]
ime: 12h34 ]
/ttbar/Events/run_01/
ams in the HwU and
l file /Users/
vents.lhe.gz with
rate files)
**********
```

The events

```
<initrwgt>
  <weightgroup type='scale_variation' combine='envelope'>
    <weight id='1001'> muR=0.10000E+01 muF=0.10000E+01 </weight>
    <weight id='1002'> muR=0.10000E+01 muF=0.20000E+01 </weight>
    <weight id='1003'> muR=0.10000E+01 muF=0.50000E+00 </weight>
    <weight id='1004'> muR=0.20000E+01 muF=0.10000E+01 </weight>
    <weight id='1005'> muR=0.20000E+01 muF=0.20000E+01 </weight>
    <weight id='1006'> muR=0.20000E+01 muF=0.50000E+00 </weight>
    <weight id='1007'> muR=0.50000E+00 muF=0.10000E+01 </weight>
    <weight id='1008'> muR=0.50000E+00 muF=0.20000E+01 </weight>
    <weight id='1009'> muR=0.50000E+00 muF=0.50000E+00 </weight>
  </weightgroup>
</initrwgt>
</header>
<init>
  2212   2212 0.65000000E+04 0.65000000E+04 -1 -1 244600 244600 -4   1
 0.68147533E+03 0.22760274E+01 0.11811897E+04      0
  </init>
  <event>
  4      0 -.11811897E+04 0.68991465E+03 0.75467716E-02 0.11800000E+00
       21 -1    0    0  501  502 0.00000000E+00 0.00000000E+00 0.16695776E+03 0.16695776E+03 0.00000000E+00 0.0000E+00 0.9000E+01
       21 -1    0    0  502  503 -.00000000E+00 -.00000000E+00 -.83539498E+03 0.83539498E+03 0.00000000E+00 0.0000E+00 0.9000E+01
        6  1    1    2  501    0 -.87405313E+02 -.30435858E+03 -.46344397E+03 0.58735266E+03 0.17300000E+03 0.0000E+00 0.9000E+01
       -6  1    1    2    0  503 0.87405313E+02 0.30435858E+03 -.20499324E+03 0.41500008E+03 0.17300000E+03 0.0000E+00 0.9000E+01
#aMCatNLO 1  5  3  3  2 0.21343976E+03 0.35860250E+02 9  0  0 0.10000001E+01 0.15353083E+01 0.66887201E+00 0.00E+00 0.0E+00
  <rwgt>
   <wgt id='1001'> -.11812E+04 </wgt>
   <wgt id='1002'> -.10571E+04 </wgt>
   <wgt id='1003'> -.13263E+04 </wgt>
   <wgt id='1004'> -.88285E+03 </wgt>
   <wgt id='1005'> -.79006E+03 </wgt>
   <wgt id='1006'> -.99128E+03 </wgt>
   <wgt id='1007'> -.16151E+04 </wgt>
   <wgt id='1008'> -.14453E+04 </wgt>
   <wgt id='1009'> -.18135E+04 </wgt>
  </rwgt>
  </event>
```

- Each event keeps information about scale variations
- To obtain scale uncertainties use the extra weights to fill histograms and take the envelope

NLO exercise
Solution

Part 2

- Use MadSpin to generate a di-leptonic (into muons) decayed sample
  - `./bin/aMCatNLO`
  - `> decay_events run_xx`
  - edit the `madspin_card`

NLO exercise
Solution

Part 2

- Use MadSpin to generate a di-leptonic (into muons) decayed sample
  - `./bin/aMCatNLO`
  - `> decay_events run_xx`
  - edit the `madspin_card`

```
#*********************************************************
#*                        MadSpin                        *
#*                                                       *
#*    P. Artoisenet, R. Frederix, R. Rietkerk, O. Mattelaer *
#*                                                       *
#*    Part of the MadGraph5_aMC@NLO Framework:           *
#*    The MadGraph5_aMC@NLO Development Team - Find us at *
#*    https://server06.fynu.ucl.ac.be/projects/madgraph  *
#*                                                       *
#*********************************************************
#Some options (uncomment to apply)
#
# set seed 1
# set Nevents_for_max_weigth 75 # number of events for the estimate of the max. weight
# set BW_cut 15                # cut on how far the particle can be off-shell
 set max_weight_ps_point 400  # number of PS to estimate the maximum for each event
#
# specify the decay for the final state particles
decay t > w+ b, w+ > mu+ vm
decay t~ > w- b~, w- > mu- vm~
# running the actual code
launch
~
```

NLO exercise
Solution

Part 2

- Use MadSpin to generat
  - `./bin/aMCatNLO`
  - `> decay_events r`
  - edit the `madspin_ca`

```
#*********************************
#*                      MadSpin
#*
#*    P. Artoisenet, R. Frederix,
#*
#*    Part of the MadGraph5_aMC@N
#*    The MadGraph5_aMC@NLO Devel
#*    https://server06.fynu.ucl.a
#*
#*********************************
#Some options (uncomment to apply
#
# set seed 1
# set Nevents_for_max_weigth 75 #
# set BW_cut 15                 #
 set max_weight_ps_point 400  # number of PS to estimate the maximum for each event
#
# specify the decay for the final state particles
decay t > w+ b, w+ > mu+ vm
decay t~ > w- b~, w- > mu- vm~
# running the actual code
launch
~
```

```
 . . .
INFO: MadSpin: Estimate the maximum weight
INFO:
INFO:    Estimating the maximum weight
INFO:    ****************************
INFO:      Probing the first 139 events
INFO:      with 400 phase space points
INFO:
INFO: Event 1/139 :  0.059s
INFO: Event 6/139 :  0.99s
INFO: Event 11/139 :  1.3s
 . . .
INFO: Decaying the events...
INFO: Event nb 1000  2.6s
INFO: Event nb 2000  4.9s
INFO: Event nb 3000  7s
 . . .
INFO: Decayed events have been written in /Users/marcozaro/Physics/
MadGraph/2.2.3new/my_tt_nlo_qcd/Events/run_01/events_decayed.lhe.gz
INFO: The decayed event file has been moved to the following location:
INFO: /Users/marcozaro/Physics/MadGraph/2.2.3new/my_tt_nlo_qcd/Events/
run_01_decayed_1/events.lhe.gz
INFO: MadSpin Done
```

MS estimates

$$\max\left(|\mathrm{M_{P+D}}|^2 / |\mathrm{M_P}|^2\right)$$

with the first events

NLO exercise
Solution

Part 2

- Re-analyse the decayed and un-decayed sample with the `py6an_HwU_pp_lplm` analysis and check the the lepton pair p_T
  - Re-shower the un-decayed sample
  - `./bin/shower run_xx`
  - edit the `shower_card`
  - Shower the decayed sample
  - `./bin/shower run_xx_decayed_1`

NLO exercise
Solution

Part 2

- Re-analyse the decayed and un-decayed sample with the `py6an_HwU_pp_lplm` analysis and check the the lepton pair p_T
  - Re-shower the un-decayed sample
  - `./bin/shower run_xx`
  - edit the `shower_card`
  - Shower the decayed sample
  - `./bin/shower run_xx_de`

```
# Decay channels                                                        *
# Write down the decay channels for the resonances, to be performed by *
# the shower.                                                           *
. . .
DM_1 = 6 > 24 5 @1d0 @100
DM_2 = -6 > -24 -5 @1d0 @100
DM_3 = 24 > 14 -13 @1d0 @100
DM_4 = -24 > -14 13 @1d0 @100
#***********************************************************************
#***********************************************************************
# Extra Libraries/analyses                                              *
# The following lines need to be changed if the user does not want to  *
# create a StdHEP/HepMC file, but to directly run an own analysis (to  *
# be placed in HWAnalyzer or analogous MCatNLO subfolders).             *
# Please use files in those folders as examples.                        *
#***********************************************************************
EXTRALIBS    =      # Extra-libraries (not LHAPDF)
                                  # Default: "stdhep Fmcfio"
                                  # PYTHIA > 8.200 may require library dl
EXTRAPATHS   = ../lib             # Path to the extra-libraries
                                  # Default: "../lib"
INCLUDEPATHS =                    # Path to header files needed by c++
                                  # Dir names separated by white spaces
ANALYSE      = mcatnlo_pyan_pp_lplm.o HwU.o
                                  # routines (please use .o as extension
                                  # and use spaces to separate files)
```

NLO exercise
Solution

Part 2

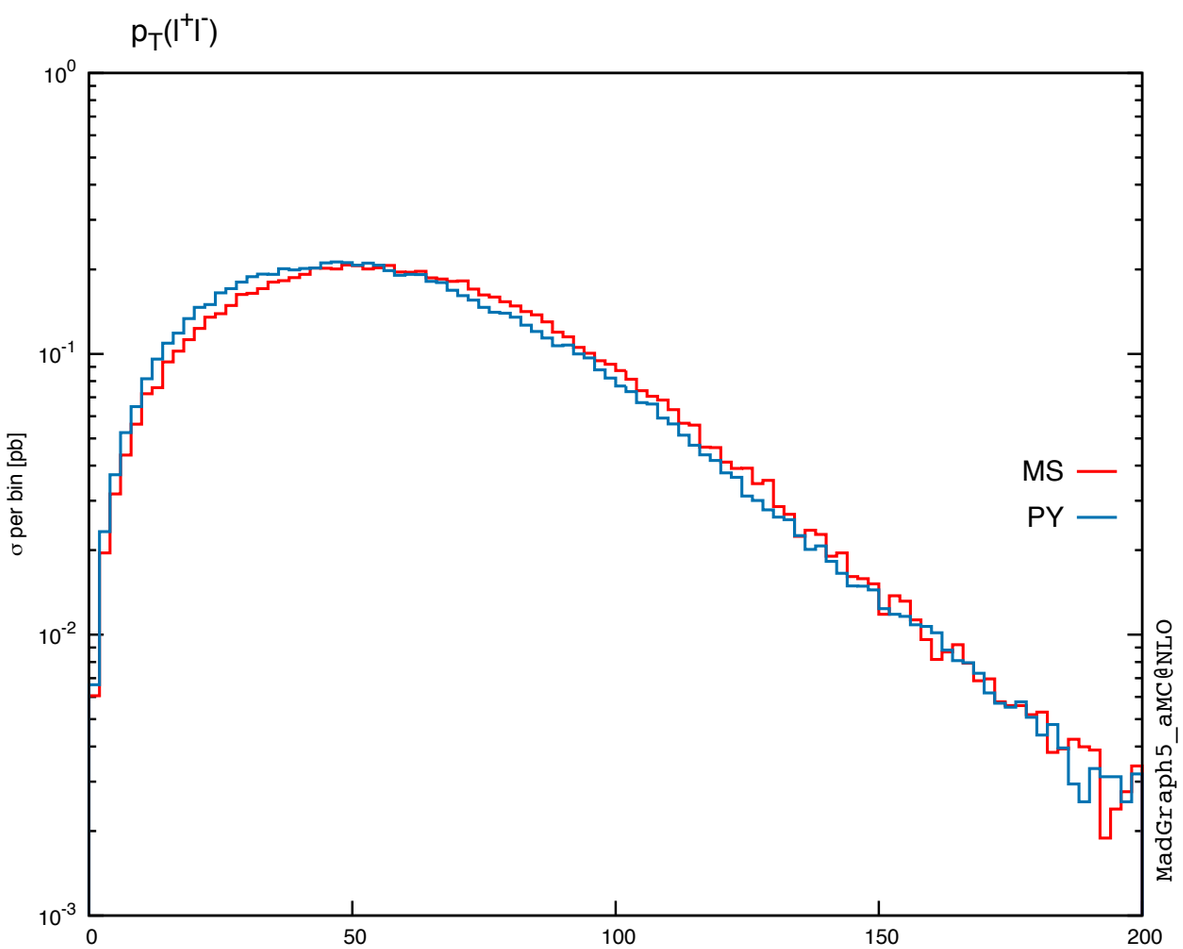
- Re-analyse the decayed and un-decayed sample with the `py6an_HwU_pp_lplm` analysis and check the the lepton pair p_T
  - Re-shower the un-decayed sample
  - `./bin/sh`
  - edit the `sh`
  - Shower the
  - `./bin/sh`

```
INFO: Preparing MCatNLO run
INFO: Compiling MCatNLO for PYTHIA6Q...
INFO:                          ... done
INFO: Showering events...
INFO: (Running in /Users/marcozaro/Physics/MadGraph/2.2.3new/my_tt_nlo_qcd/
MCatNLO/RUN_PYTHIA6Q_3)
INFO:  Idle: 0,  Running: 1,  Completed: 0 [ current time: 12h32 ]
INFO:  Idle: 0,  Running: 0,  Completed: 1 [  2m 35s  ]
INFO:  Idle: 0,  Running: 0,  Completed: 0 [ current time: 12h34 ]
INFO: The file /Users/marcozaro/Physics/MadGraph/2.3.1/ttbar/Events/run_01/
plot_PYTHIA6Q_2_0.HwU has been generated, with histograms in the HwU and
GnuPlot formats, obtained by showing the parton-level file /Users/
marcozaro/Physics/MadGraph/2.3.1/ttbar/Events/run_01/events.lhe.gz with
PYTHIA6Q.
INFO: Run complete
. . .
INFO:  Idle: 0,  Running: 1,  Completed: 0 [ current time: 12h32 ]
INFO:  Idle: 0,  Running: 0,  Completed: 1 [  2m 35s  ]
INFO:  Idle: 0,  Running: 0,  Completed: 0 [ current time: 12h34 ]
INFO: INFO: The file /Users/marcozaro/Physics/MadGraph/2.3.1/ttbar/Events/
run_01_decayed_1/plot_PYTHIA6Q_1_0.HwU has been generated, with histograms in
the HwU and GnuPlot formats, obtained by showing the parton-level file /
Users/marcozaro/Physics/MadGraph/2.3.1/ttbar/Events/run_01_decayed_1/
events.lhe.gz with PYTHIA6Q.
INFO: Run complete
```

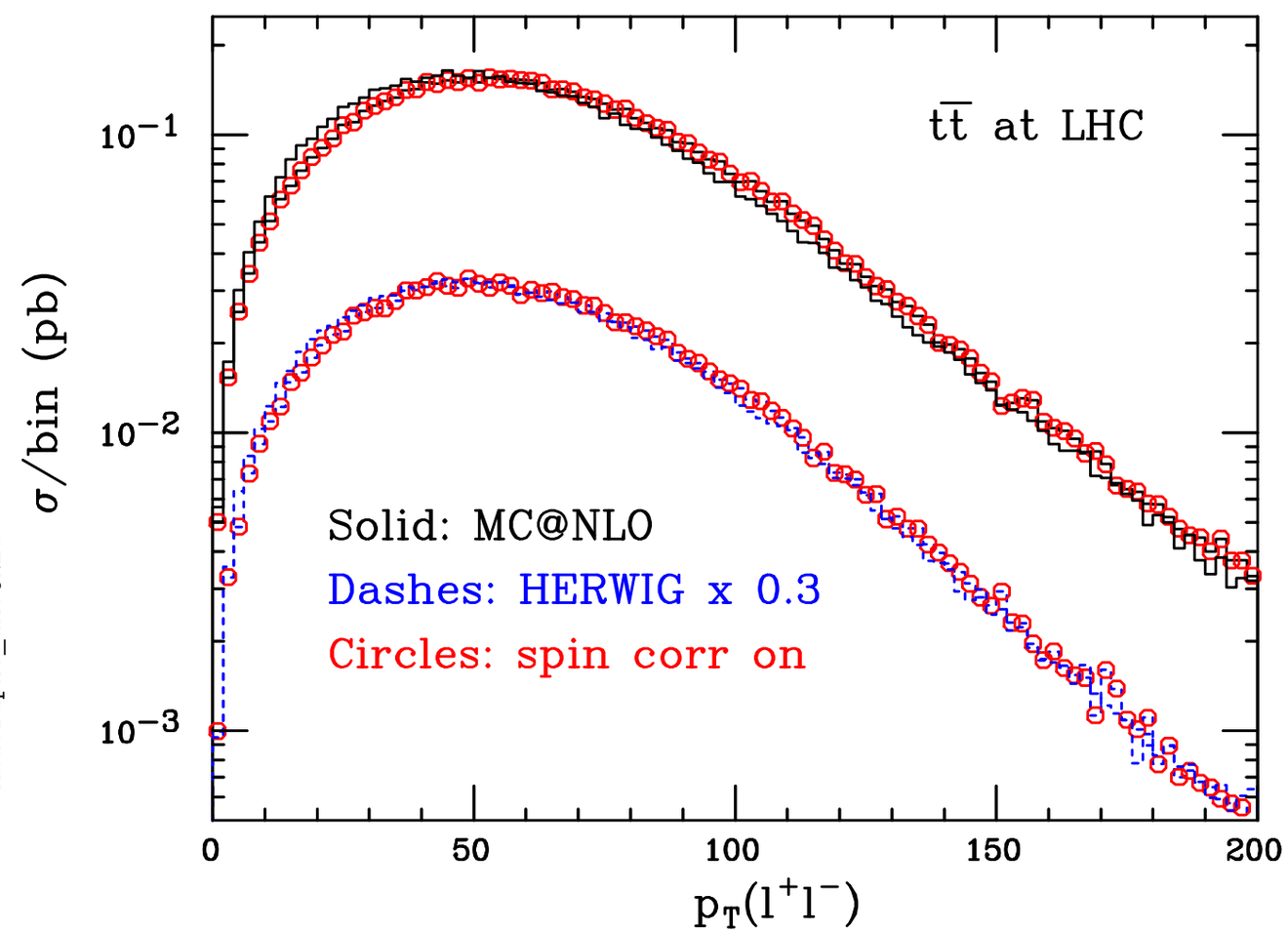
NLO exercise
Solution

Part 2

- Re-analyse the decayed and undecayed sample with the `mcatnlo_pyan_pp_lplm` analysis and check the the lepton pair p_T



Plot from Frixione, Leanen, Motylinski, Webber, arXiv:hep-ph/0702198



Part 3:
Generating large event samples

Generating large event samples: How theorists do

- Set the nevents entry in the `run_card` to the desired number of events (`5000000 = nevents`)
- Launch event generation (`./bin/generate_events`)
- Wait…
- What happens?
  - The integrator tries to find an optimal grid for each integration channel, with an accuracy $\sim 1/\sqrt{N_{\text{events}}}$.
  - Each integration channel is assigned a number of events proportional to its weight
  - Events are generated…

Generating large event samples: How theorists do

- Set the nevents entry in the `run_card` to the desired number of events (`5000000 = nevents`)
- Launch event generation (`./bin/generate_events`)
- Wait
- Wha
  - The integration channel, with an accuracy $\sim 1/\sqrt{N_{\text{events}}}$.
  - Each integration channel is assigned a number of events proportional to its weight
  - Events are generated…

Example: ttbar production at NLO

```
P0_gg_ttx/GF2/events.lhe          2035804    4.659965e+02    1.000000e+00
P0_gg_ttx/GF3/events.lhe          2029525    4.636892e+02    1.000000e+00
P0_gg_ttx/GF1/events.lhe          458422     1.049532e+02    1.000000e+00
P0_uxu_ttx/GF1/events.lhe         238742     5.447329e+01    1.000000e+00
P0_uux_ttx/GF1/events.lhe         237507     5.437405e+01    1.000000e+00
```

Drawbacks

- It takes time
- It only uses a limited number of CPUs (~ integration channels)
- If one integration channel crashes, the full event sample has to be regenerated

Fixes #1:

Fixes #1:

- It only uses a limited number of CPUs (~ integration channels)
  - For NLO runs, one can use the nevt_job variable in the run_card to set the maximum number of events for each job. Channels with large number of event will be split into more jobs

Fixes #1:

- It onl ation
  chann
  - For he
    run_ each job.
    Cha more
    jobs

Example: ttbar production at NLO
with nevt_job = 100000

P0_gg_ttx/GF1_1/events.lhe	92434	2.124127e+01	2.000017e-01
P0_gg_ttx/GF1_2/events.lhe	92433	2.124104e+01	1.999996e-01
P0_gg_ttx/GF1_3/events.lhe	92433	2.124104e+01	1.999996e-01
P0_gg_ttx/GF1_4/events.lhe	92433	2.124104e+01	1.999996e-01
P0_gg_ttx/GF1_5/events.lhe	92433	2.124104e+01	1.999996e-01
P0_gg_ttx/GF2_1/events.lhe	96676	2.218363e+01	4.761914e-02
P0_gg_ttx/GF2_2/events.lhe	96676	2.218363e+01	4.761914e-02
P0_gg_ttx/GF2_3/events.lhe	96676	2.218363e+01	4.761914e-02
P0_gg_ttx/GF2_4/events.lhe	96676	2.218363e+01	4.761914e-02

. . . 53 channels in total . . .

P0_gg_ttx/GF3_19/events.lhe	96748	2.221255e+01	4.761872e-02
P0_gg_ttx/GF3_20/events.lhe	96748	2.221255e+01	4.761872e-02
P0_gg_ttx/GF3_21/events.lhe	96748	2.221255e+01	4.761872e-02
P0_uux_ttx/GF1_1/events.lhe	79622	1.822114e+01	3.333333e-01
P0_uux_ttx/GF1_2/events.lhe	79622	1.822114e+01	3.333333e-01
P0_uux_ttx/GF1_3/events.lhe	79622	1.822114e+01	3.333333e-01
P0_uxu_ttx/GF1_1/events.lhe	79018	1.813615e+01	3.333333e-01
P0_uxu_ttx/GF1_2/events.lhe	79018	1.813615e+01	3.333333e-01
P0_uxu_ttx/GF1_3/events.lhe	79018	1.813615e+01	3.333333e-01

Fixes #1:

- It onl ... ation chann
  - For ... he run_ ... each job. Cha ... o more jobs

Example: ttbar production at NLO
with nevt_job = 100000

P0_gg_ttx/GF1_1/events.lhe	92434	2.124127e+01	2.000017e-01
P0_gg_ttx/GF1_2/events.lhe	92433	2.124104e+01	1.999996e-01
P0_gg_ttx/GF1_3/events.lhe	92433	2.124104e+01	1.999996e-01
P0_gg_ttx/GF1_4/events.lhe	92433	2.124104e+01	1.999996e-01
P0_gg_ttx/GF1_5/events.lhe	92433	2.124104e+01	1.999996e-01
P0_gg_ttx/GF2_1/events.lhe	96676	2.218363e+01	4.761914e-02
P0_gg_ttx/GF2_2/events.lhe	96676	2.218363e+01	4.761914e-02
P0_gg_ttx/GF2_3/events.lhe	96676	2.218363e+01	4.761914e-02
P0_gg_ttx/GF2_4/events.lhe	96676	2.218363e+01	4.761914e-02

. . . 53 channels in total . . .

P0_gg_ttx/GF3_19/events.lhe	96748	2.221255e+01	4.761872e-02
P0_gg_ttx/GF3_20/events.lhe	96748	2.221255e+01	4.761872e-02
P0_gg_ttx/GF3_21/events.lhe	96748	2.221255e+01	4.761872e-02
P0_uux_ttx/GF1_1/events.lhe	79622	1.822114e+01	3.333333e-01
P0_uux_ttx/GF1_2/events.lhe	79622	1.822114e+01	3.333333e-01
P0_uux_ttx/GF1_3/events.lhe	79622	1.822114e+01	3.333333e-01
P0_uxu_ttx/GF1_1/events.lhe	79018	1.813615e+01	3.333333e-01
P0_uxu_ttx/GF1_2/events.lhe	79018	1.813615e+01	3.333333e-01
P0_uxu_ttx/GF1_3/events.lhe	79018	1.813615e+01	3.333333e-01

- But still,
  If one integration channel crashes, the full event sample has to be regenerated

Fixes #2:

- If one integration channel crashes, the full event sample has to be regenerated
  - Do more runs with a smaller number of events, and combine them at the end
  - Warning! The grid's accuracy has still to match the final number of events, but this has to be done only once

LO best practice: `multi-run` and `gridpack`

- `multi_run`: run N times sequentially the event generation for $N_{\rm tot}/N$ events each time (updating the random seed), then combine the various samples
  `./bin/madevent multi-run NRUNS (RUN_NAME)`
- All samples are statistically equivalent
  If a failure occurs, one can still combine those which succeeded
- Drawbacks:
  - Runs sequentially
  - The grid is computed before each event generation

LO best practice: multi-run and gridpack

more on https://cp3.irmp.ucl.ac.be/projects/madgraph/wiki/GridDevelopment

- gridpack: set the grids only once, then generate many small event samples in parallel
- Set True = gridpack in the run_card, then execute ./bin/generate_events
- Grids will be setup (Running survey) and a tarball will be created → run_01_gridpack.tar.gz
- Inside the tarball you have a run script and the madevent directory
- Untar it, compile inside the madevent dir (./bin/compile) and clean unnecessary files (./bin/clean4grid), then re-tar and send over all CPUs
- run takes two arguments, the number of events and the seed. E.g. ./run 10000 37

NLO best practice

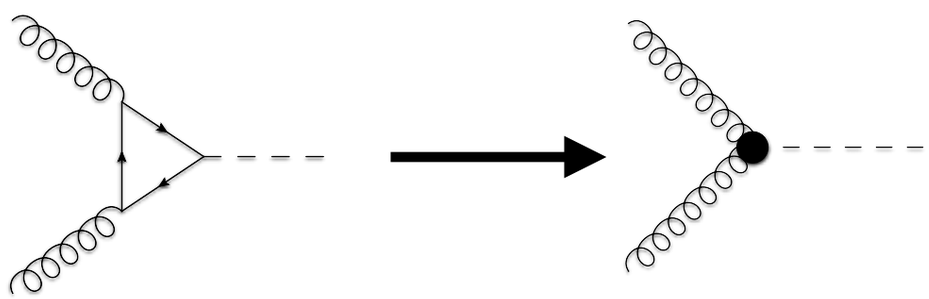
- Similarly to the `gridpack`, one can first set the integration grid, then generate many small event samples in parallel
- How to?
  - The `req_acc` variable in the `run_card` can be set to the desired accuracy independently of the required number of events. For example one can set `0.001 = req_acc` and `500 = nevents`.
  - Once the grids are set (with the usual ./bin/generate_events command) one can tar the directory and send over the cluster nodes, where only the event generation can be run with `./bin/generate_events --only-generation --nocompile`
  - Remember! Update the random seed (`iseed` in the `run_card` or directly in the `SubProcesses/randinit` file)

Part 4:
Loop induced processes

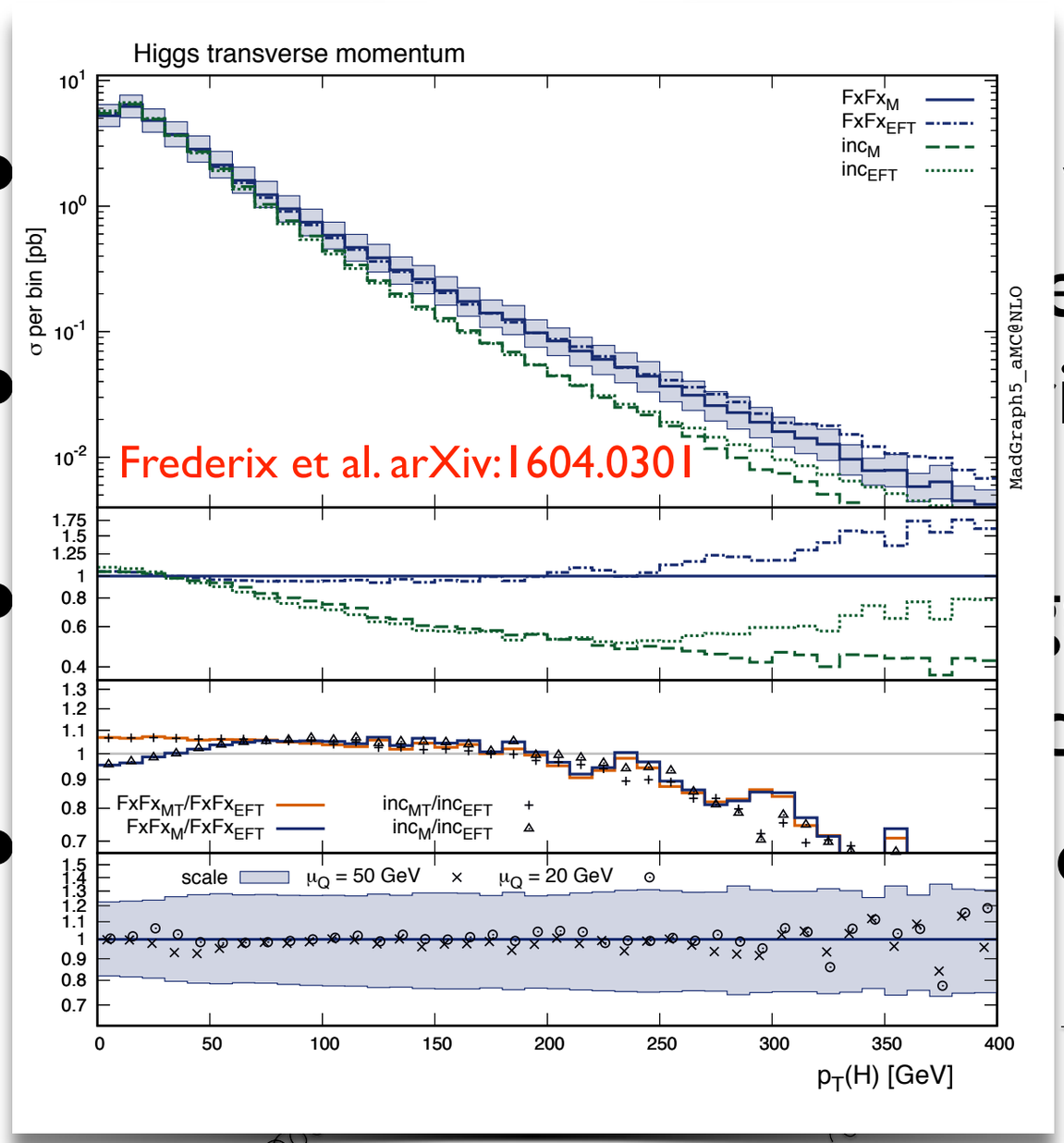
Loop-induced processes

Loop-induced processes

- There are some processes which do not receive contributions from any tree-level diagrams
- Even at LO, diagrams contributing to these processes feature loops
- Notable examples are Higgs production in gluon fusion, $gg \to HZ$, $gg \to HH$ and the decay $H \to \gamma\gamma$
- Sometimes an effective-theory approach can help…



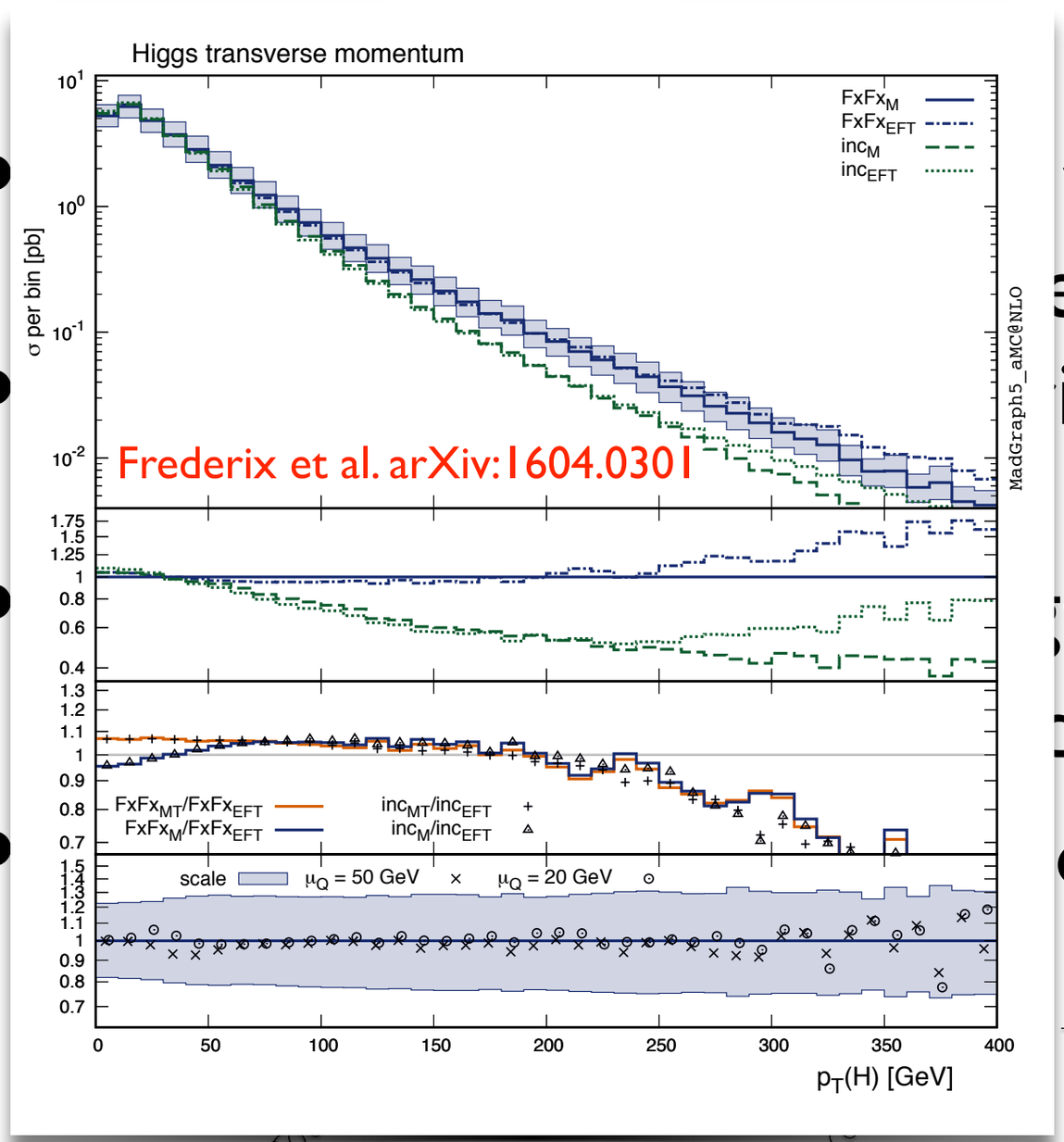
Loop-induced processes



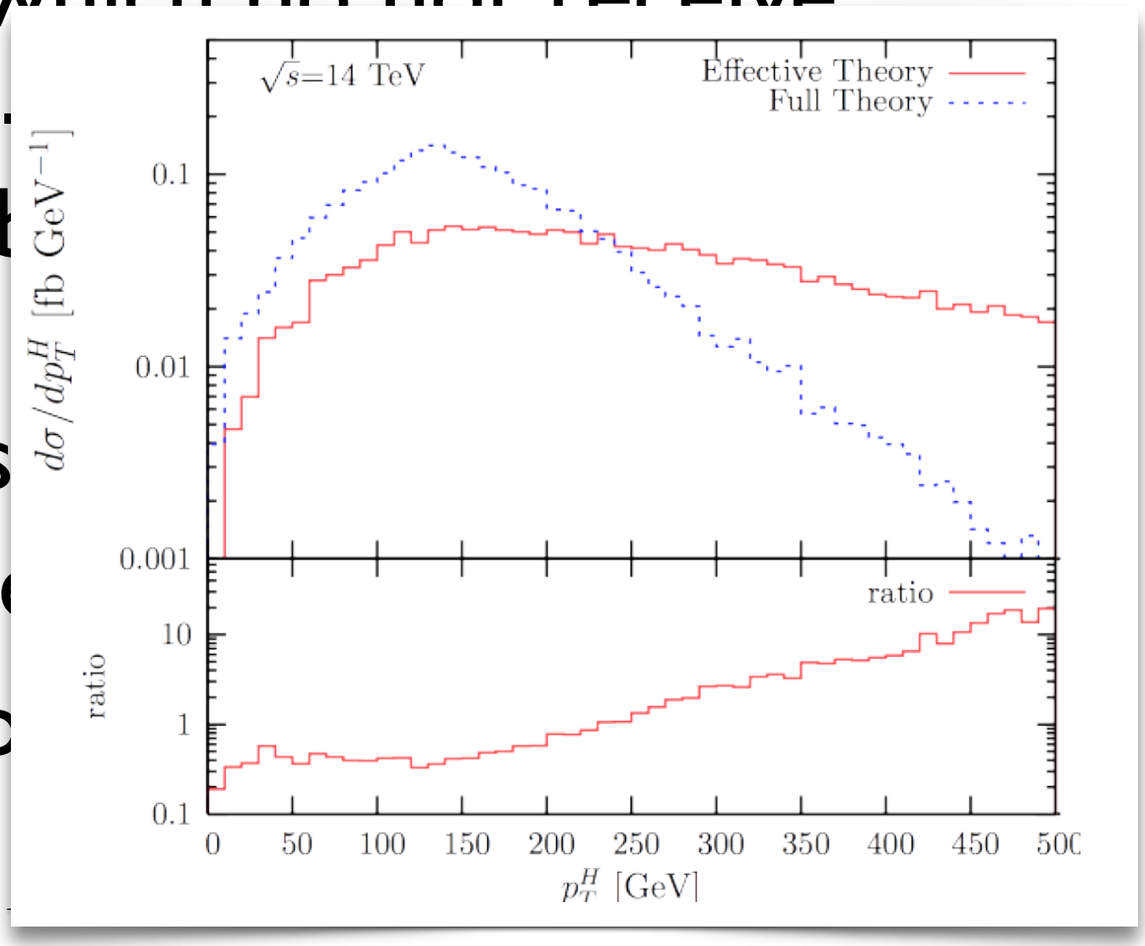
- which do not receive
- e-level diagrams
- ibuting to these processes

- s production in gluon fusion,
- decay $H \to \gamma\gamma$
- ory approach can help…

Loop-induced processes



- which do not receive
- e.
- ib
- s
- d
- c



- …but sometimes it dramatically fails

Loop-induced processes

- Since one year ago, the event generation for loop-induced processes has the same level of automation of any LO process with MadEvent Hirschi & Mattelaer, arXiv:1507.00020
- It can be as easy as

```
> generate p p > h h [QCD]
> output My_loop_induced_hh
> launch
```

- Note that here `[QCD]` does not mean *'do NLO QCD corrections'*, rather *'include loops with QCD particles'*. The code switches to loop-induced since there are no tree-level diagrams

Loop-induced exercise

- Generate events for Higgs production in gluon fusion, both in the loop-induced and in the effective theory (use the `heft` model in that case)
- Are bottom-quark loops included in the loop-induced? How can I disable them?
- Run `pythia` and `MadAnalysis` on the event samples in order to obtain plots (you can also use your favourite plotting routine to plot the events after `pythia`)
- Compare the Higgs p_T plot with t/b, without b and in the EFT and comment

Pythia and MadAnalysis

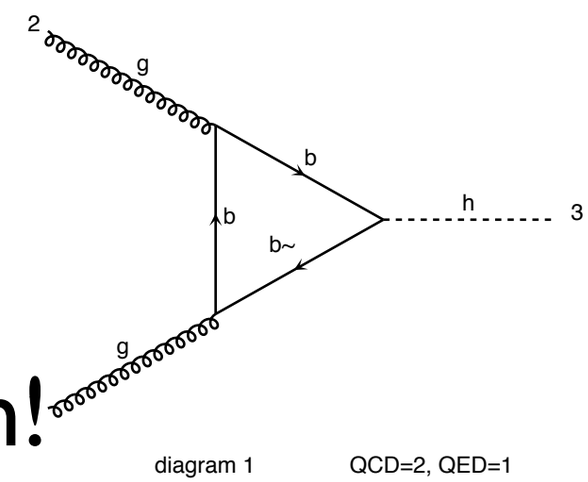
- Install `pythia` and `MadAnalysis` via the `MG5_aMC` shell
  ```
  > install pythia-pgs
  > install MadAnalysis
  ```
  BEFORE having generated the process folder (otherwise uncomment the `pythia-pgs_path` and `madanalysis_path` inside `Cards/me5_configuration.txt`)
- You can choose to run Pythia together with the event generation or in a second stage with
  ```
  ./bin/madevent
  > pythia run_XXX
  ```
- To plot the (decayed) Higgs set "`plot_decayed yes`" in the `plot_card`
- The p_T range in the plots can be changed from the `plot_card`
  ```
  # Put here the plot ranges
  #*********************************************************************
  # Do NOT put spaces at the beginning of the following lines!
  # Begin PlotRange # This is TAG. Do not modify this line
  pt        4    0 200   # bin size, min value, max value
  ```

Loop-induced exercise: Solution

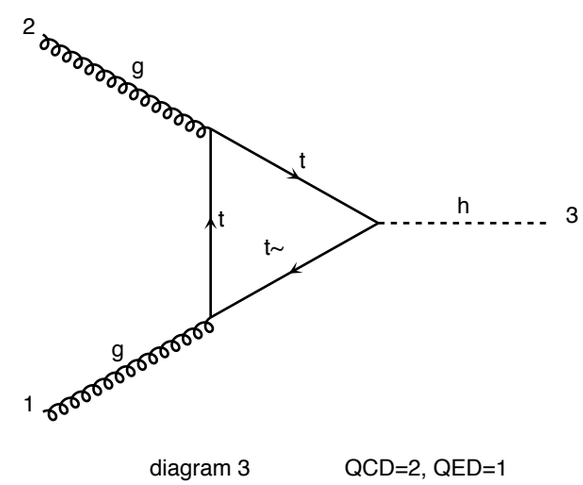
- Generate events for Higgs production in gluon fusion, both in the loop-induced and in the effective theory (use the `heft` model in that case)
  - `> generate g g > h [QCD]`
  - `> output gg_h_loopinduced`
  - `> launch`
  - `> import model heft`
  - `> generate g g > h`
  - `> output gg_h_heft`
  - `> launch`

Loop-induced exercise: Solution

- Are bottom-quark loops included in the loop-induced? How can I disable them?
  - `> generate g g > h [QCD]`
    `Total: 1 processes with 4 diagrams`
  - `> display diagrams`



- Yes, b quarks are there. Just veto them!
  - `> generate g g > h / b [QCD]`
    `Total: 1 processes with 2 diagrams`



Loop-induced exercise: Solution

- Run `pythia` and `MadAnalysis` on the event samples in order to obtain plots
  - If you installed pythia/MA before generating the folders, after `./bin/generate_events` you can choose to run them directly

```
The following switches determine which programs are run:
 1 Run the pythia shower/hadronization:                                pythia=ON
 2 Run PGS as detector simulator:                                         pgs=OFF
 3 Run Delphes as detector simulator:                                delphes=NOT INSTALLED
 4 Decay particles with the MadSpin module:                          madspin=OFF
 5 Add weights to the events based on changing model parameters:   reweight=Not available (requires
NumPy)
  Either type the switch number (1 to 5) to change its default setting,
  or set any switch explicitly (e.g. type 'madspin=ON' at the prompt)
  Type '0', 'auto', 'done' or just press enter when you are done.
 [0, 1, 2, 4, 5, auto, done, pythia=ON, pythia=OFF, ... ][60s to answer]
```

  - After pythia, you will have a .hep file in the `Events/run_XXX` directory, which can be analysed.
  - The .top/.ps/.jpg plots can be found in `HTML/run_XXX/plots_parton/` and `HTML/run_XXX/plots_pythia_tag_Y/`

file:///Users/marcozaro/Physics/MadGraph/MG5_aMC_v2_4.

Available Results

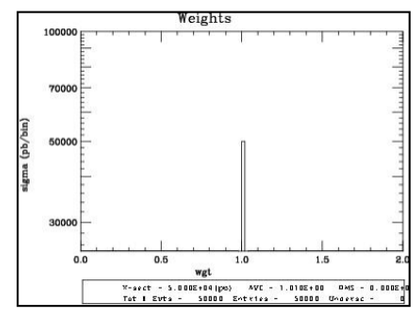
Run	Collider	Banner	Cross section (pb)	Events	Data	Output	
run_01	p p 6500.0 x 6500.0 GeV	tag_1	15.71 ± 0.042	10000	parton madevent	LHE	rem laun
					pythia	LOG STDHEP LHE plots	rem
		tag_2	15.71 ± 0.042	10000	pythia madevent	LOG	rem
		tag_3	15.71 ± 0.042	10000	pythia madevent	LOG STDHEP LHE plots	rem laun
run_full	p p 6500.0 x 6500.0 GeV	tag_3	15.73 ± 0.012	50000	parton madevent	LHE plots	rem laun
					pythia	LOG STDHEP LHE plots	rem laun
run_nob	p p 6500.0 x 6500.0 GeV	tag_3	17.61 ± 0.0074	50000	parton madevent	LHE plots	rem laun
					pythia	LOG STDHEP LHE plots	rem laun

Main Page

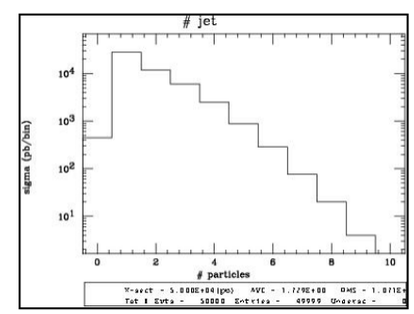
file:///Users/marcozaro/Physics/MadGraph/MG5_aMC_v...

Plots for Unknown at pythia level

Name	Variable
weigth	weights of the events (normally 1)
Ht	scalar sum of pt of all particles + missing Et
pt(i)	transverse momentum
y(i)	rapidity in the lab
R(i,j)	distance in the (y,phi) plane
m(i,j)	invariant mass
cos(i,j)	angle between direction of i in the resframe of i+j and the direction of i+j in the labframe



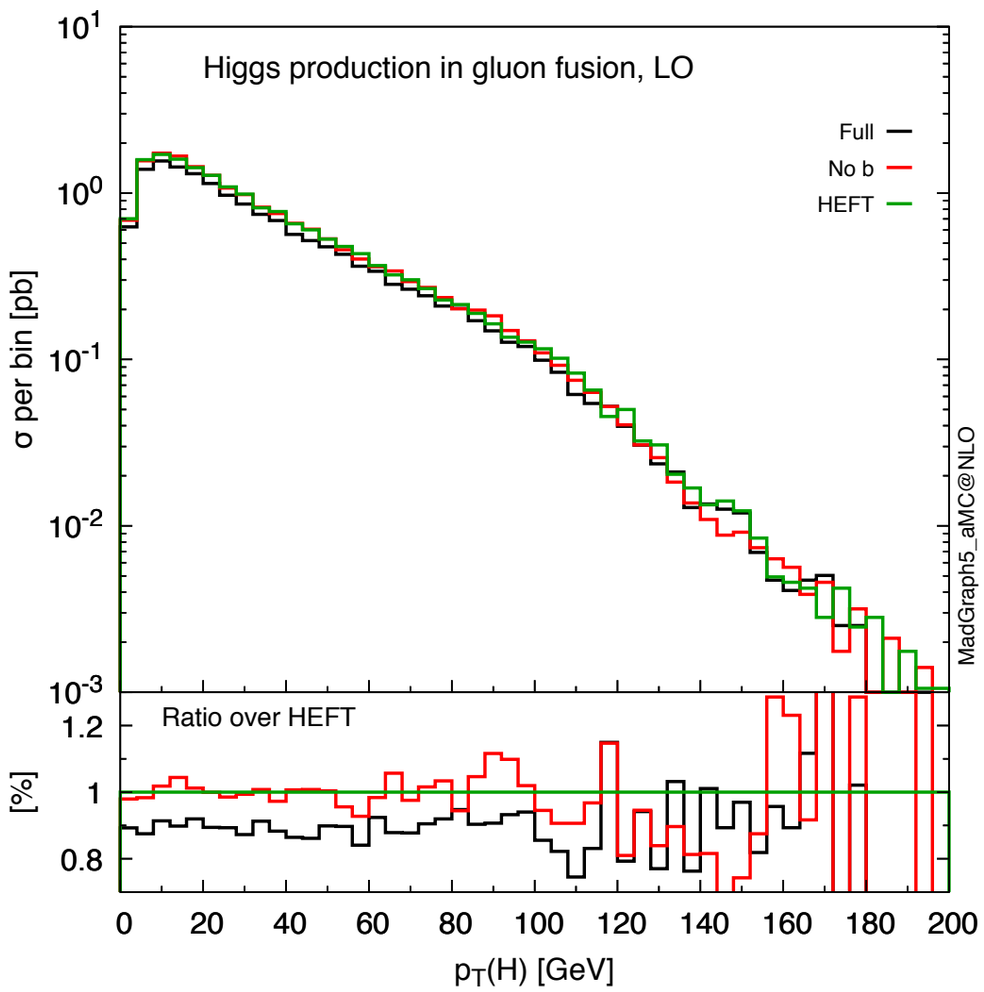
Download PS ma_plot000.ps



Download PS ma_plot001.ps

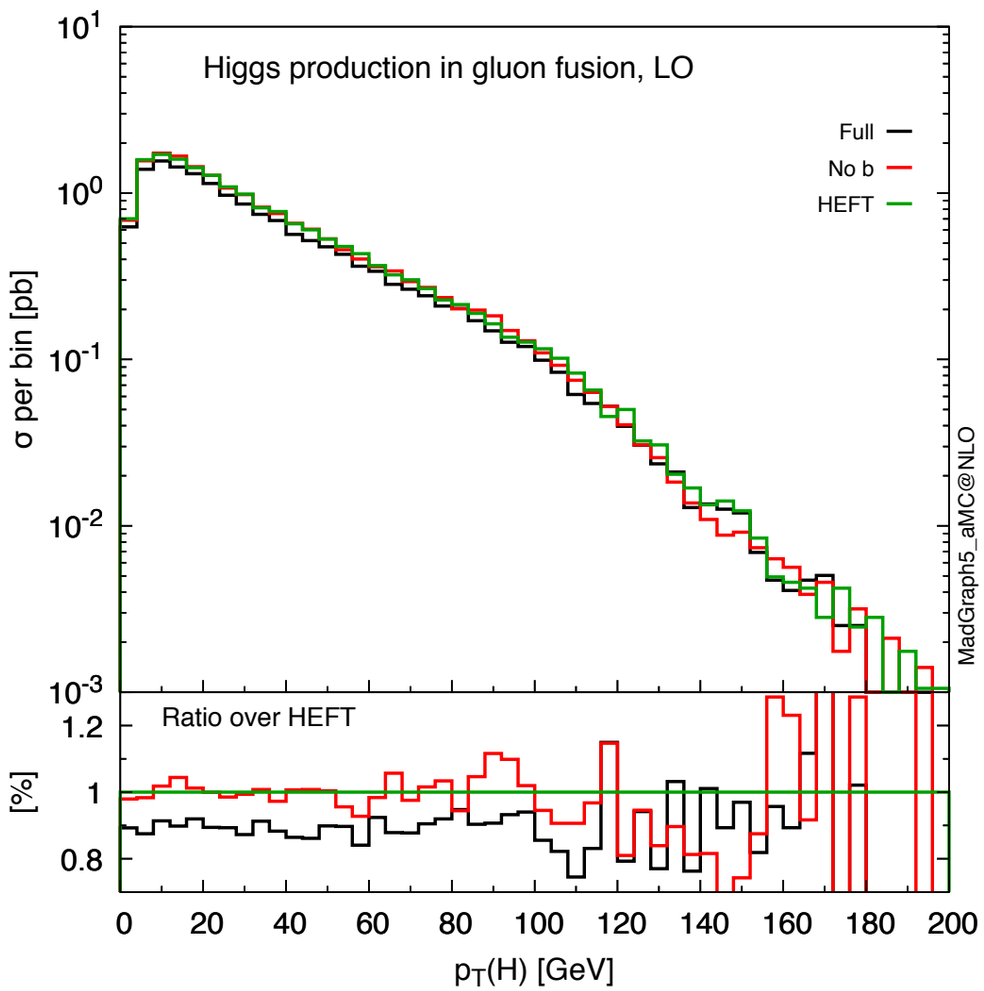
Loop-induced exercise: Solution

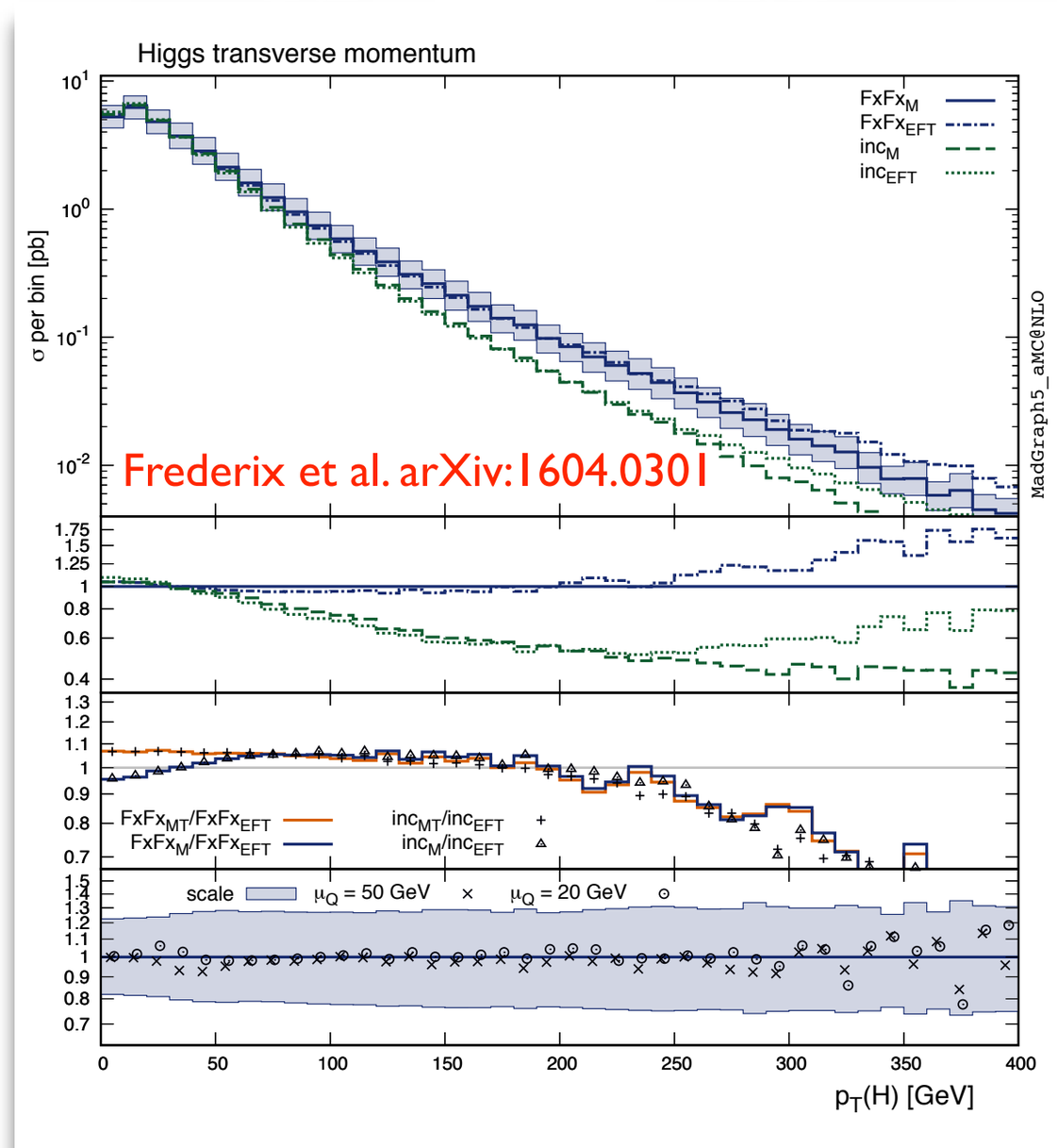
- Compare the Higgs p_T plot with t/b, without b and in the EFT and comment

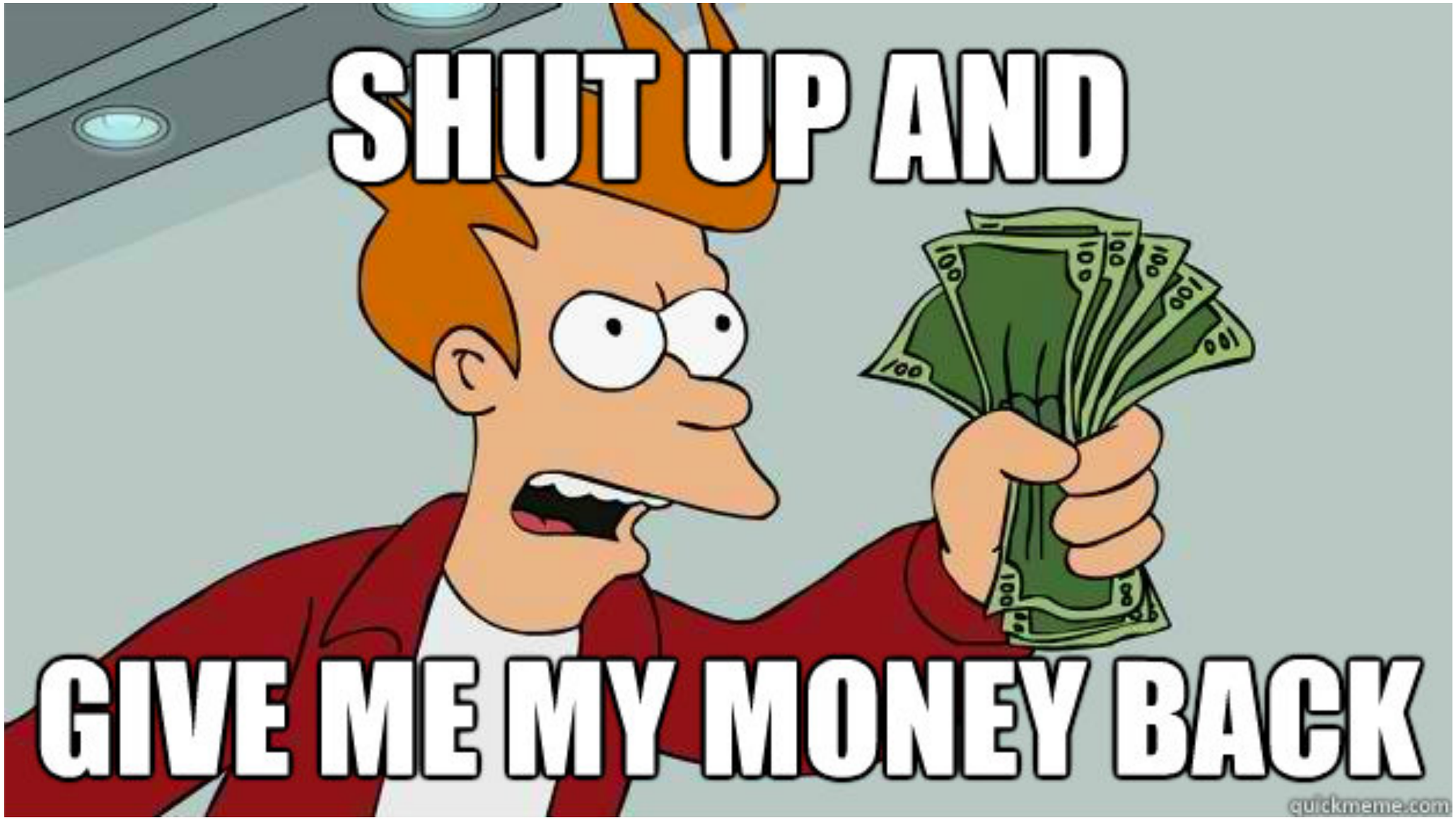


Loop-induced exercise: Solution

- Compare the Higgs p_T plot with t/b, without b and in the EFT and comment






SHUT UP AND
GIVE ME MY MONEY BACK
quickmeme.com

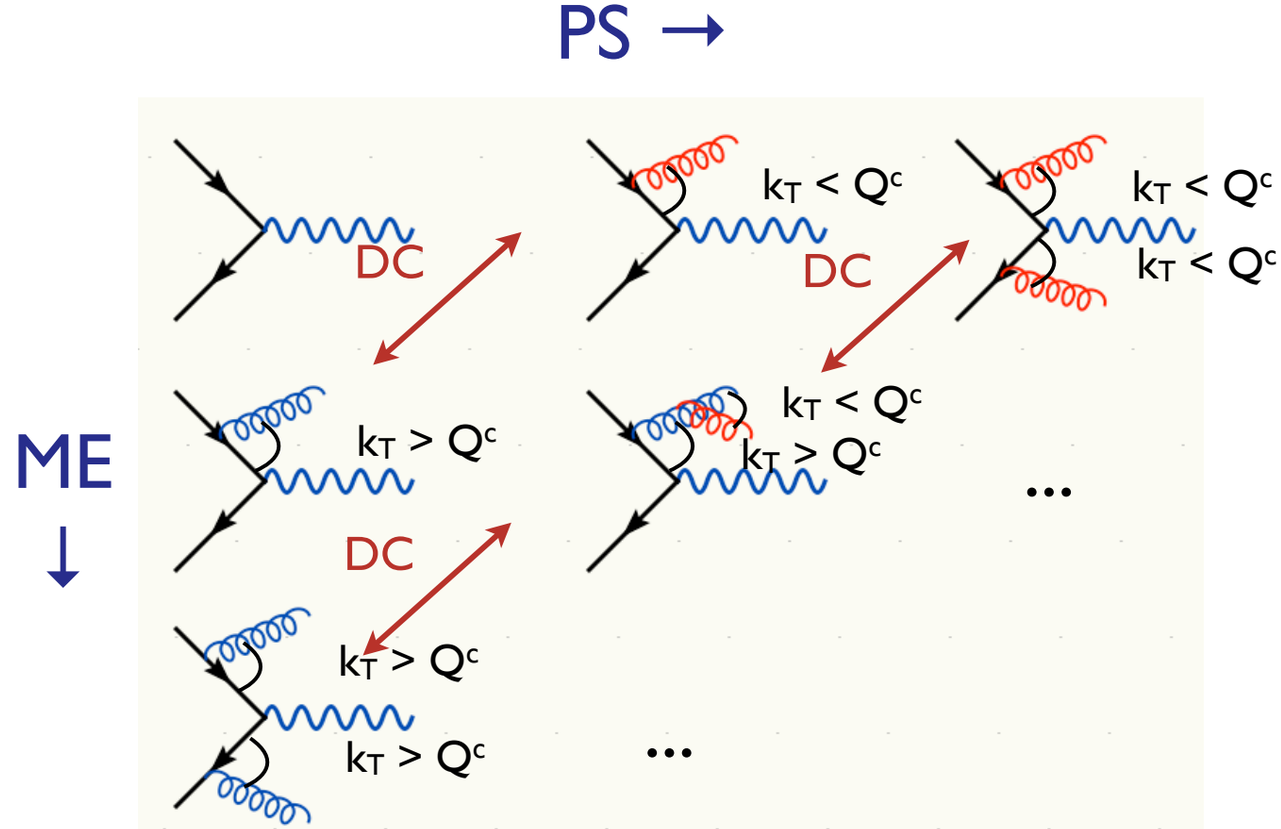
Loop-induced exercise: Solution

- Compare the Higgs p_T plot with t/b, without b and in the EFT and comment
- Well, at this level of refinement, all samples look the same (apart for 'Full' having a smaller rate due to the t/b interference)
- Small-p_T effects due to a prescription on the resummation (shower) scale; large-p_T effects to the inclusion of extra radiation

Part 5:
Merging

Merging at LO: CKKW and MLM

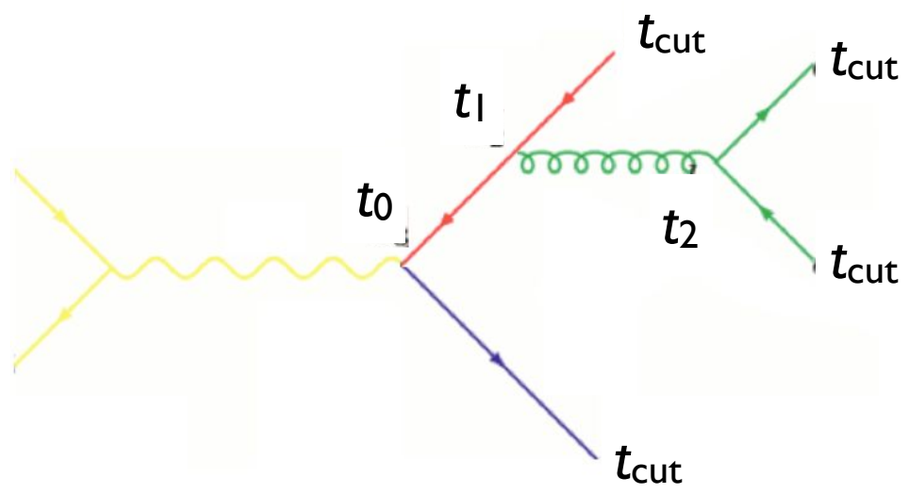
- Aim: improve shower description by including matrix-element corrections, without double counting
- Double counting can be avoided by introducing a cutoff scale Q_c: above use ME, below use shower



The evil is in the details…

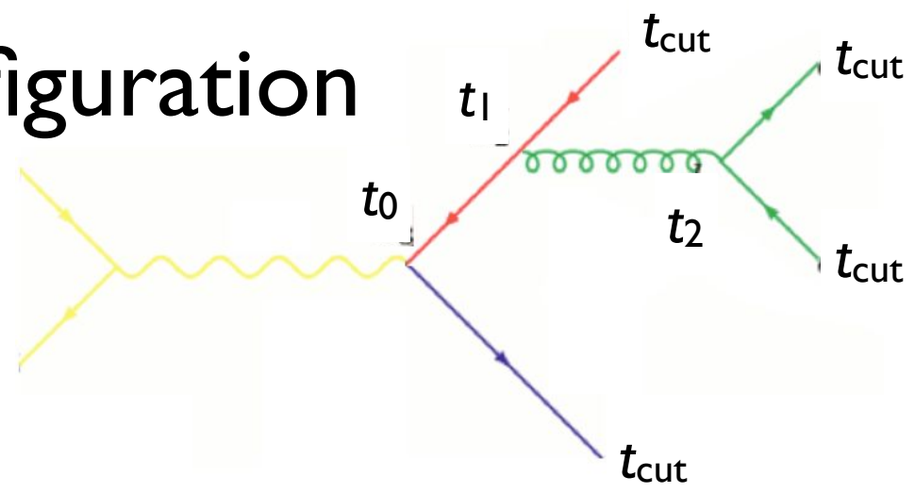
- Some refinement is needed in order to have smooth transition between PS- and ME-dominated regions: make ME and PS as similar as possible

The evil is in the details…



The evil is in the details…

- Consider this configuration

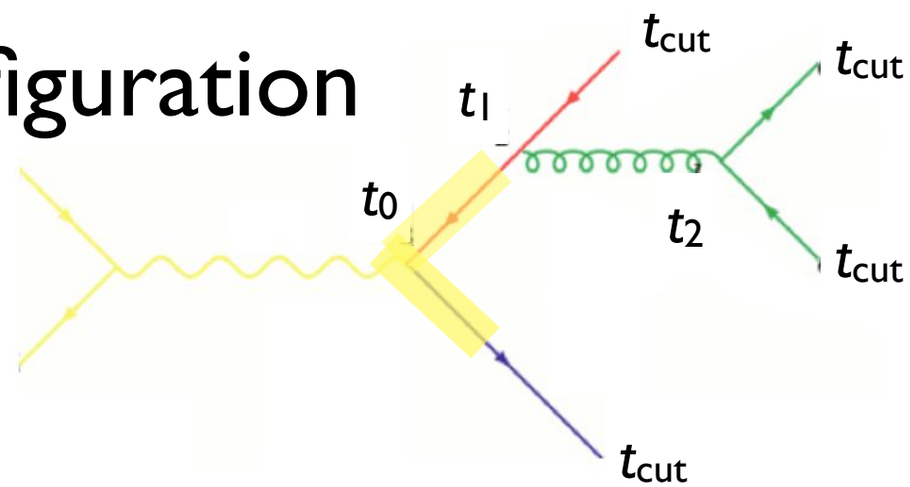


- The probability for the splitting at t_1 is given by

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

The evil is in the details…

- Consider this configuration

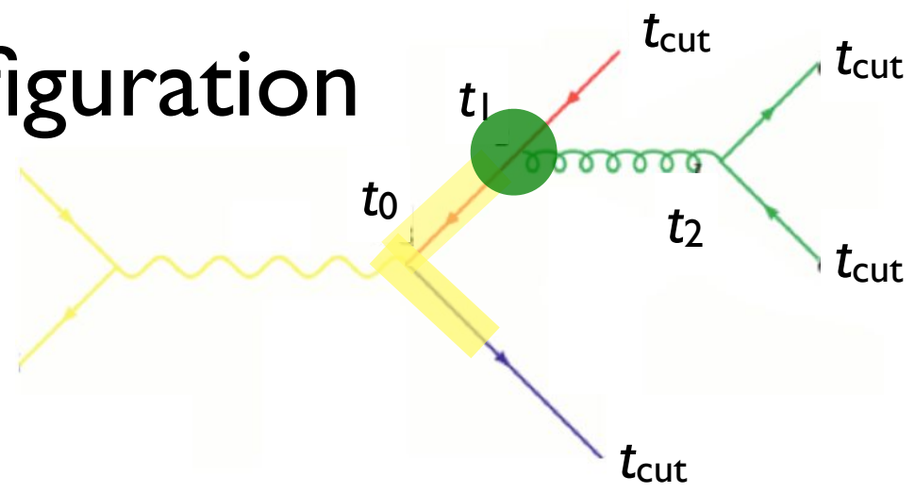


- The probability for the splitting at t_1 is given by

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

The evil is in the details…

- Consider this configuration



- The probability for the splitting at t_1 is given by

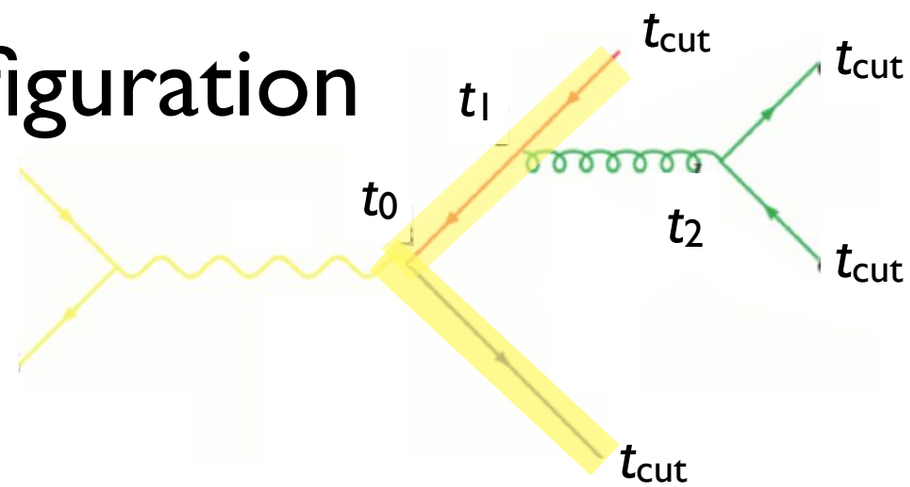
$$(\Delta_q(t_1, t_0))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z)$$

- The probability for the full history

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

The evil is in the details…

- Consider this configuration



- The probability for the splitting at t_1 is given by

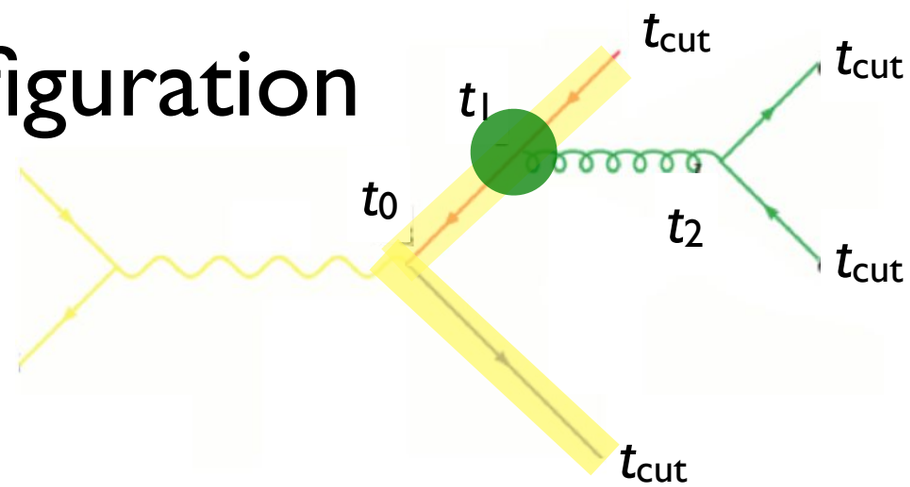
$$(\Delta_q(t_1, t_0))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z)$$

- The probability for the full history

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

The evil is in the details…

- Consider this configuration



- The probability for the splitting at t_1 is given by

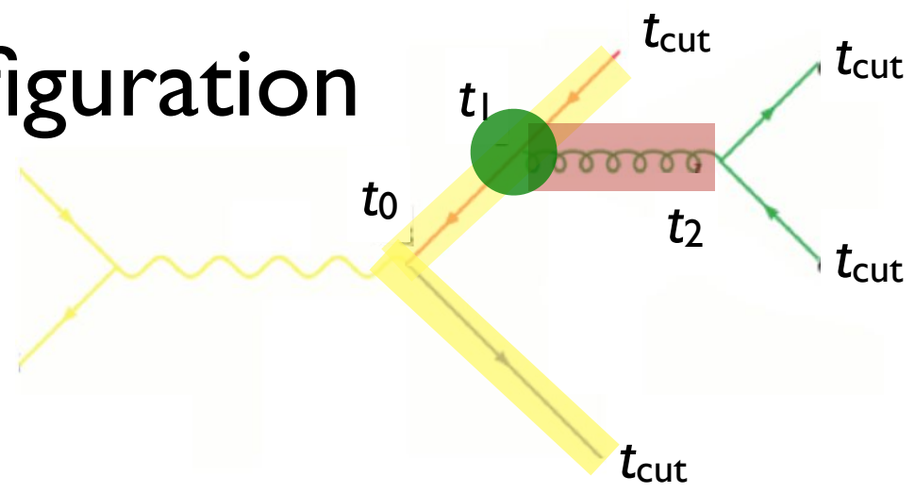
$$(\Delta_q(t_1, t_0))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z)$$

- The probability for the full history

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

The evil is in the details…

- Consider this configuration



- The probability for the splitting at t_1 is given by

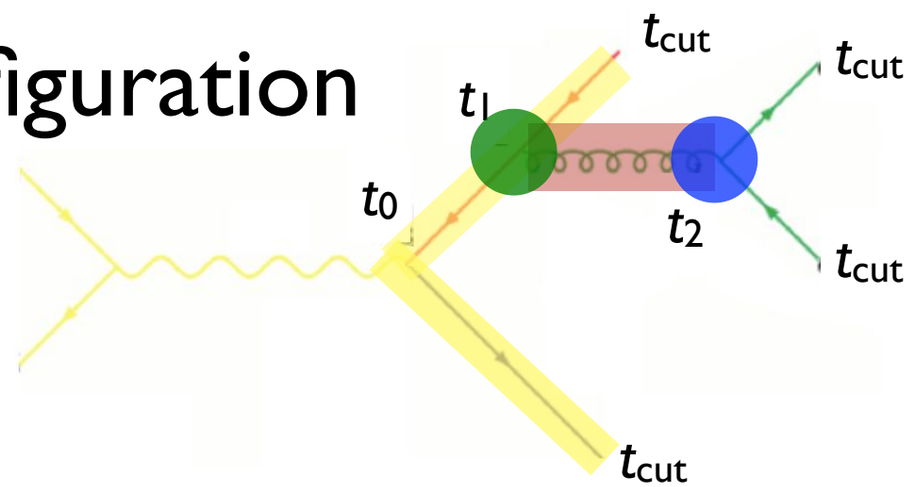
$$(\Delta_q(t_1, t_0))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z)$$

- The probability for the full history

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

The evil is in the details…

- Consider this configuration



- The probability for the splitting at t_1 is given by

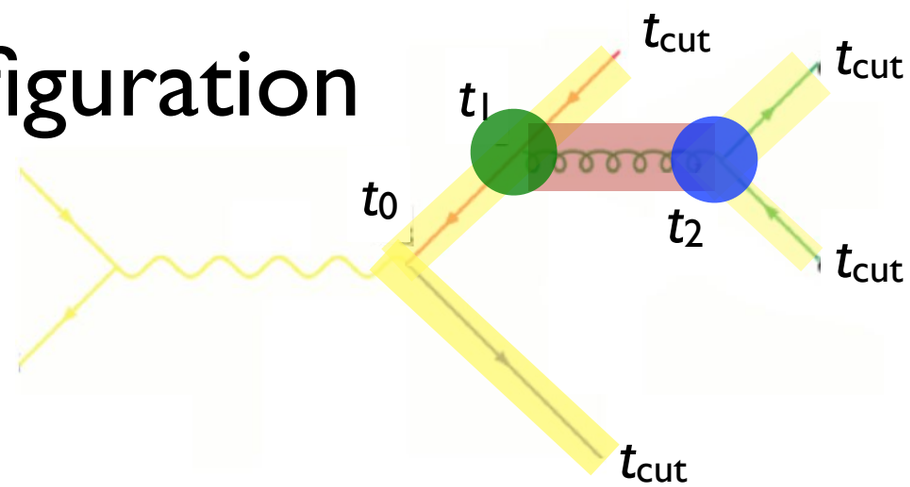
$$(\Delta_q(t_1, t_0))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z)$$

- The probability for the full history

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

The evil is in the details…

- Consider this configuration



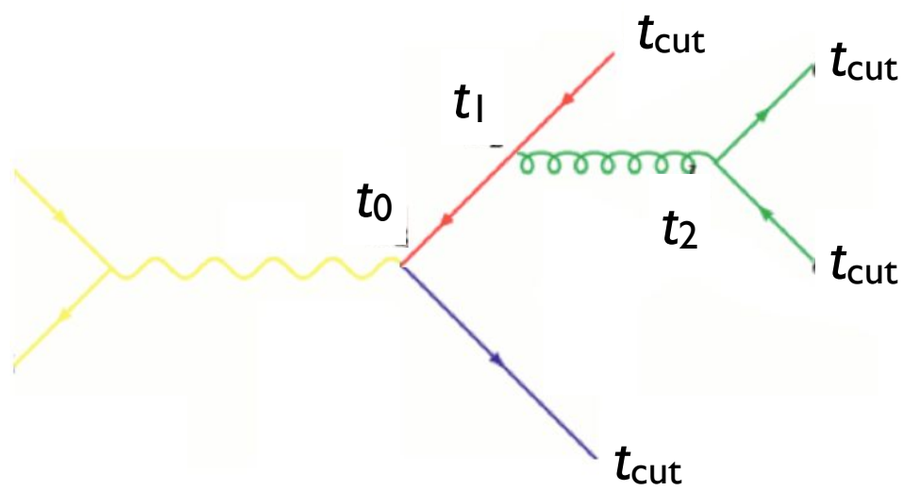
- The probability for the splitting at t_1 is given by

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

- The probability for the full history

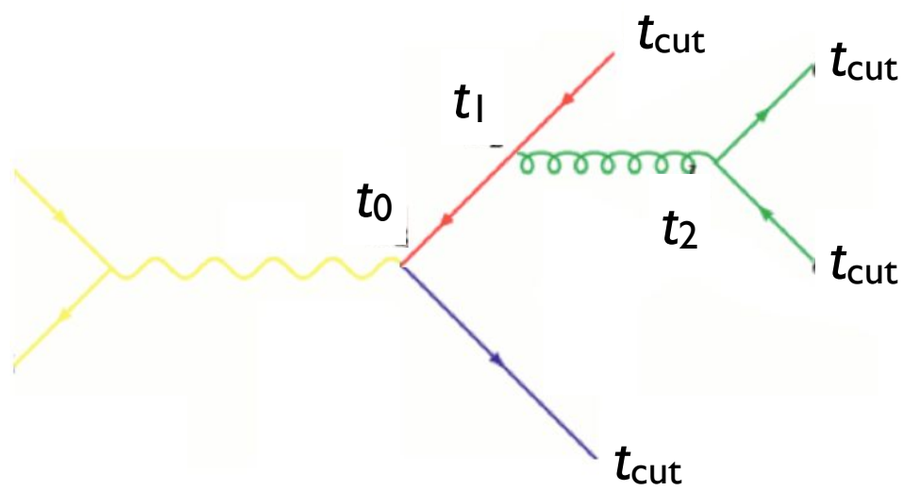
$$(\Delta_q(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1) (\Delta_q({\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z) \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$

The evil is in the details…



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

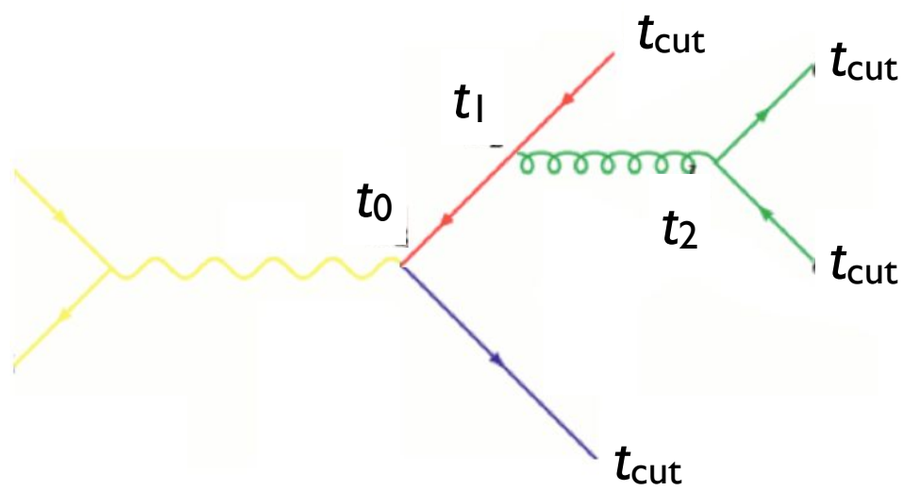
The evil is in the details…



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

Corresponds to the (collinear approximation of the) matrix element
BUT with α_s evaluated at the scale of each splitting

The evil is in the details…

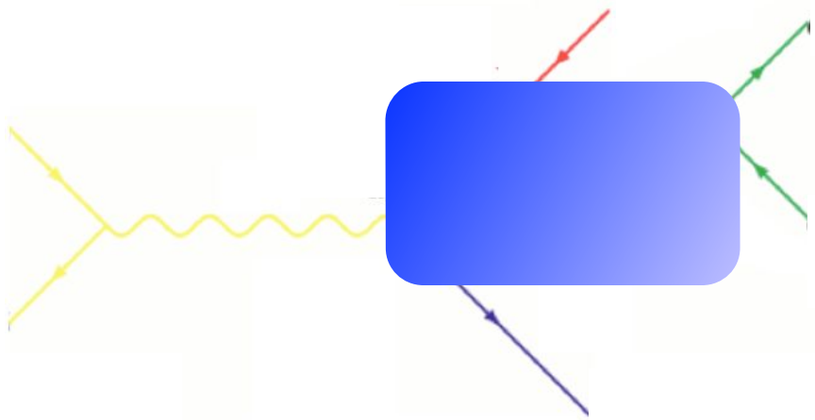


$$(\Delta_q(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q({\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z) \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$

Corresponds to the (collinear approximation of the) matrix element BUT with α_s evaluated at the scale of each splitting

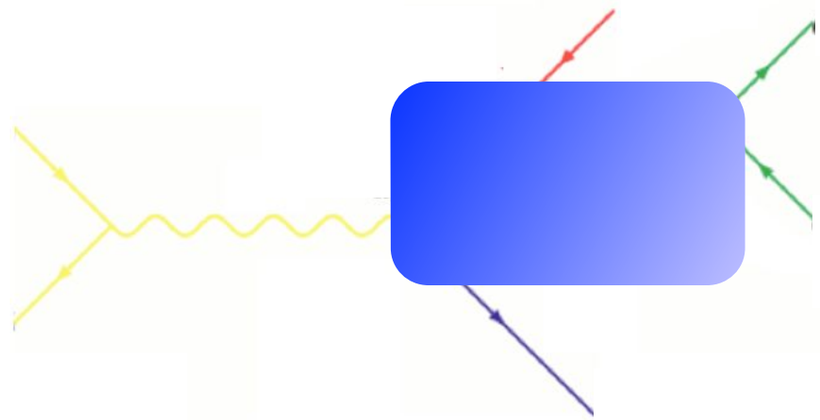
Sudakov suppression due to disallowing additional radiation above the scale t_{cut}

In practice: how to?



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

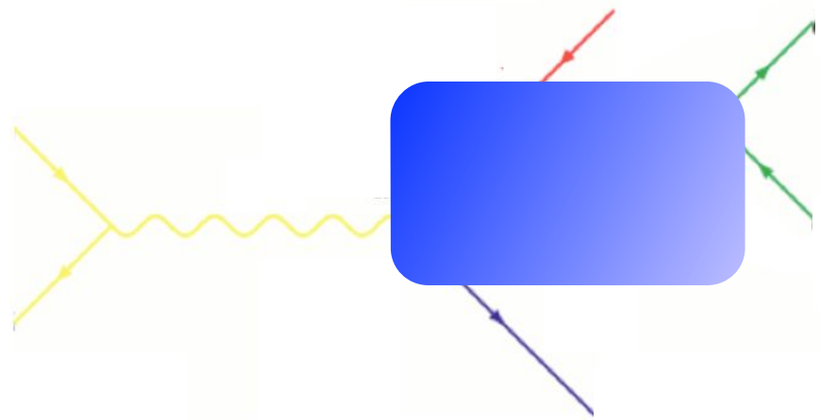
In practice: how to?



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

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

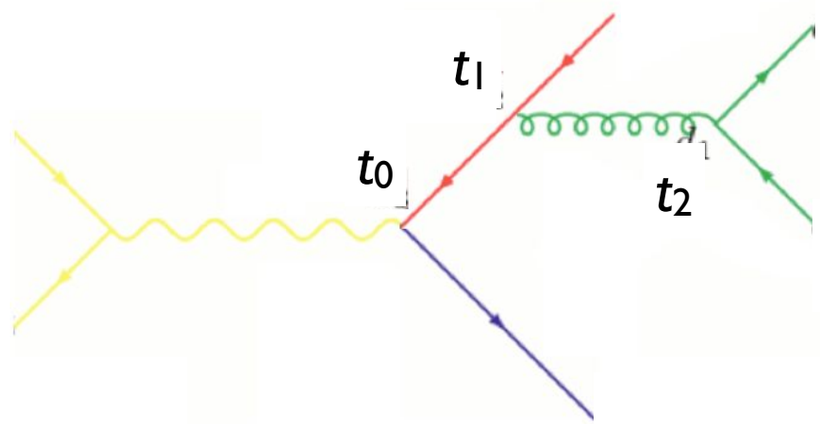
In practice: how to?



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

- 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"

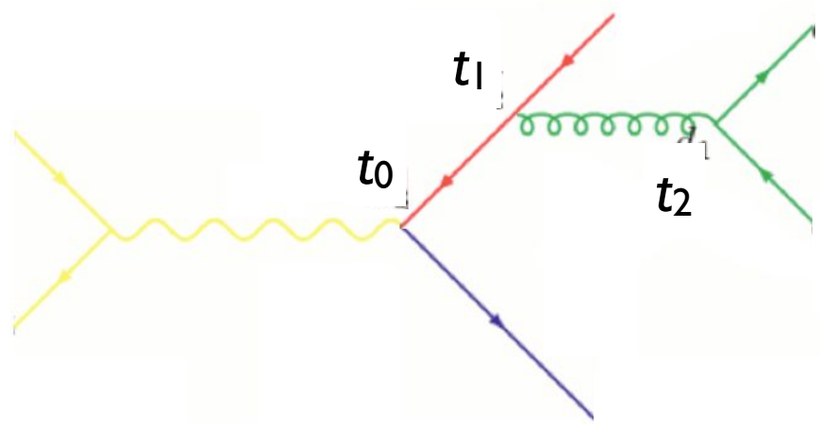
In practice: how to?



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

- 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"

In practice: how to?

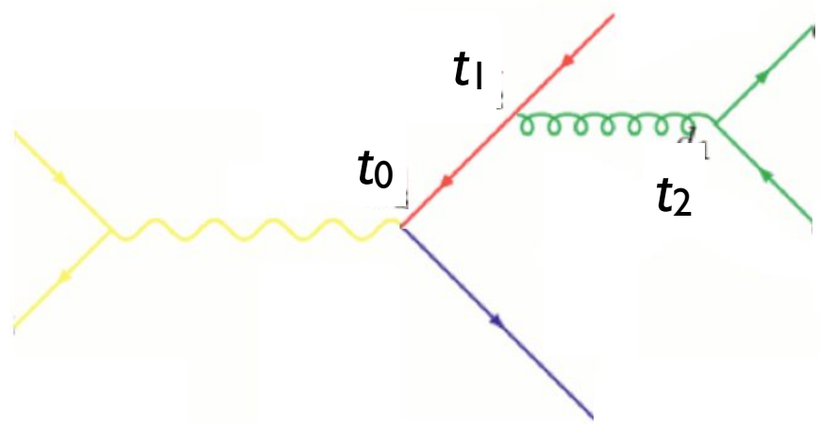


$$|\mathcal{M}|^2(\hat{s}, p_3, p_4, \ldots)$$

- 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

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

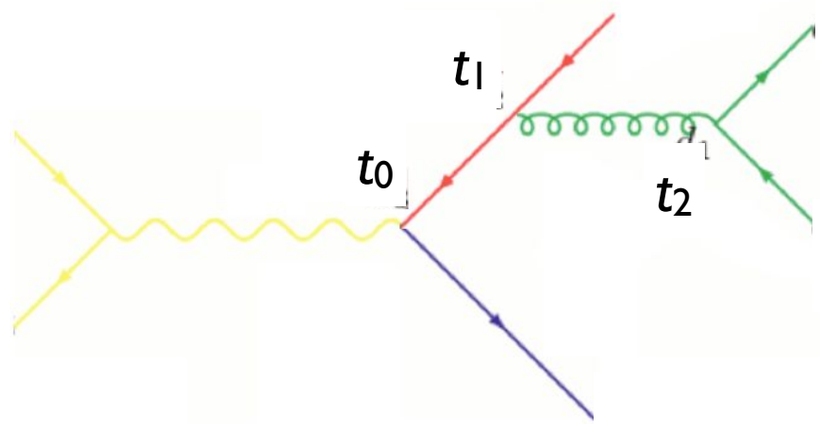
In practice: how to?



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

- 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

In practice: how to?



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

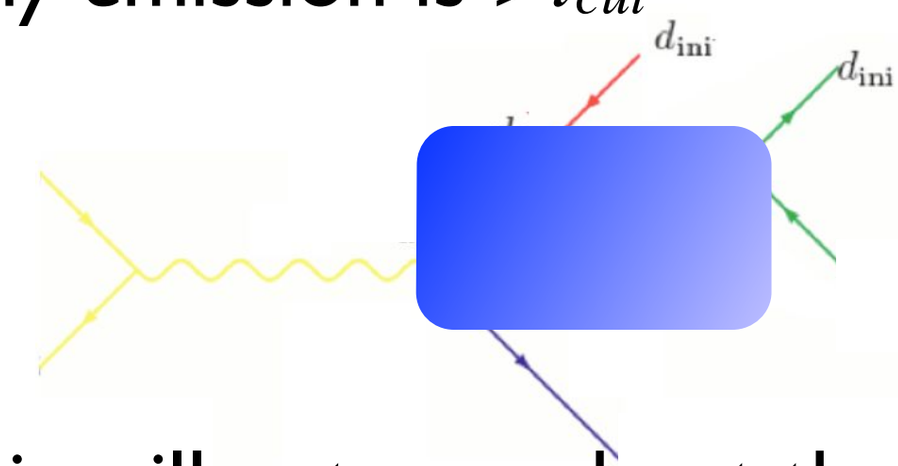
- To get an equivalent treatment of the corresponding matrix element, do as follows:
  1. Cluster the event using some clustering algorithm
     - this gives us a corresponding "parton shower history"
  2. Reweight α_s in each clustering vertex with the clustering scale
     $$|\mathcal{M}|^2 \to |\mathcal{M}|^2 \frac{\alpha_s(t_1)}{\alpha_s(t_0)} \frac{\alpha_s(t_2)}{\alpha_s(t_0)}$$
  3. Use some algorithm to apply the equivalent Sudakov suppression $(\Delta_q(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q({\rm cut}, t_2))^2$
  4. For ISR, also apply PDF reweighing

The Sudakov suppression: CKKW(-L)

- CKKW: Catani, Krauss, Kuhn, Webber, hep-ph/0109231, Krauss, hep-ph/0205283
  - Use best known analytical expression for the Sudakov (NLL)
    - Mismatch between analytical Sudakov and PS
  - Use truncated-showers: no emissions above t_{cut}
    - Requires dedicated implementations
- CKKW-L: Lonnblad, hep-ph/0112284, Hoeche et al. 0903.1219
  - Obtain Sudakov suppression by running shower on the event progressively removing extra patrons, and vetoing emissions harder than those at the ME and t_{cut}
    - The Sudakov is by construction the same as the shower
    - A new implementation of the algorithm has to be done for each shower

The Sudakov suppression: MLM

- MLM: Mangano, Alwall et al. arXiv:0706.2569
  - Run the shower to obtain the Sudakov suppression, rejecting the event if any emission is $>t_{cut}$



  - In principle this will not care about the shower history, and apply a Sudakov suppression $(\Delta_q(Q^2, t_{cut}))^4$; In practice this is not an issue
  - No modification needed in the shower

How to in MadGraph?

- Generation of merged samples in MadGraph is very simple. Just generate the lowest-multiplicity process and add the higher multiplicities

```
> generate p p > w+ @0
> generate p p > w+ j @1
> . . .
> output
> launch
```

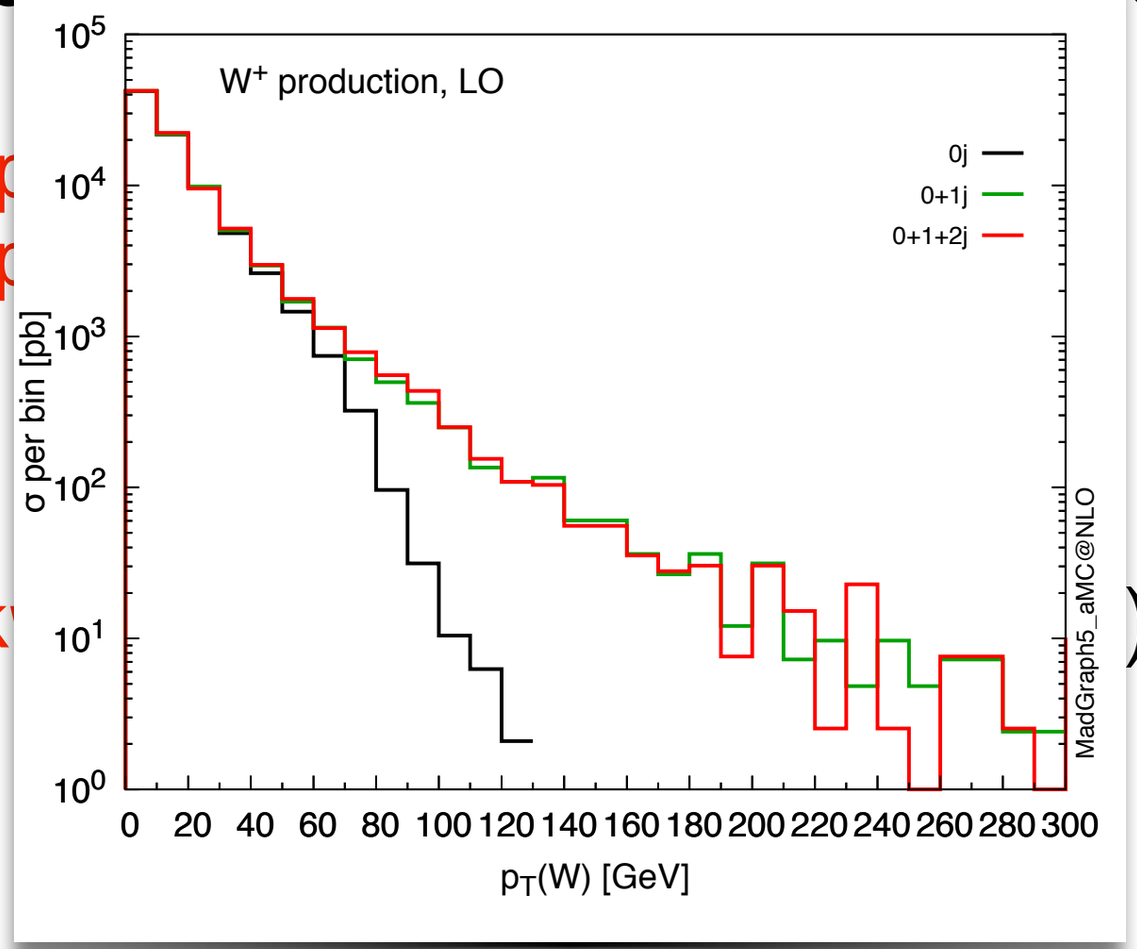
- Then set `ickkw=1` in the `run_card` (for MLM)

How to in MadGraph?

- Generation of merged samples in MadGraph is very simple. Just generate the lo dd the higher multiplicities

```
> generate p
> generate p
> . . .
> output
> launch
```

- Then set ickk)

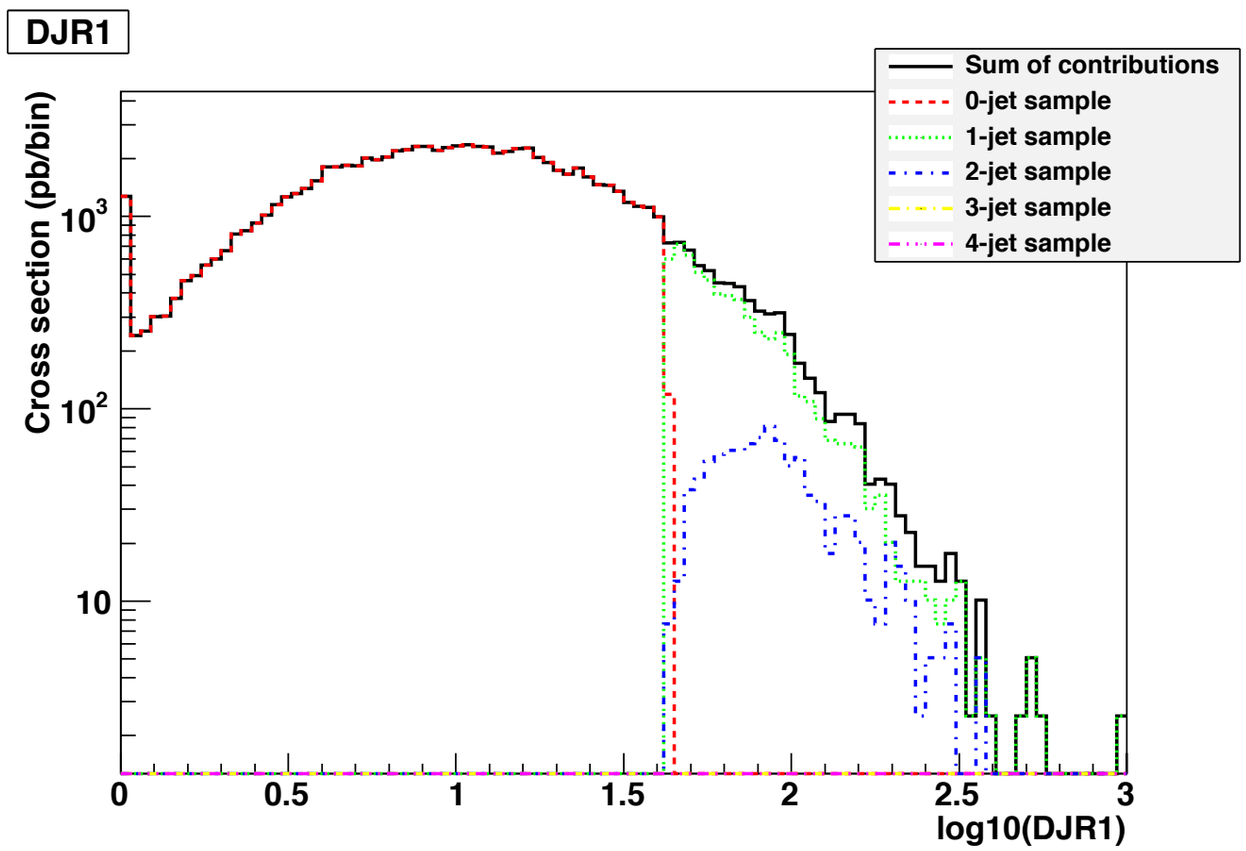


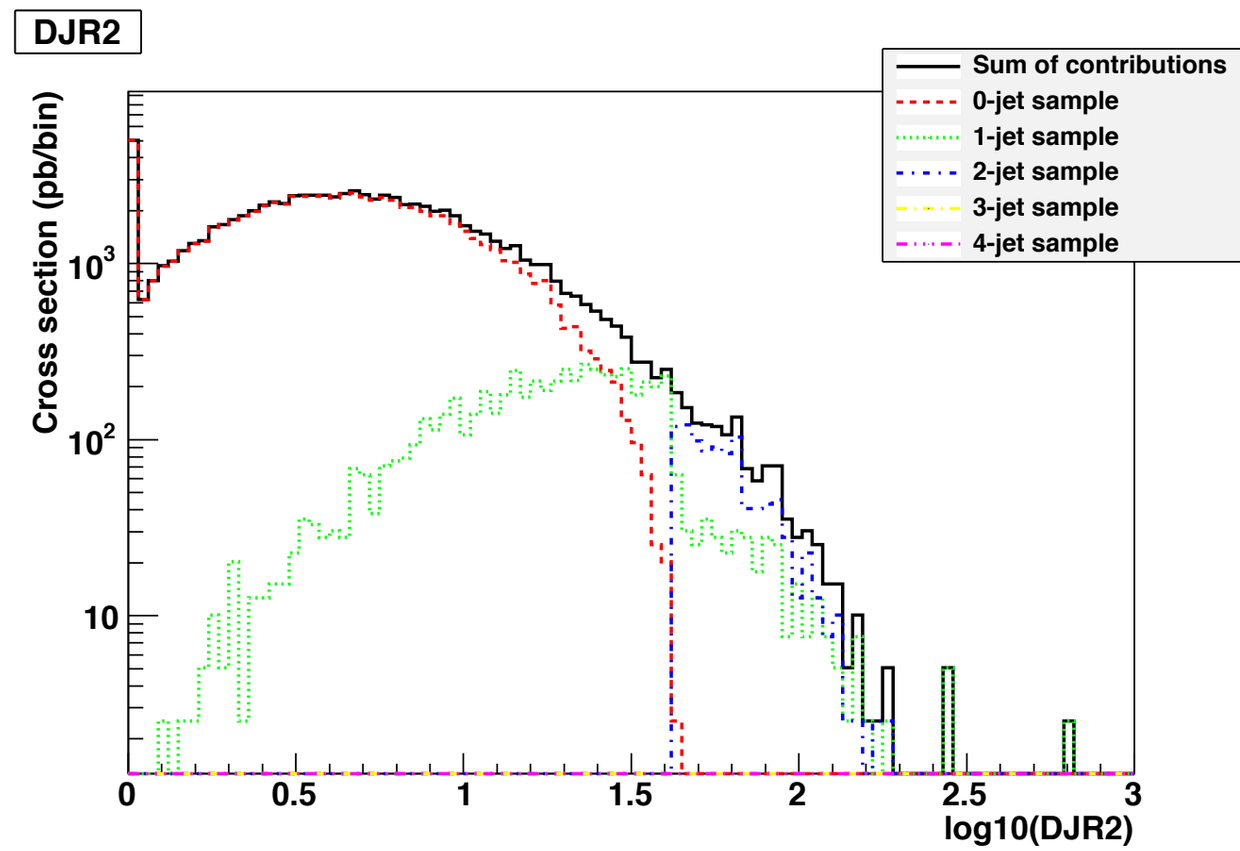
`xqcut` in the `run_card.dat` and `Qcut` in the `pythia_card.dat`

- `xqcut` acts as a generation cut for the events, at parton level: events in the Nth-multiplicity sample will have N jets with p_T>`xqcut`
- `Qcut` is the merging scale, used at the shower0level. If not specified it is set to
  `max(1.4 * xqcut, xqcut+10)`
  otherwise can be specified by adding `QCUT= XX` to the `pythia_card`. One should have (for k_T-MLM)
  `Qcut > xqcut`

Example: W+0,1,2 jets

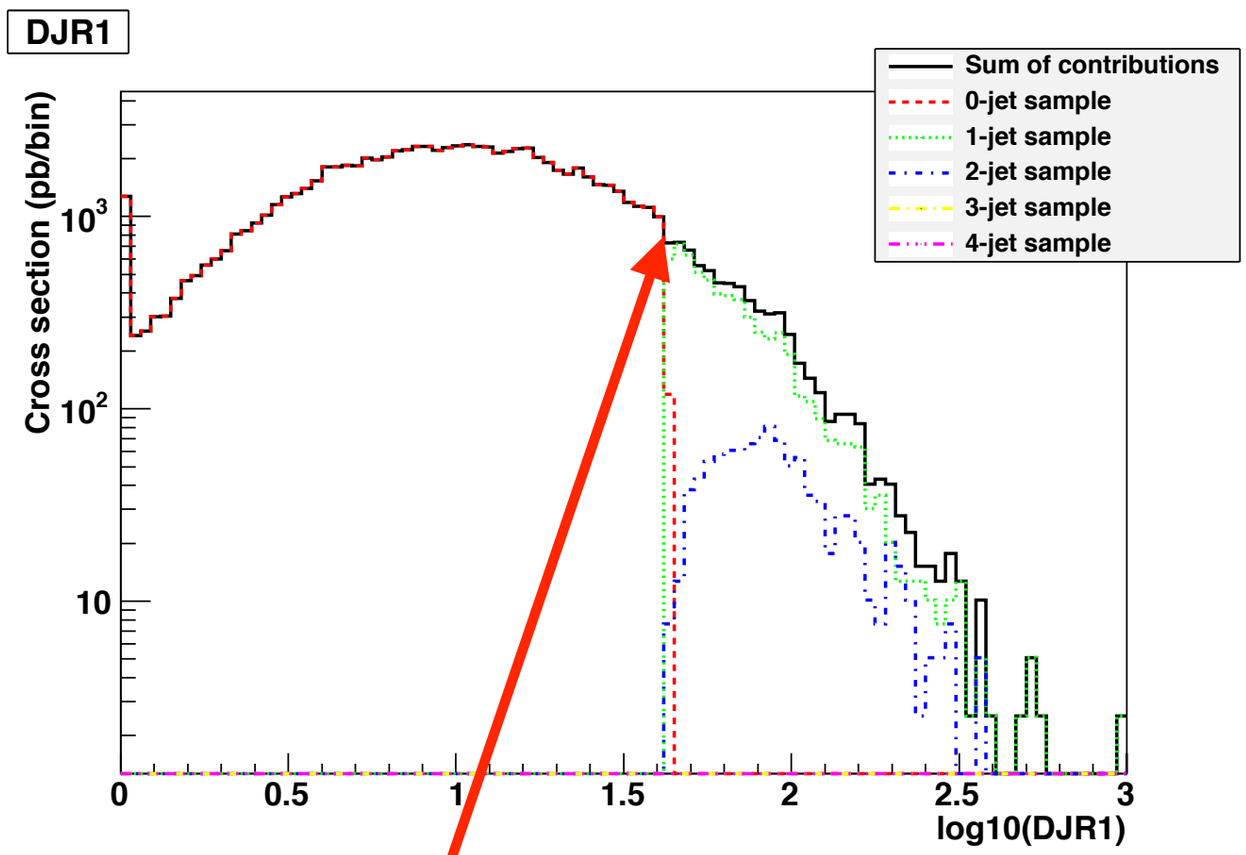
Default: `xqcut` = 30 GeV

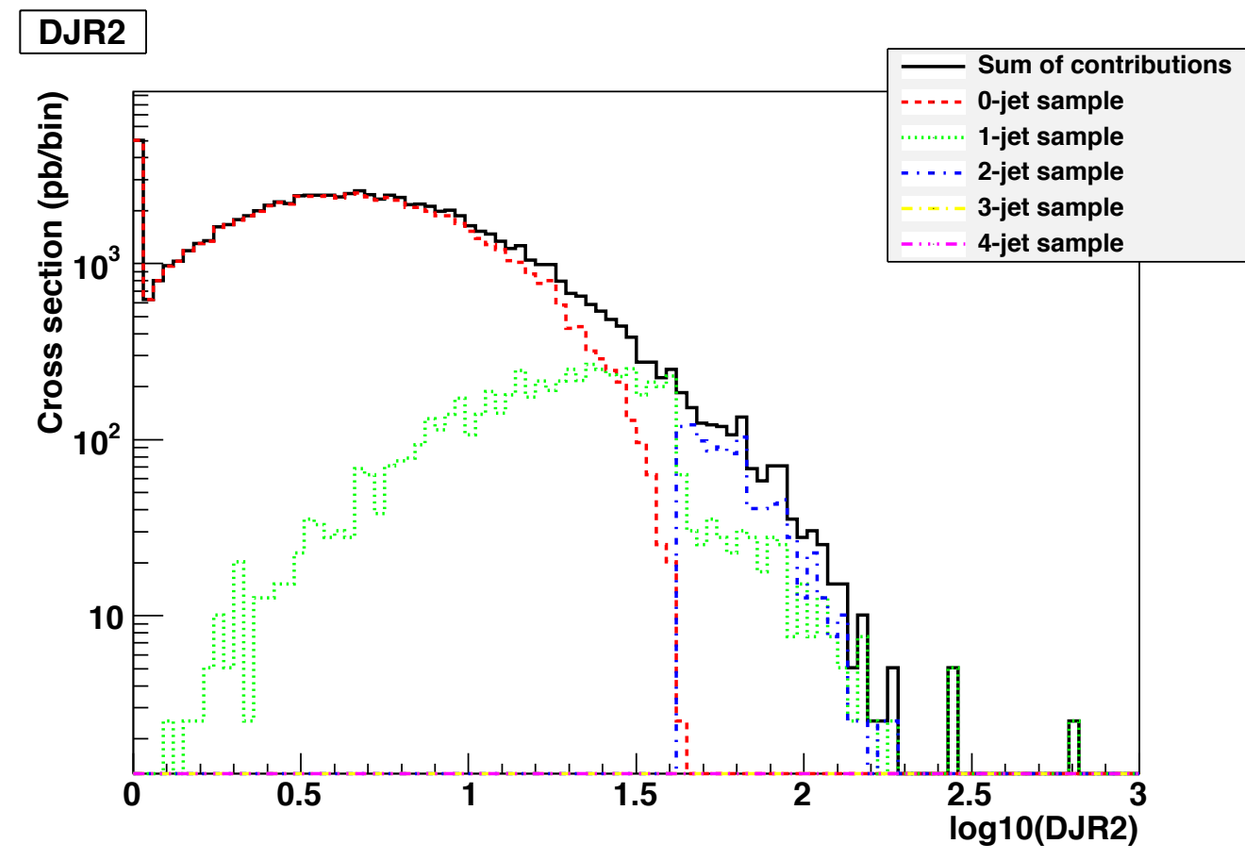




Example: W+0,1,2 jets

Default: `xqcut` = 30 GeV

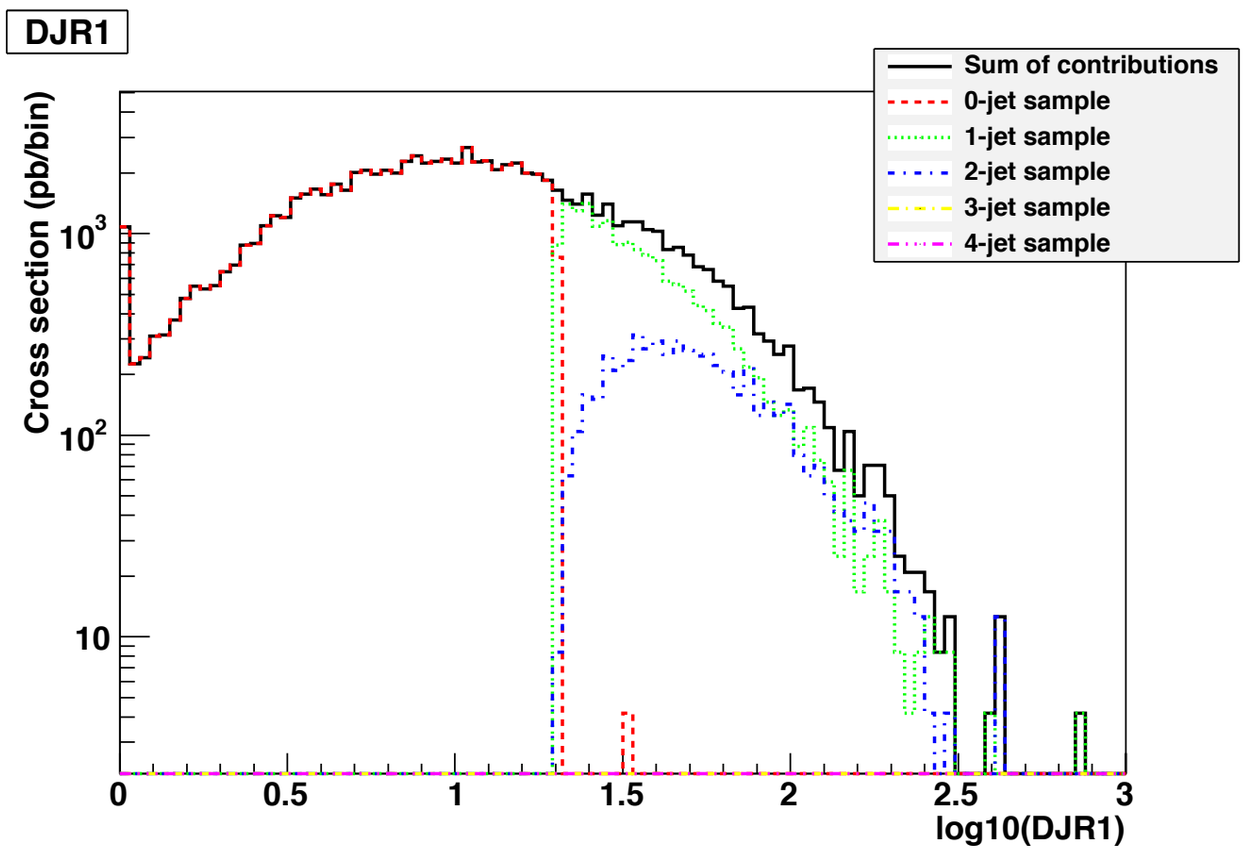


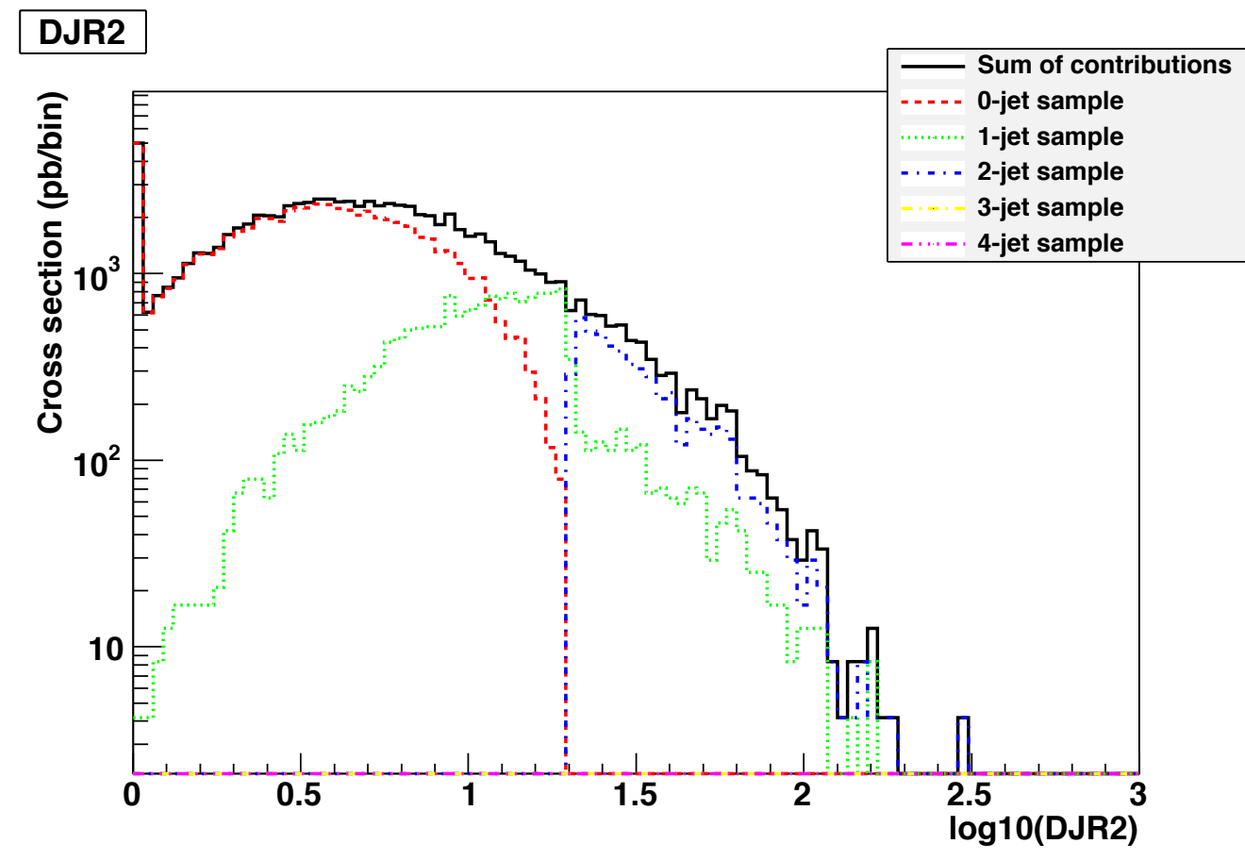


Small gap

Example: W+0,1,2 jets

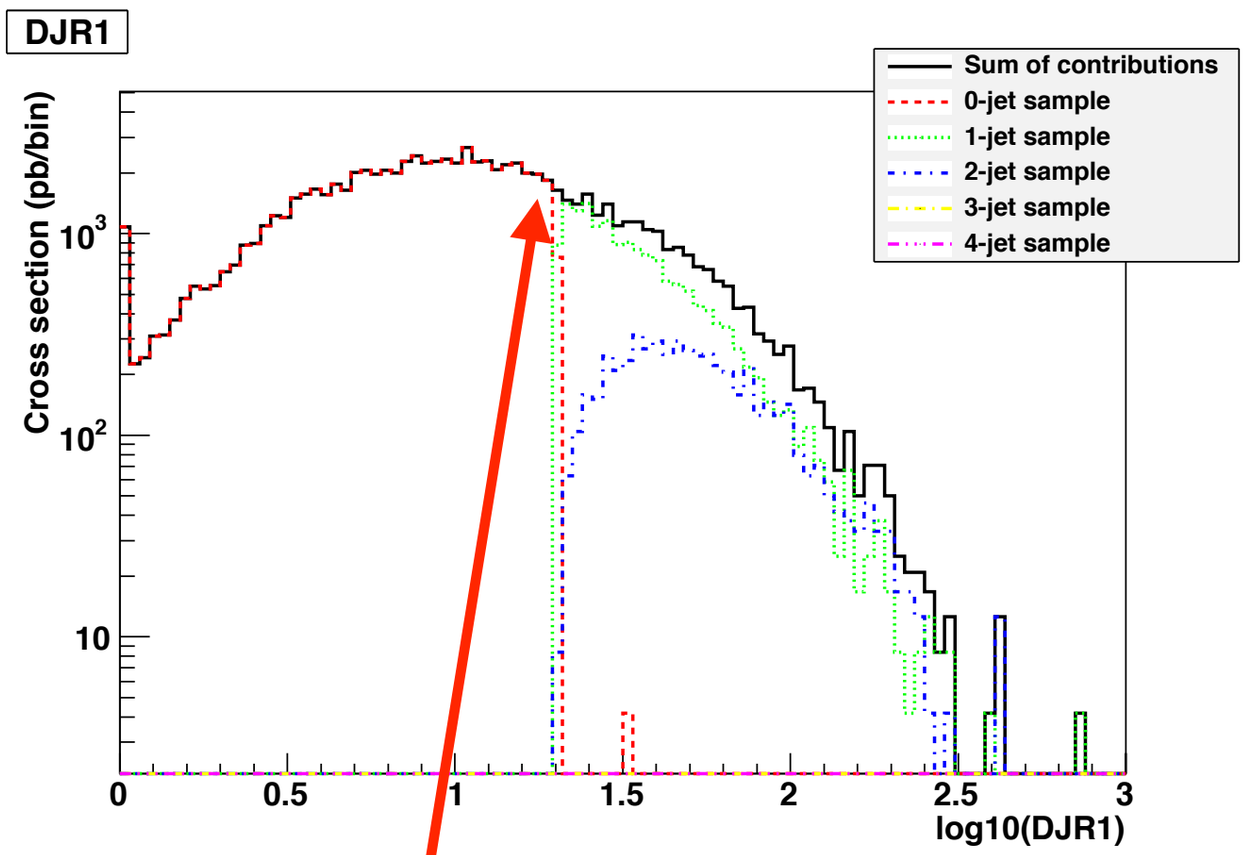
xqcut = 10 GeV

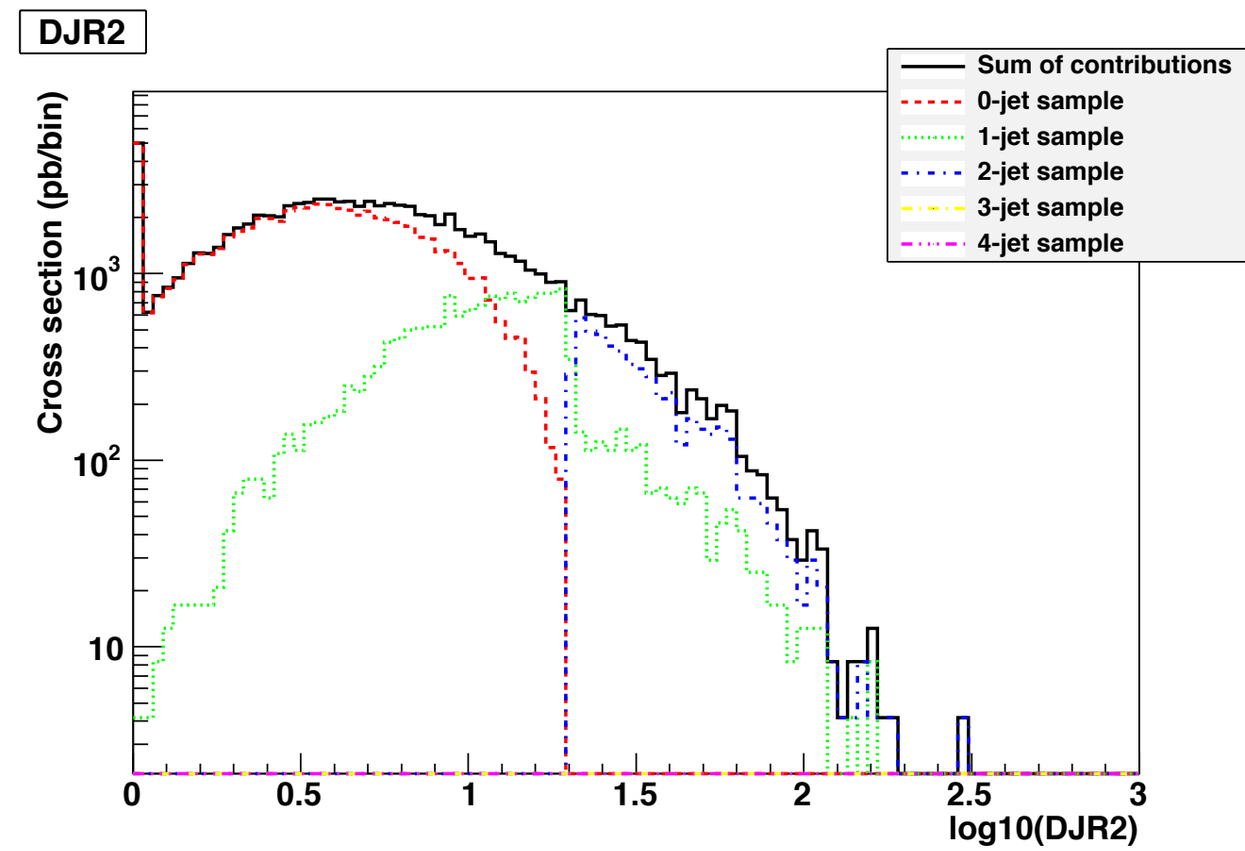




Example: W+0,1,2 jets

xqcut = 10 GeV

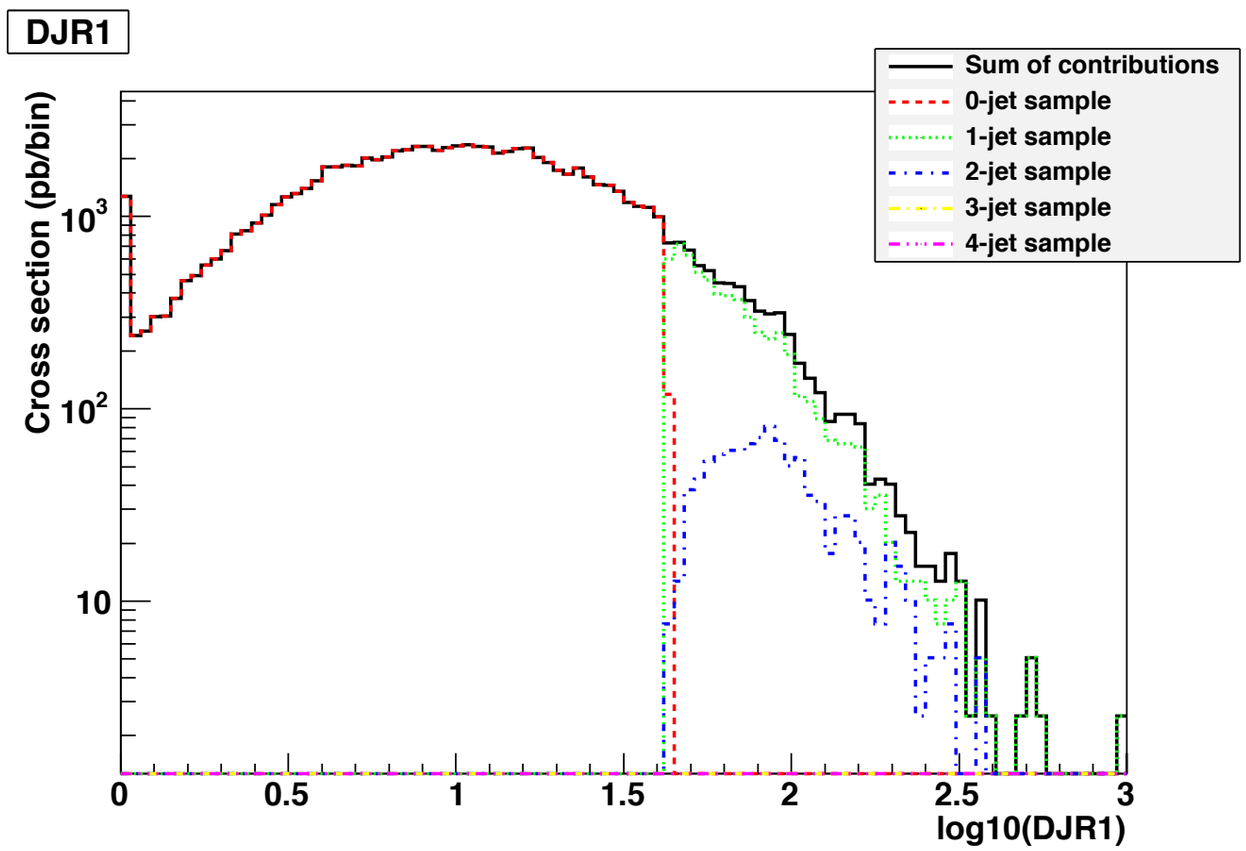


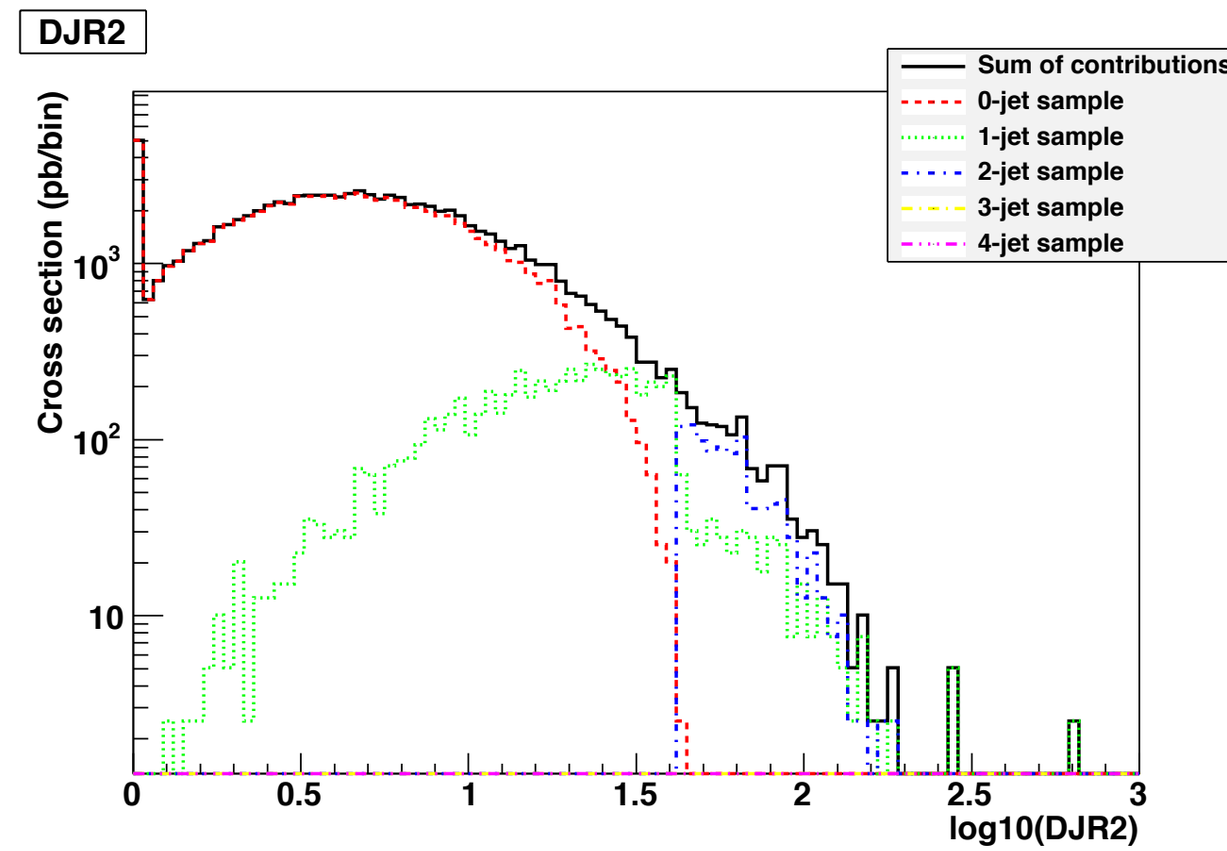


Small kink

Example: W+0,1,2 jets

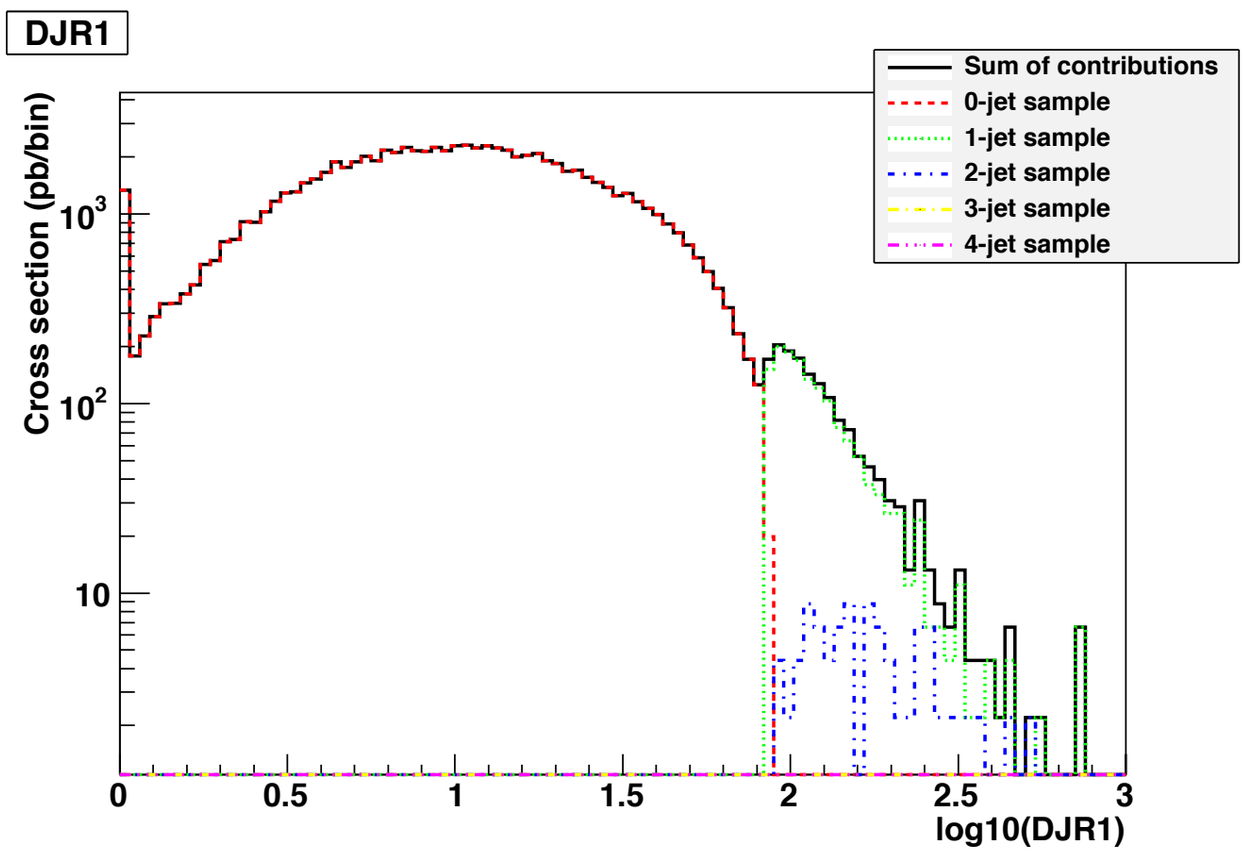
xqcut = 30 GeV

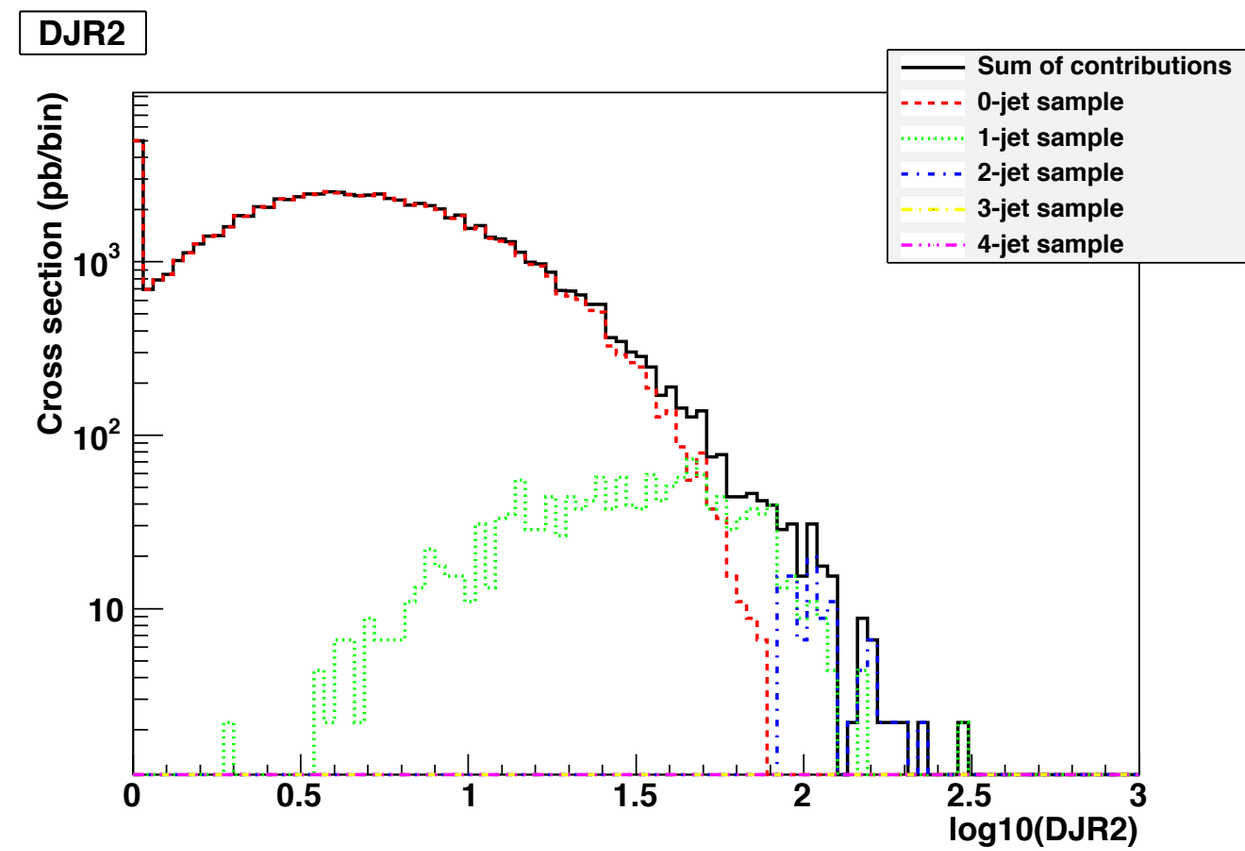




Example: W+0,1,2 jets

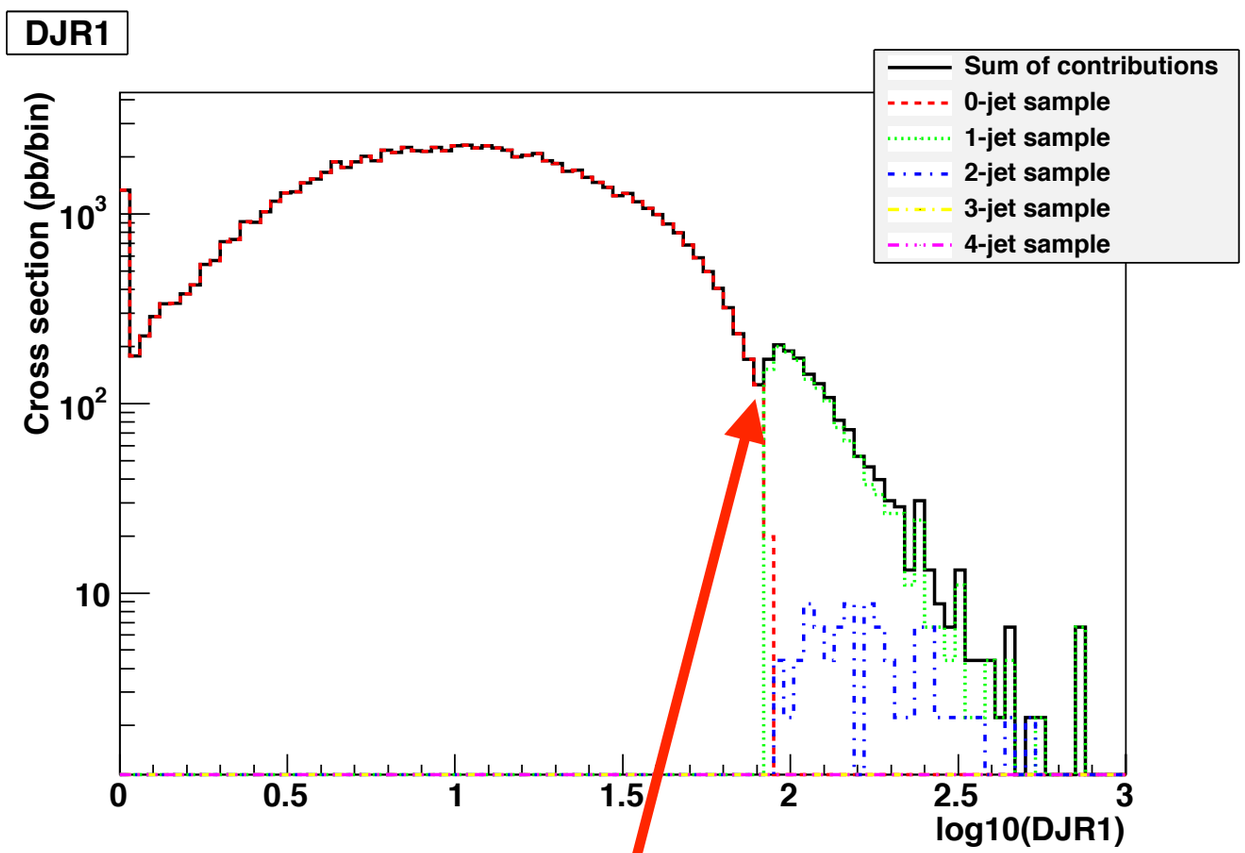
xqcut = 60 GeV

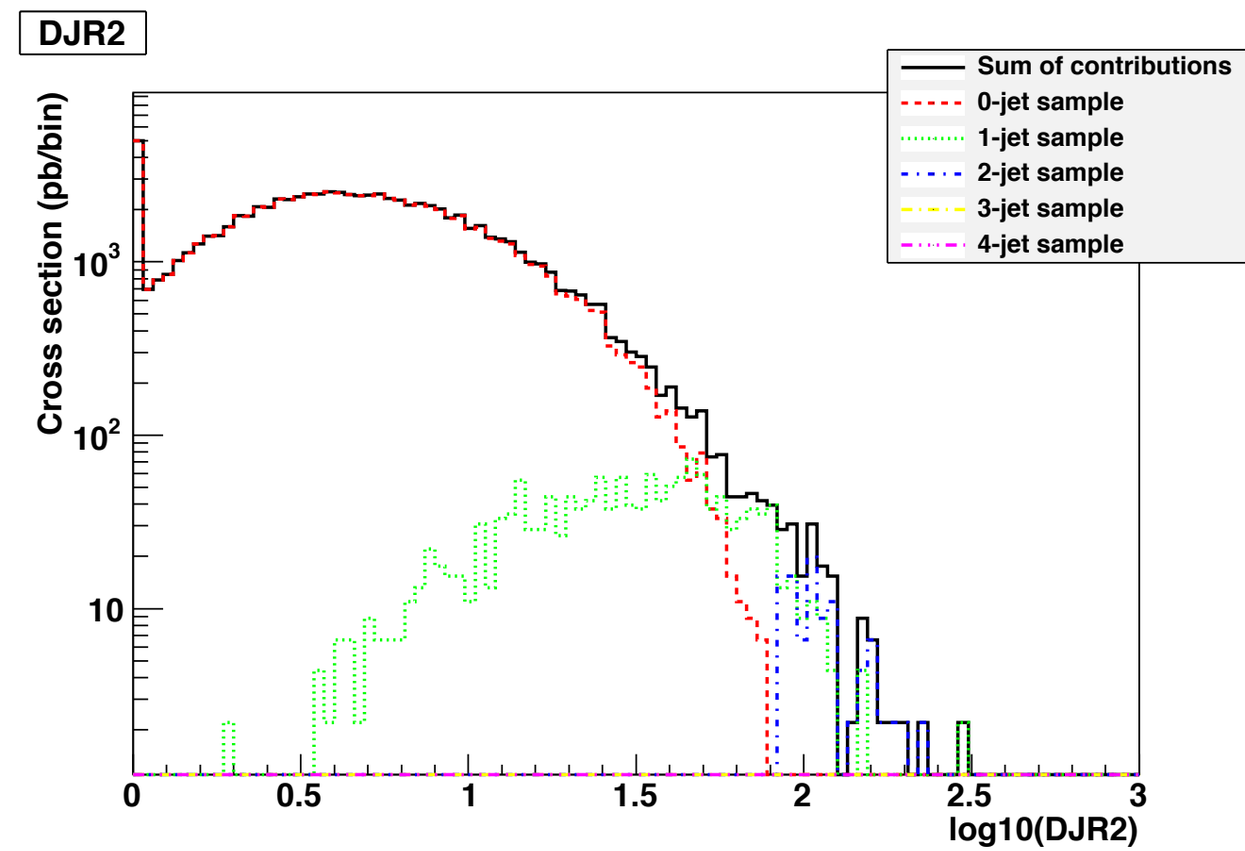




Example: W+0,1,2 jets

xqcut = 60 GeV

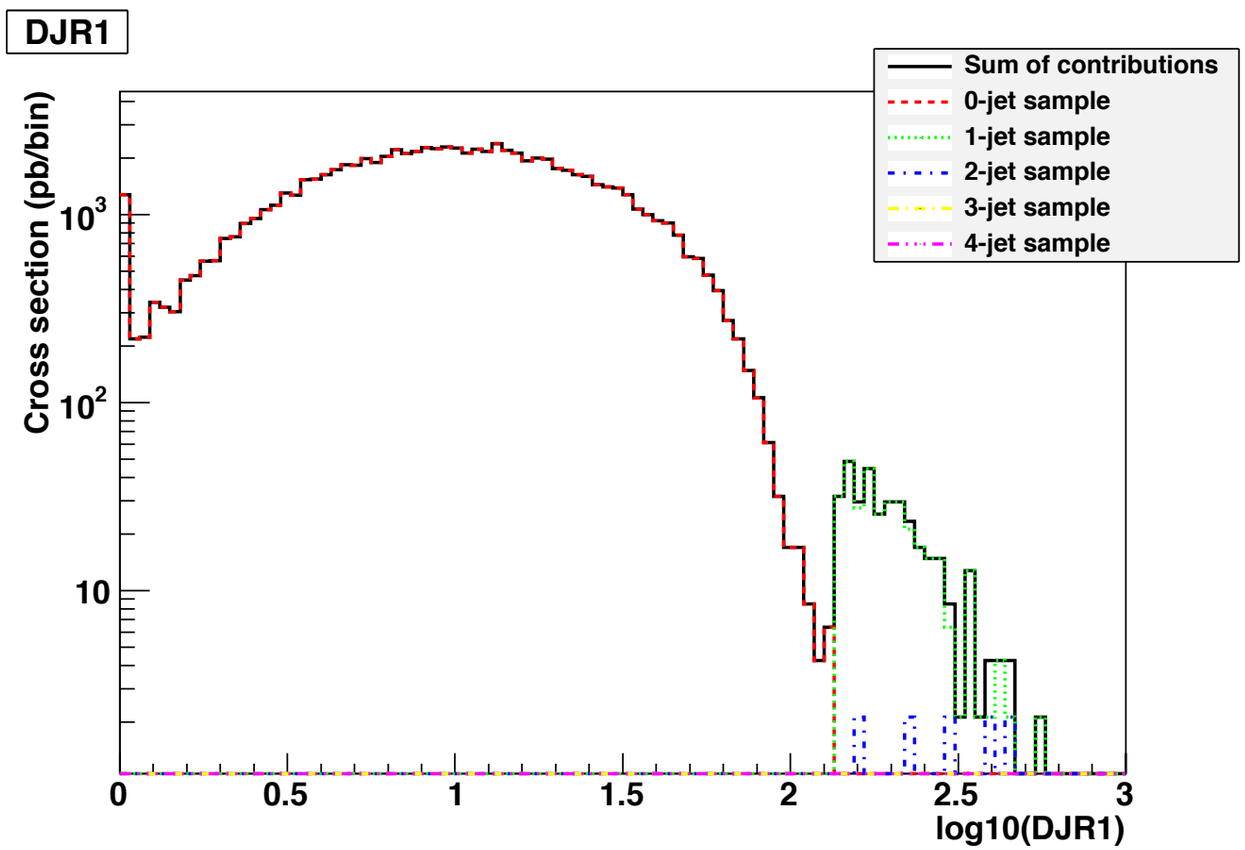


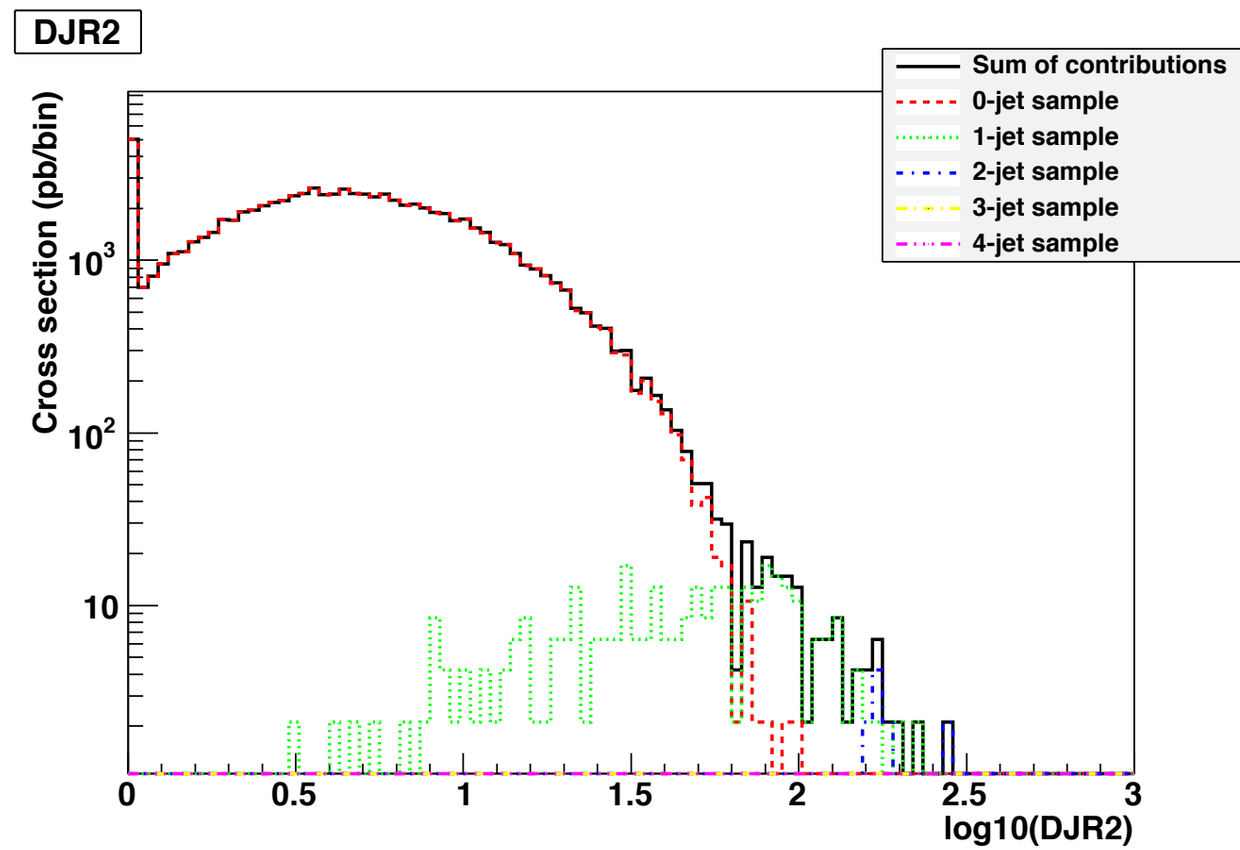


The shower struggles to generate 100 GeV jets from the 0-jet sample

Example: W+0,1,2 jets

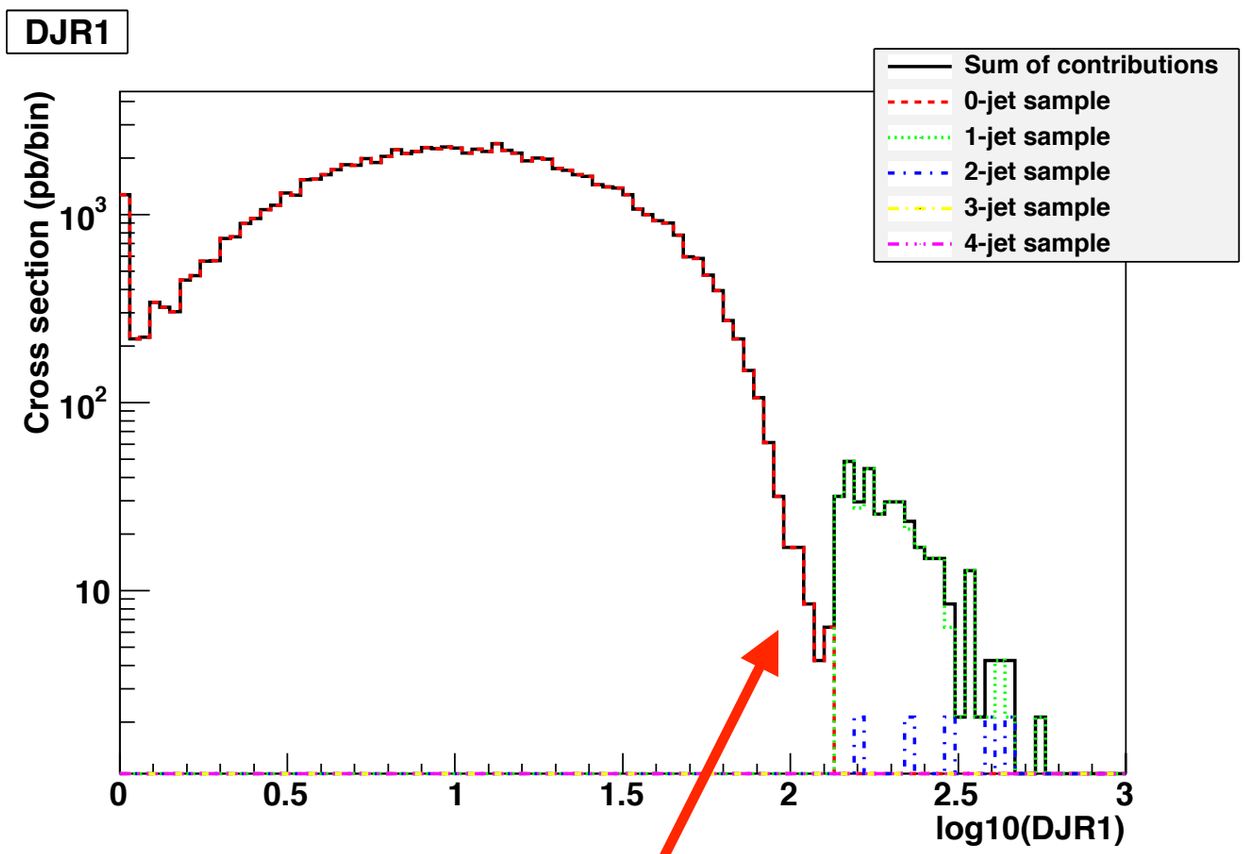
xqcut = 100 GeV

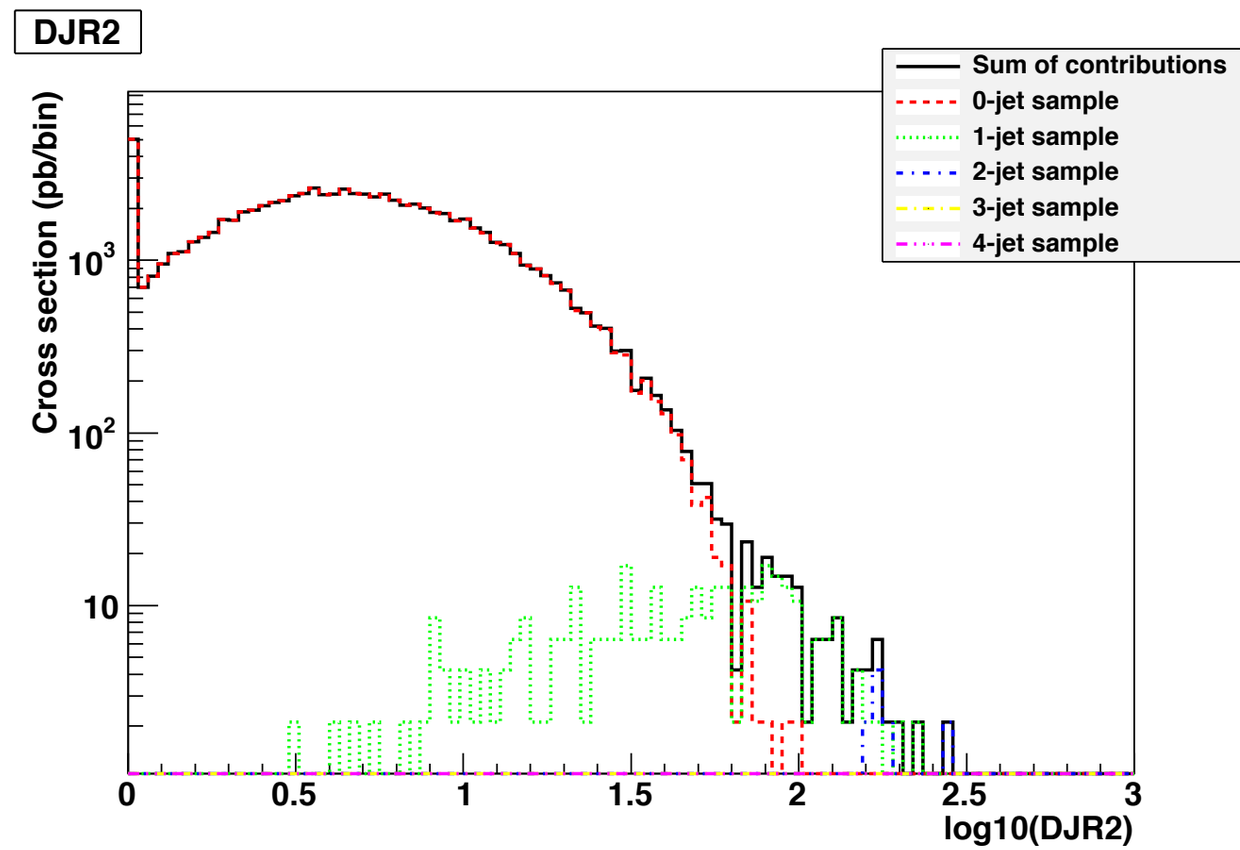




Example: W+0,1,2 jets

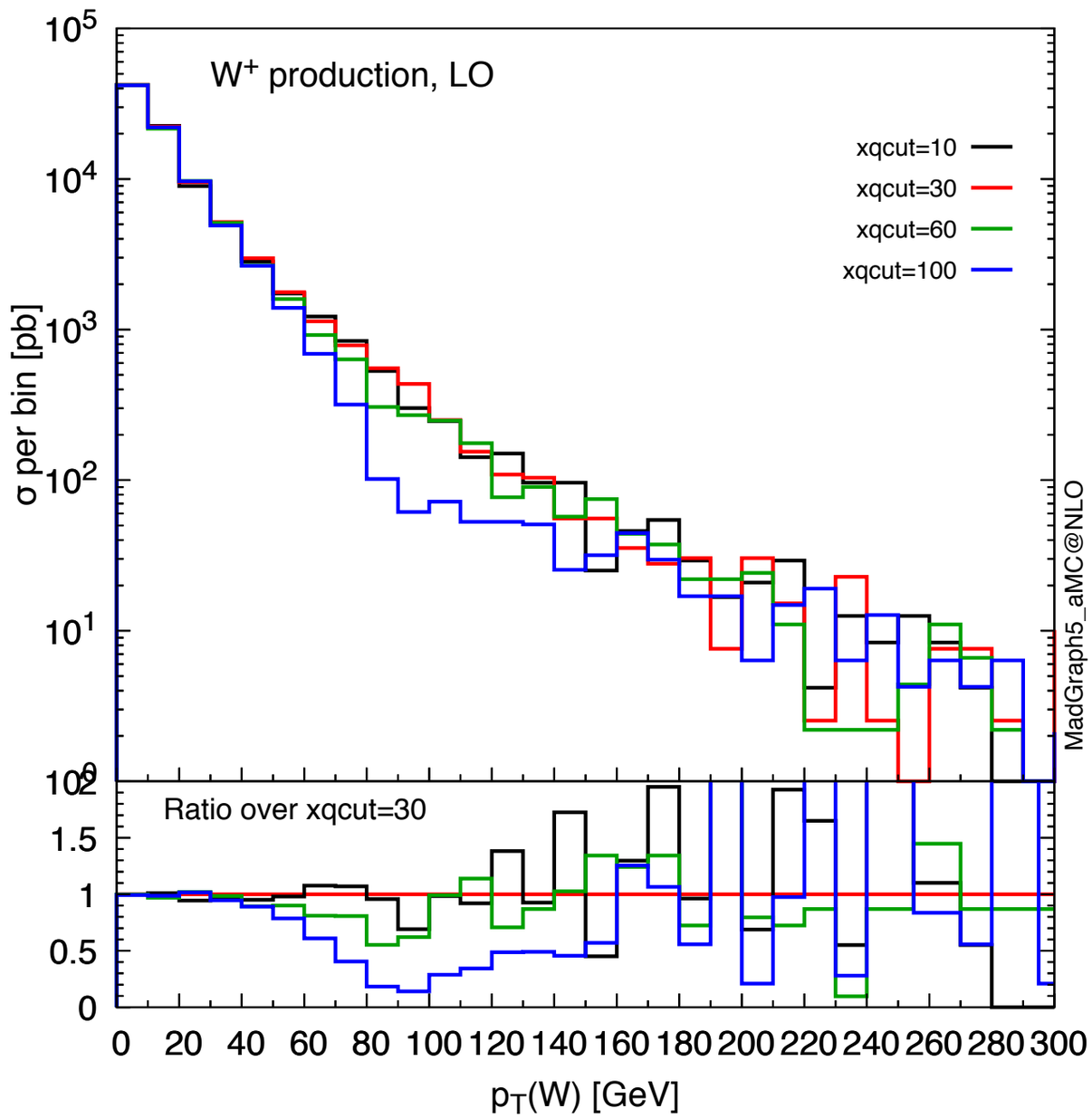
xqcut = 100 GeV



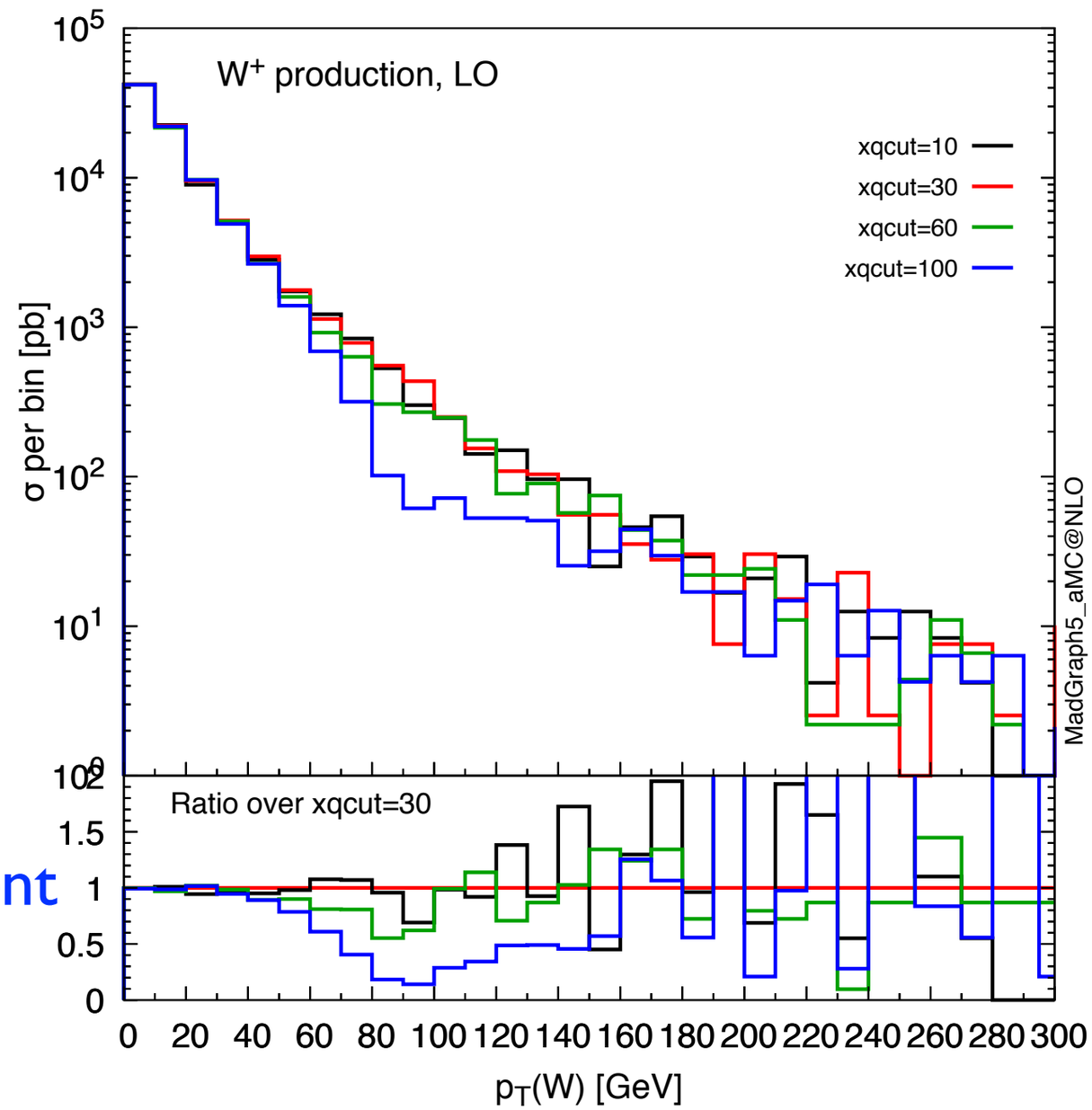


Even worse!

Example: W+0,1,2 jets

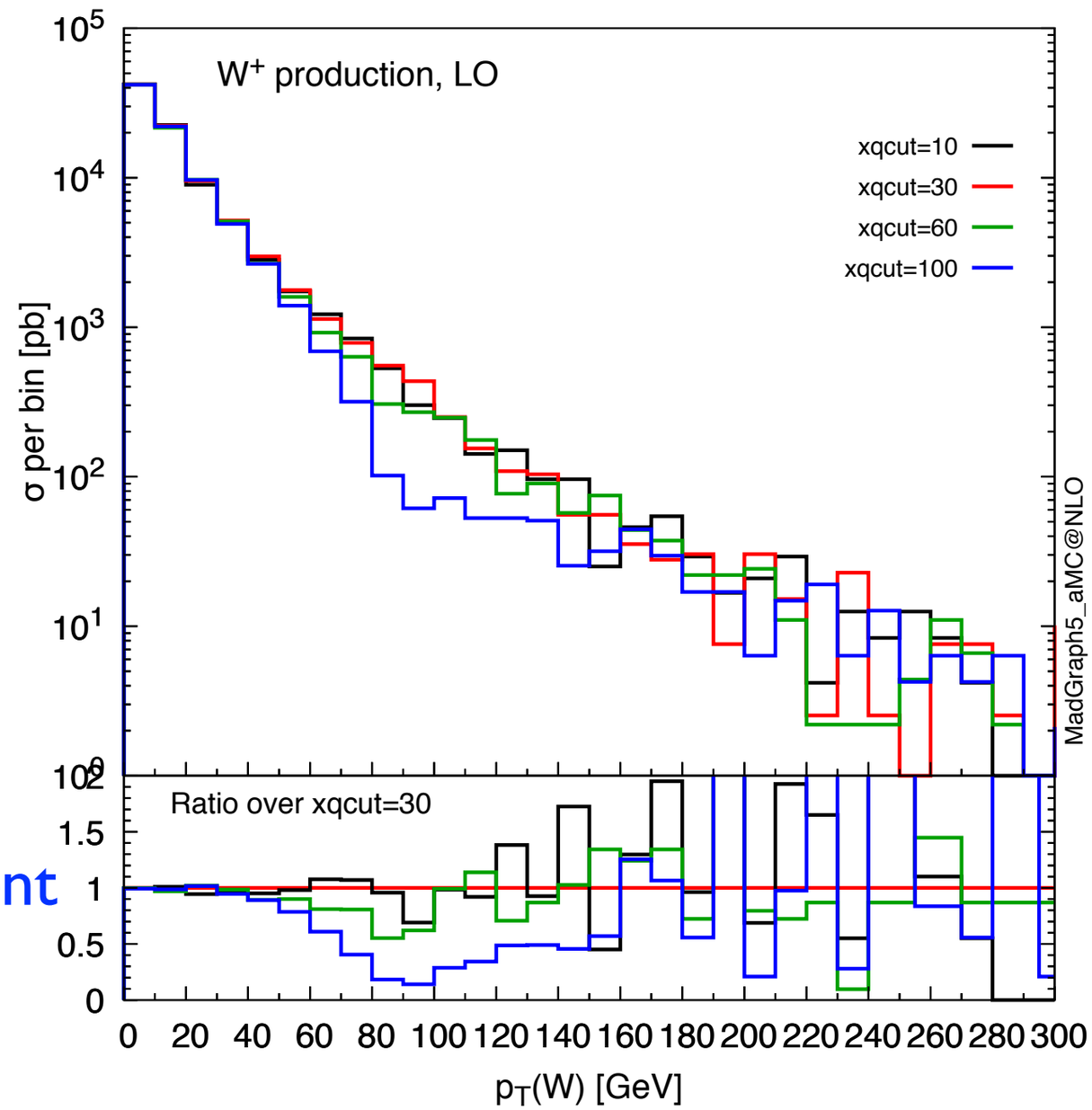


Example: W+0,1,2 jets



10 and 30 in reasonable agreement

Example: W+0,1,2 jets

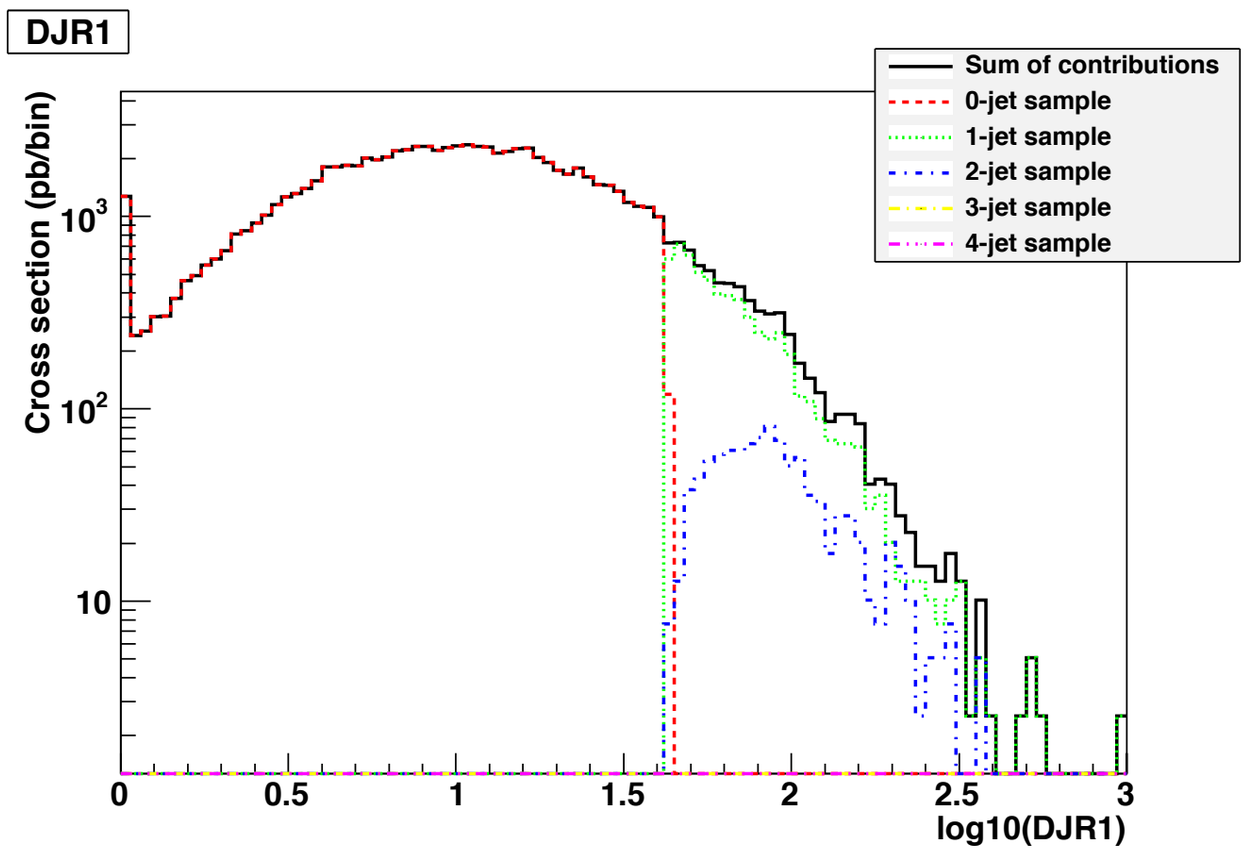


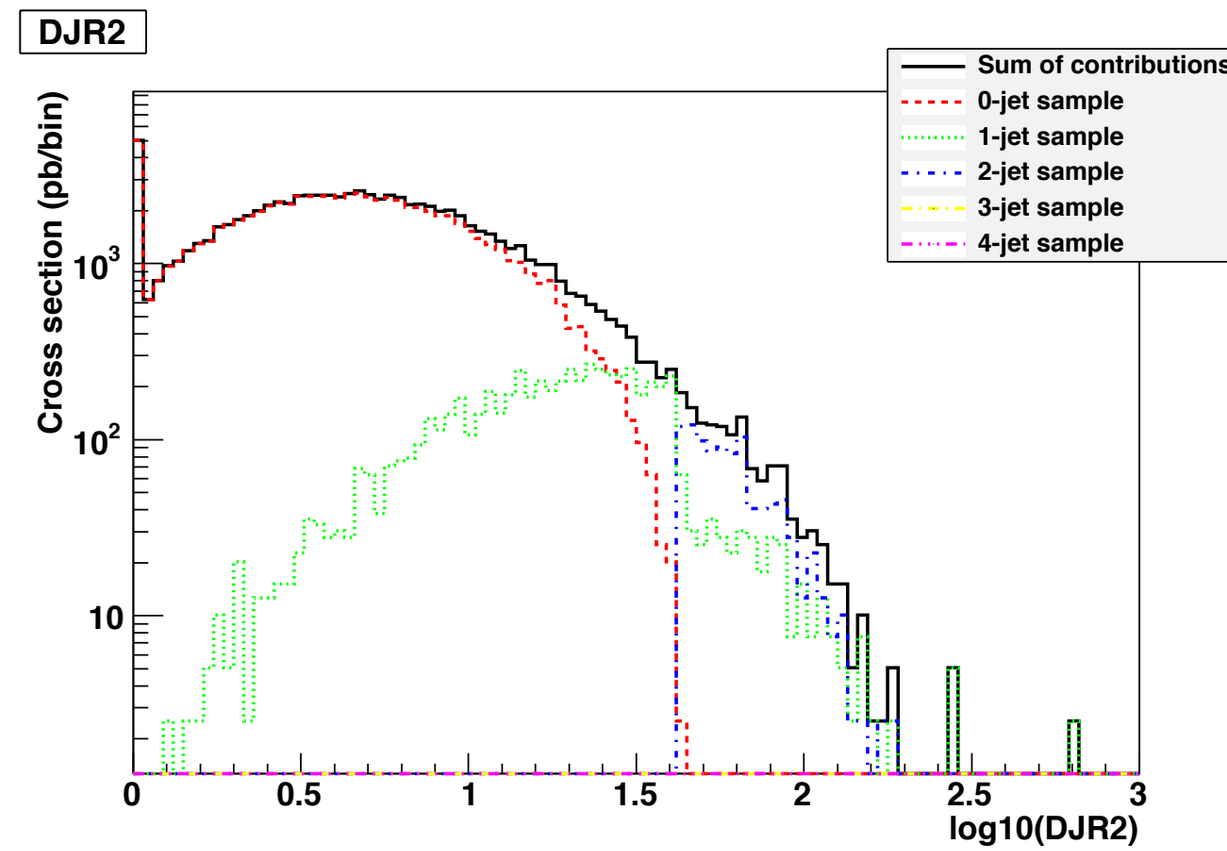
10 and 30 in reasonable agreement

60 and 100 show gap

Example: W+0,1,2 jets

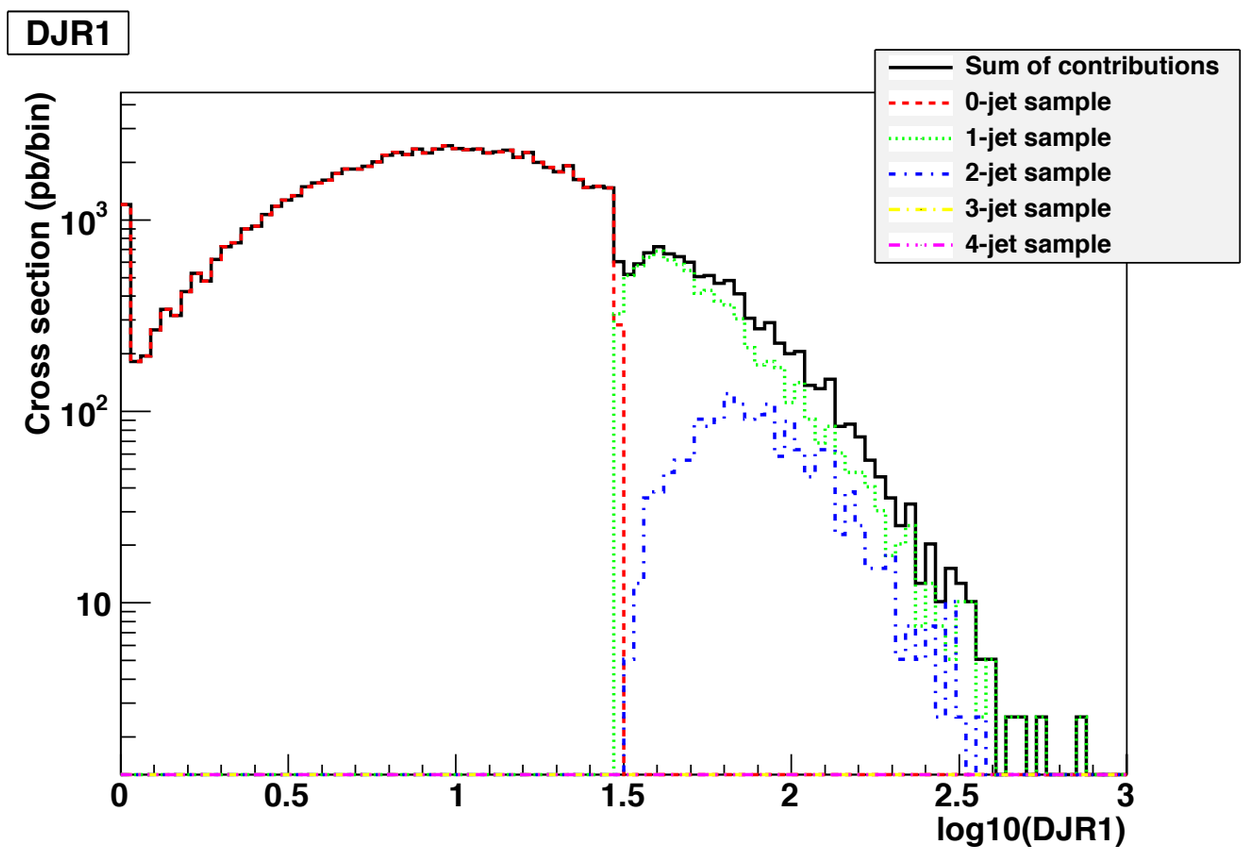
Default: `xqcut` = 30 GeV, (`Qcut` = 42 GeV)

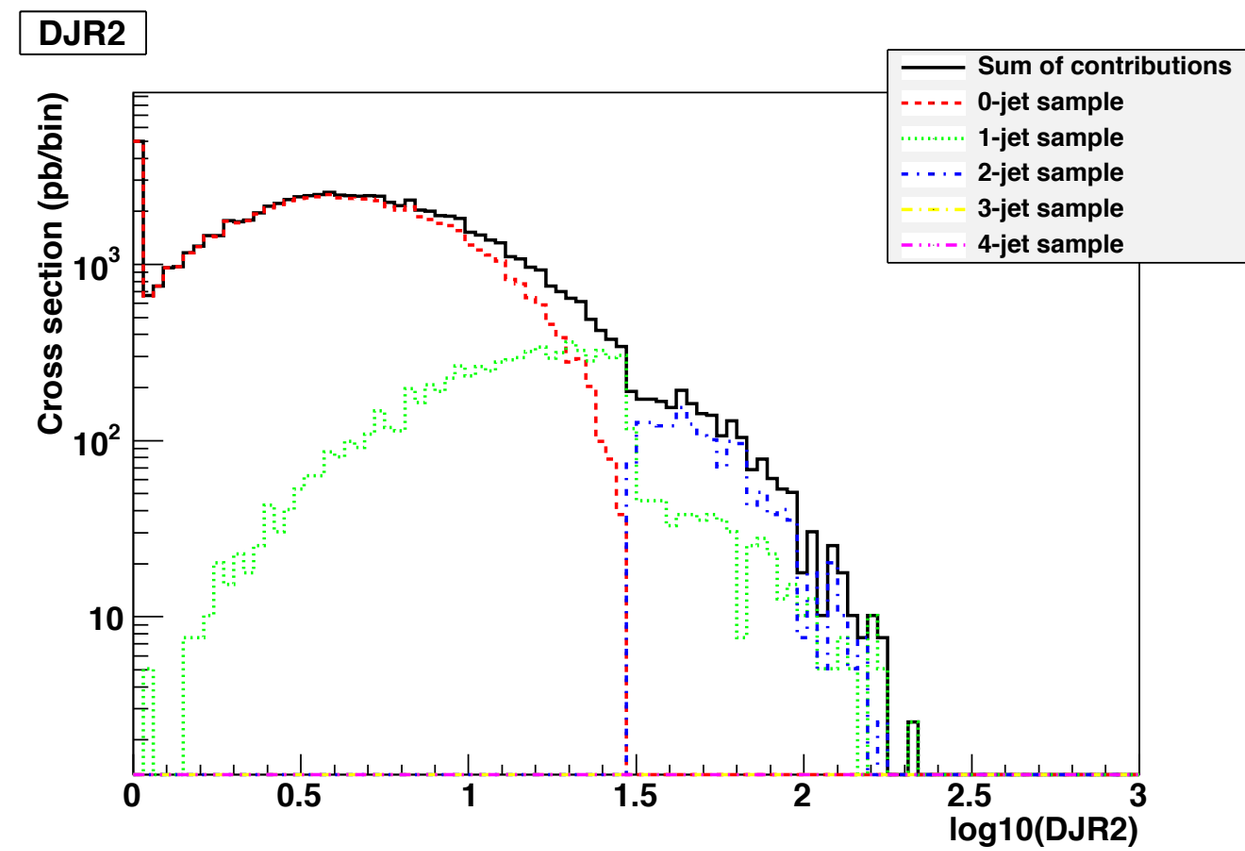




Example: W+0,1,2 jets

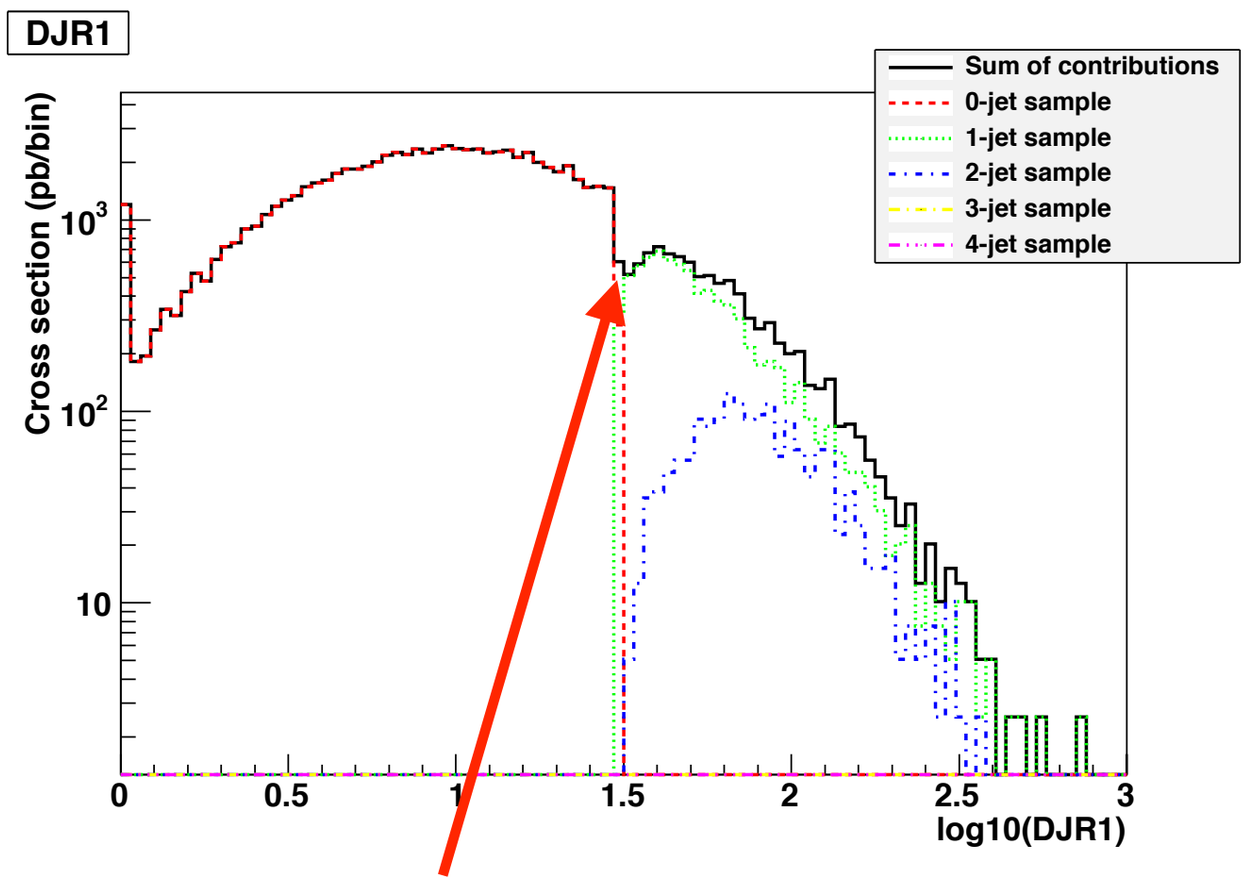
xqcut = 30 GeV, Qcut = 30 GeV

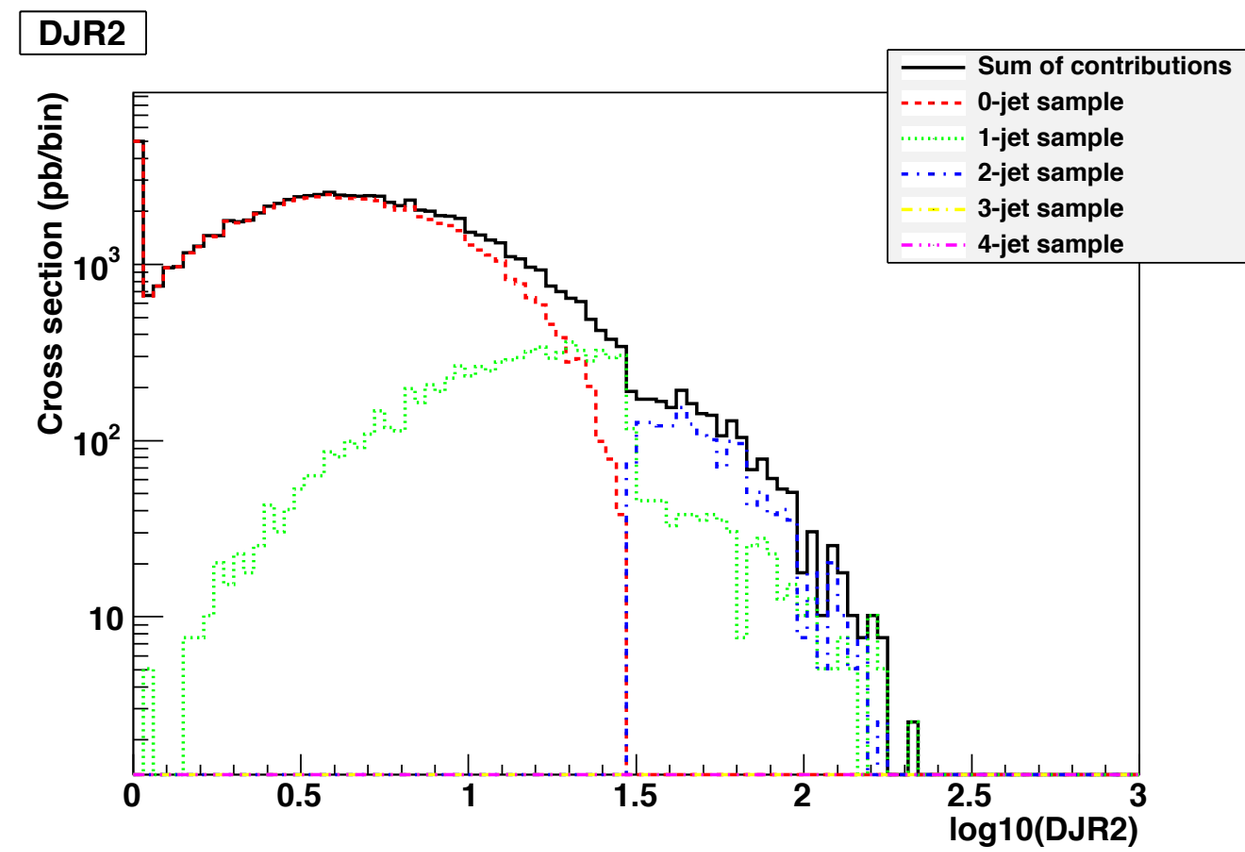




Example: W+0,1,2 jets

xqcut = 30 GeV, Qcut = 30 GeV

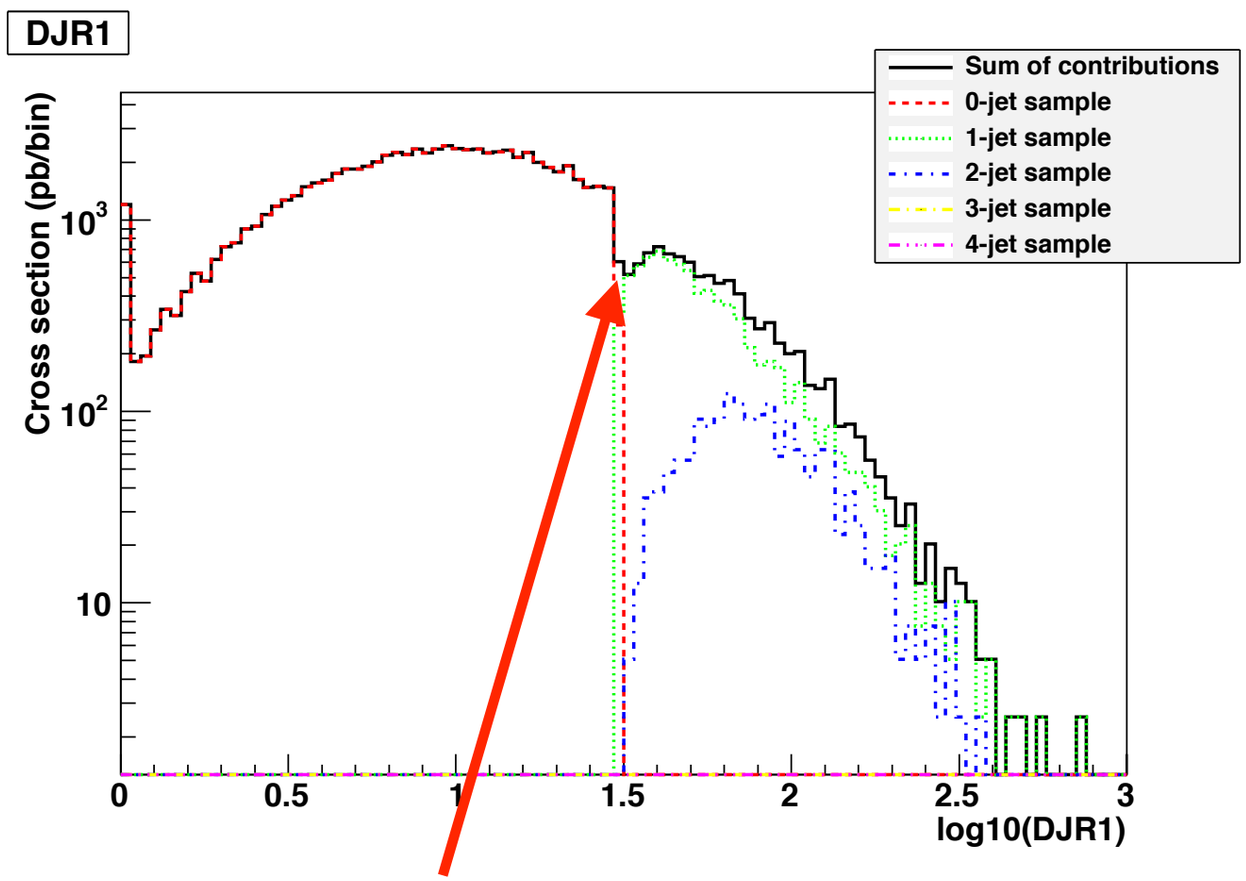


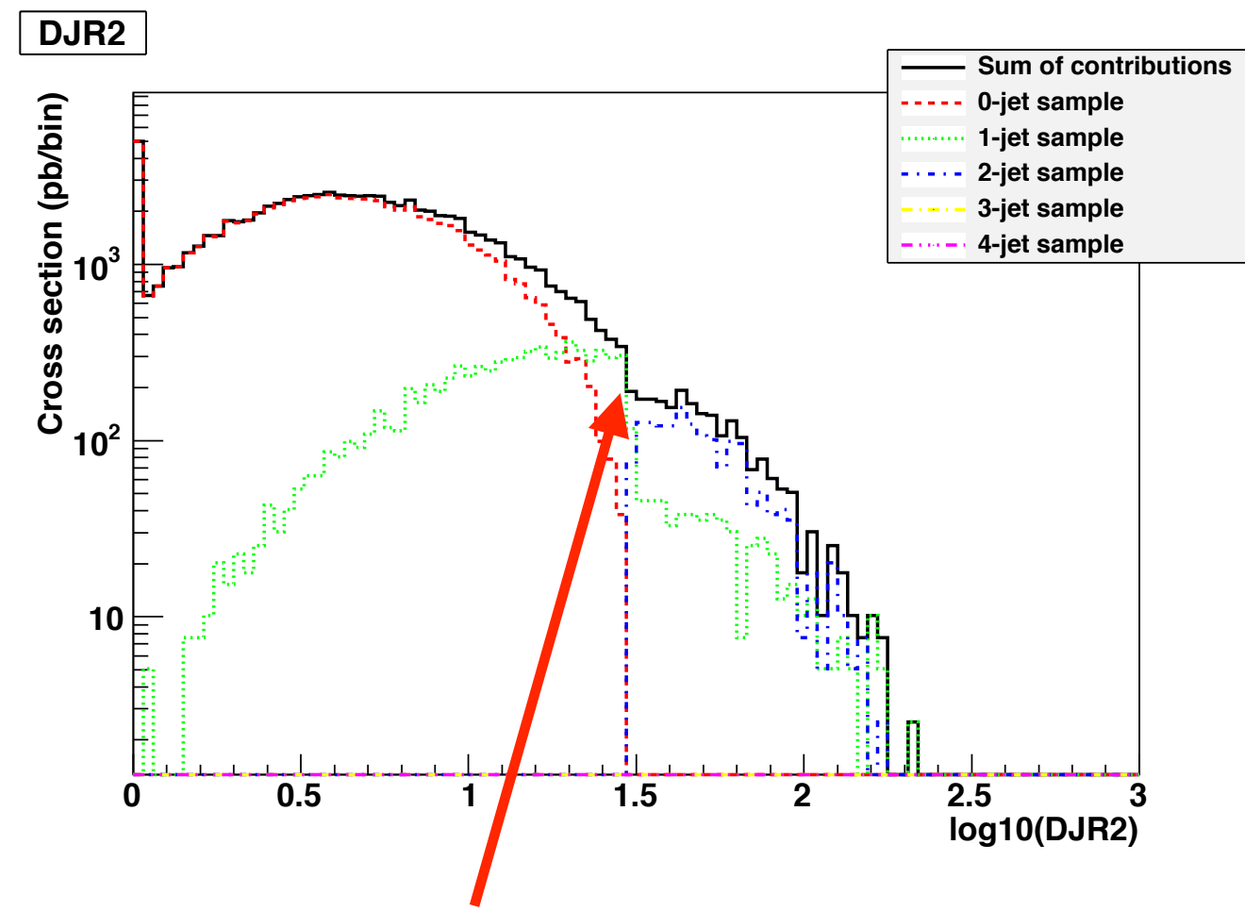


Qcut too low, the 1-jet sample does not fill here!

Example: W+0,1,2 jets

xqcut = 30 GeV, Qcut = 30 GeV



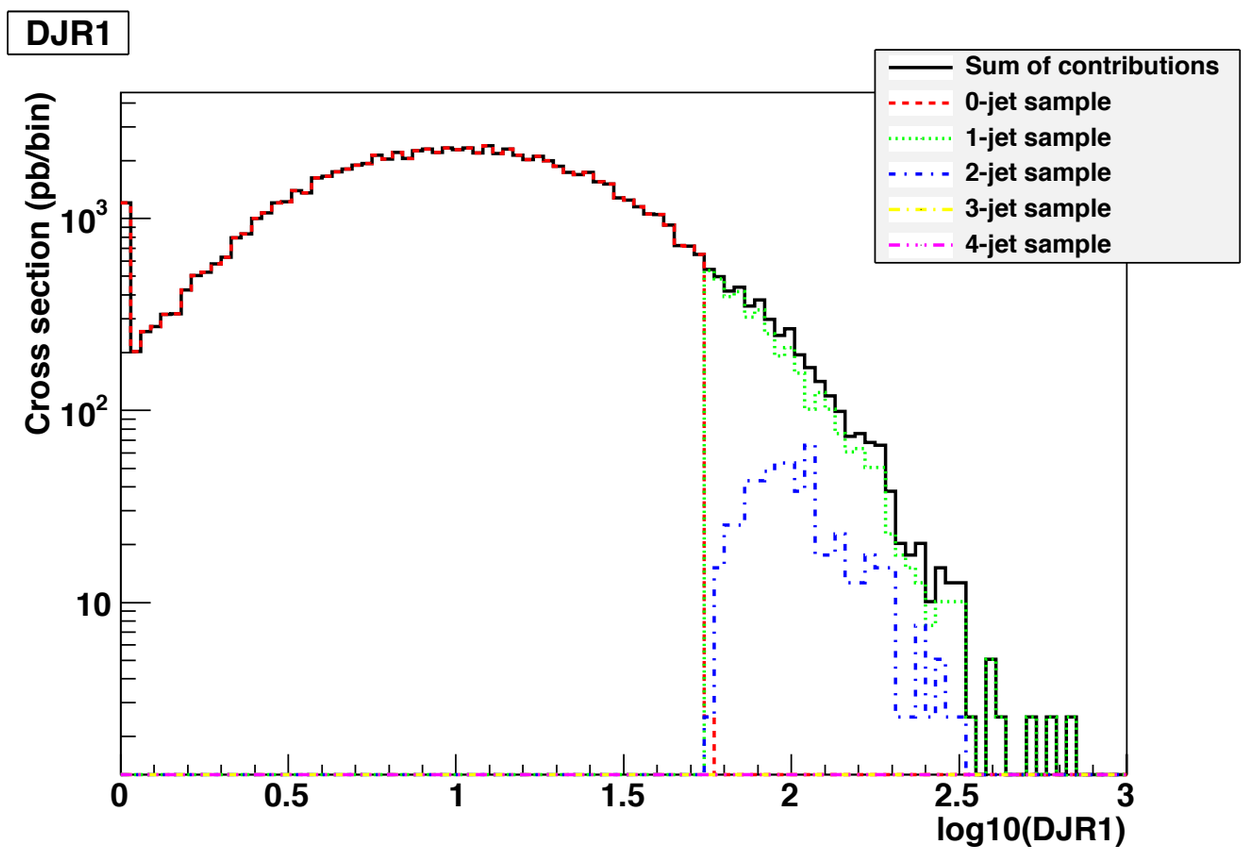


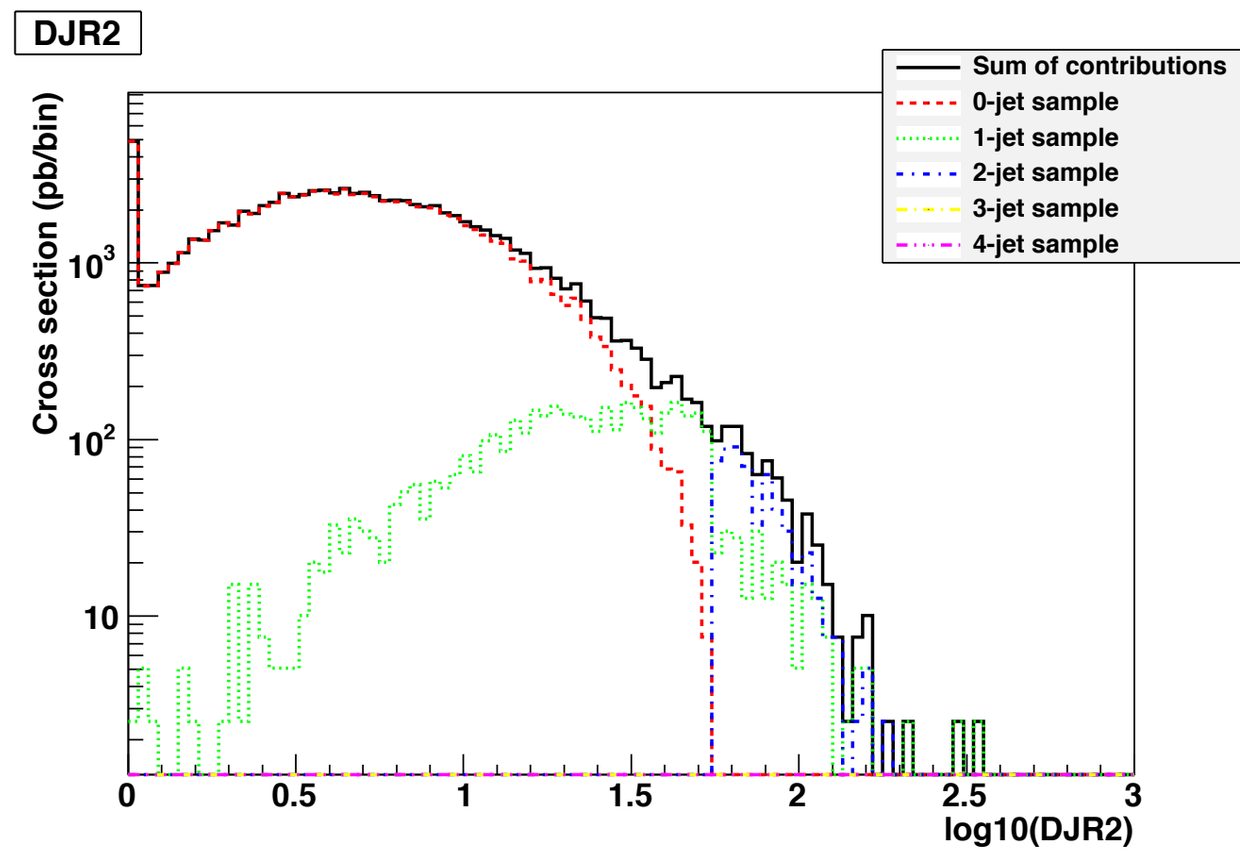
Qcut too low, the 1-jet sample does not fill here!

Qcut too low, the 2-jet sample does not fill here!

Example: W+0,1,2 jets

xqcut = 30 GeV, Qcut = 55 GeV





Matching remains smooth

Bottom line

- Choose `Qcut` around 1/6 - 1/2 * the hard scale of the process and `xqcut` accordingly smaller

What about NLO?

- Since recently (end 2012) novel merging techniques are available also at NLO
- WRT LO, at NLO one has to be more careful about double-counting configurations
- Details are too long for the tutorial, can be found in the original works **FxFx**: Frederix, Frixione arXiv:1209.6215, **UNLOPS**: Lonnblad, Prestel: 1211.7278
- Both methods available in MG5_aMC

NLO merging: How to?

- It is very similar to the LO case:

```
> generate p p > w+ [QCD] @0
> generate p p > w+ j [QCD] @1
> . . .
> output
> launch
```

- Then set `ickkw=3/4` in the `run_card` (FxFx/UNLOPS)
- Merging in Pythia8 is fully automated. With other showers (e.g. HW6) events in which jet do not match original patrons should be removed by hands. More can be found http://amcatnlo.web.cern.ch/amcatnlo/FxFx_merging.htm

NLO merging: Shower settings for Pythia8

- When doing FxFx merging in Pythia8, just change the following two variables in the `shower_card`:

```
#*********************************************************************
# FxFx merging parameters                                !ONLY FOR PYTHIA8!
#*********************************************************************
Qcut        = 40     # Merging scale
njmax       = 1      # Maximal multiplicity in the merging
```

- Alternatively, these commands should be added to the Pythia8 input file

```
! 9) Multijet merging.
JetMatching:doFxFx = on            ! switch on FxFx
JetMatching:merge = on             ! switch on MLM-merging machinery
JetMatching:qCut = 40.000           ! merging scale
JetMatching:qCutME = 30.0       ! minimum-jet(pT) cut
JetMatching:coneRadius = 1.0       ! jet radius definition
JetMatching:etaJetMax = 1000.0     ! max jet rapidity definition
JetMatching:nJetMax = 1       ! max multiplicity
JetMatching:scheme = 1             ! MadGraph MLM-type merging
JetMatching:setMad = off           ! Don't read from LHE in MG5LO format
```

Merging exercise

- Generate a merged sample (up to +1 jet) for Higgs production in gluon fusion in the HEFT.
  !!remember to set `plot_decayed yes` in the `plot_card`!!
- If you can access a small cluster:
  - do it also with loop-induced processes
  - do it at NLO with FxFx merging
    - use the NLO heft model from here:
      http://feynrules.irmp.ucl.ac.be/wiki/HiggsCharacterisation

```
From the README file
> import model HC_NLO_X0_UFO-heft
> generate p p > x0 (j ...) / t [QCD]
```

- Compare the results with those from the Loop-induced exercise

Merging exercise: solution

- Generate a merged sample (up to +1 extra jet) for Higgs production in gluon fusion in the HEFT.

```
> import model heft
> generate p p > h @0
> add process p p > h j @1
> output hj_heft_merged01
> launch
```

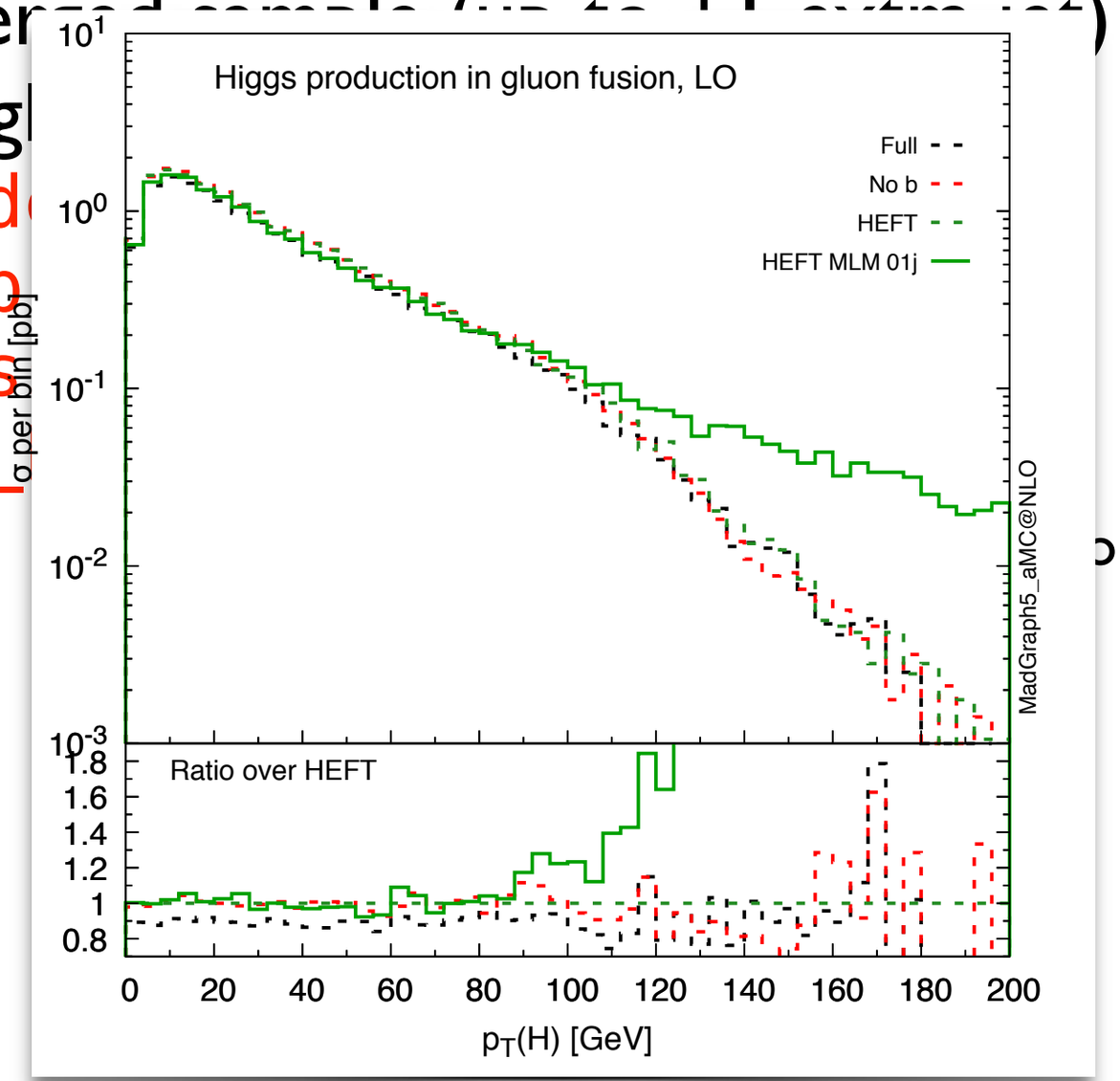
`ickkw` is already set to 1 in the `run_card`

Merging exercise: solution

- Generate a merged sample (up to 1 extra jet) for Higgs production in gl

```
> import mod
> generate p
> add process
> output hj_
> launch
```

o 1 in the run_card



Merging exercise: solution

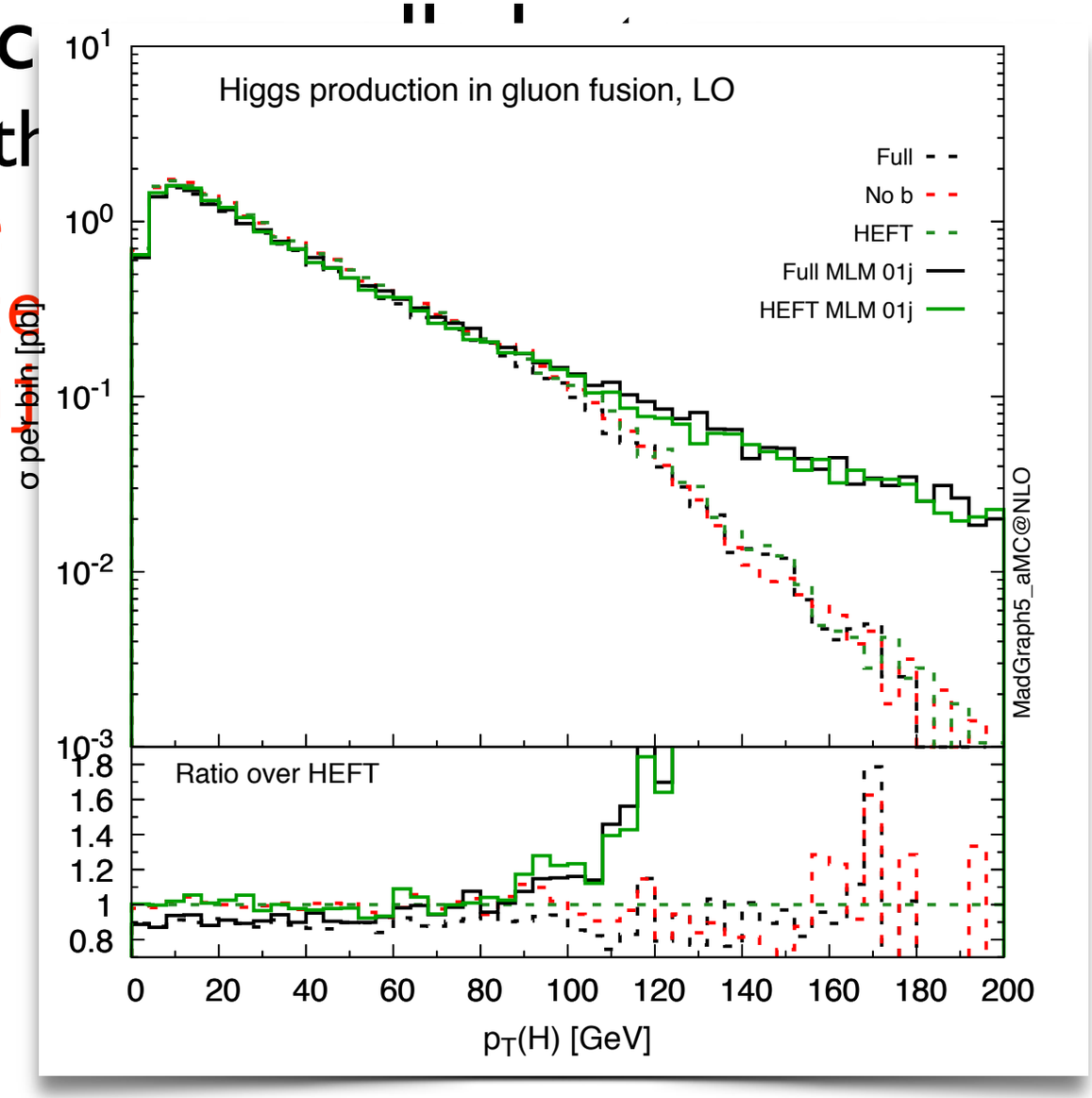
- If you can access a small cluster:
  - do it also with loop-induced processes

```
> generate p p > h [QCD] @0
> add process p p > h j [QCD] @1
> output hj_loopind_merged01
> launch
```

Merging exercise: solution

- If you can acc
  - do it also with

```
> generate
> add proce
> output h
> launch
```



Merging exercise: solution

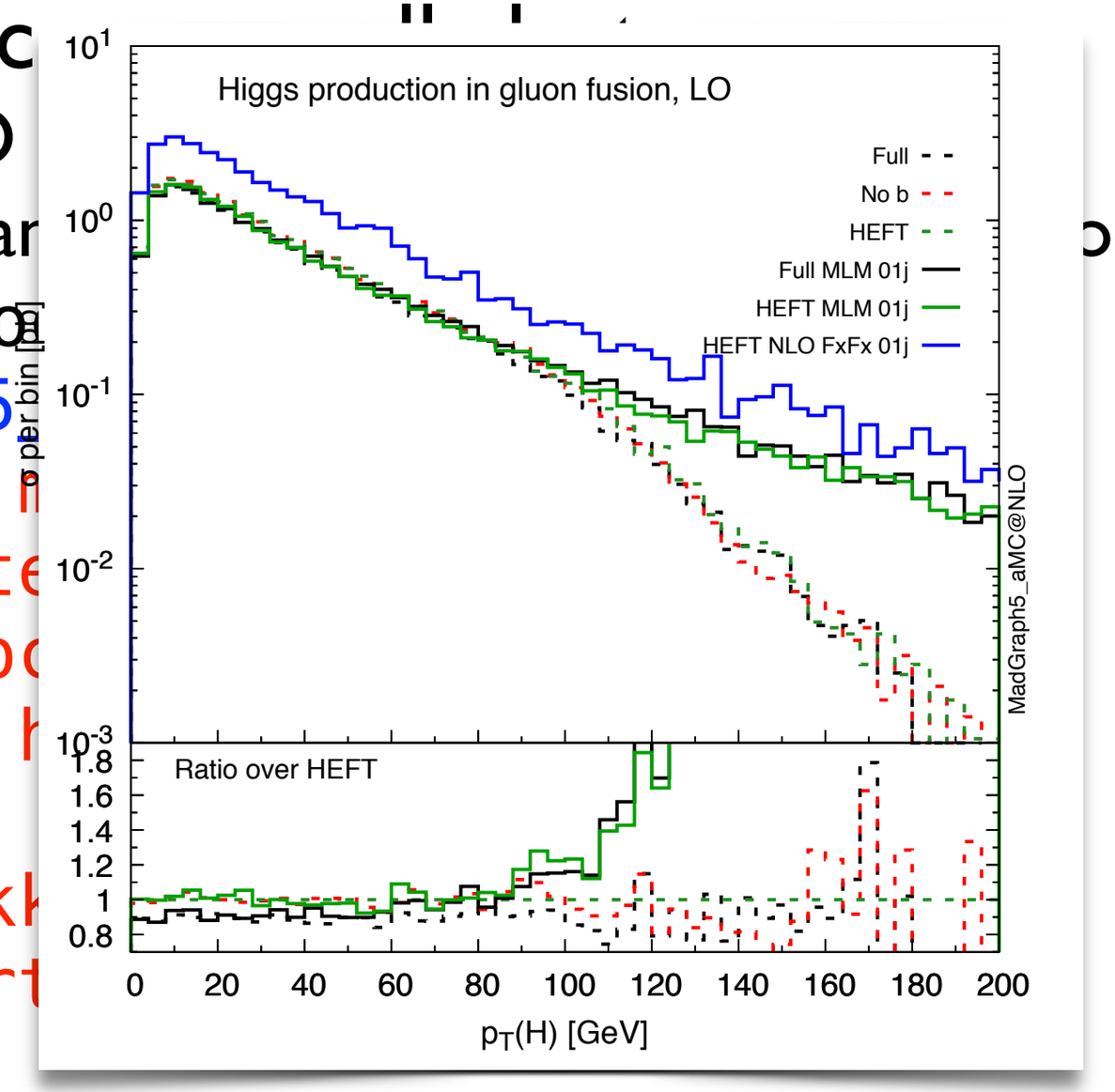
- If you can access a small cluster:
  - do it at NLO with FxFx merging
    - Download and untag the `HC_NLO_X0_UFO` model, and copy the dir inside `models/`

```
./bin/MG5_aMC
> import model HC_NLO_X0_UFO-heft
> generate p p > x0 / t [QCD] @0
> add process p p > x0 j / t [QCD] @1
> output hj_NLO_merged01
> launch
> set ickkw 3 # For FxFx
> set parton_shower PYTHIA8
```

Merging exercise: solution

- If you can acc
  - do it at NLO
    - Download ar odel, and copy the dir inside mo

```
./bin/MG5
> import m
> generate
> add proc
> output h
> launch
> set ickk
> set part
```



@1

Part 6:
Reweight unleashed

Reweighting event samples

- Experimentalist's problem: Event generation, shower and detector simulation can be very time-consuming operations
- Theorist's #1 (QCD) problem: computation of uncertainties means to perform different runs with different parameters (scales, PDFs, α_s, masses, …)
- Theorist's #2 (BSM) problem: doing a parameter scan and generating event samples
- Theorist's #3 problem: generating events with slow matrix elements

Reweighting event samples

- Idea of reweighting is simple: one event file, with several weights per event, corresponding to uncertainties, parameter-scan, …
- How to?
  - Start from one event (weight + particle info), corresponding to a given matrix element / parameter set M_{orig}
  - The new weight corresponding to a new matrix element M_{new} will be given by

$$W_{new} = \frac{|M_{new}|^2}{|M_{orig}|^2} W_{orig}$$

  - It works both if the original events are weighted or unweighted
  - The new sample will be weighted, with a (slightly) larger variance, depending on the variance of $|M_{new}/M_{orig}|^2$
  - This way the problems of the previous slides can be solved

The evil is in the details (again)

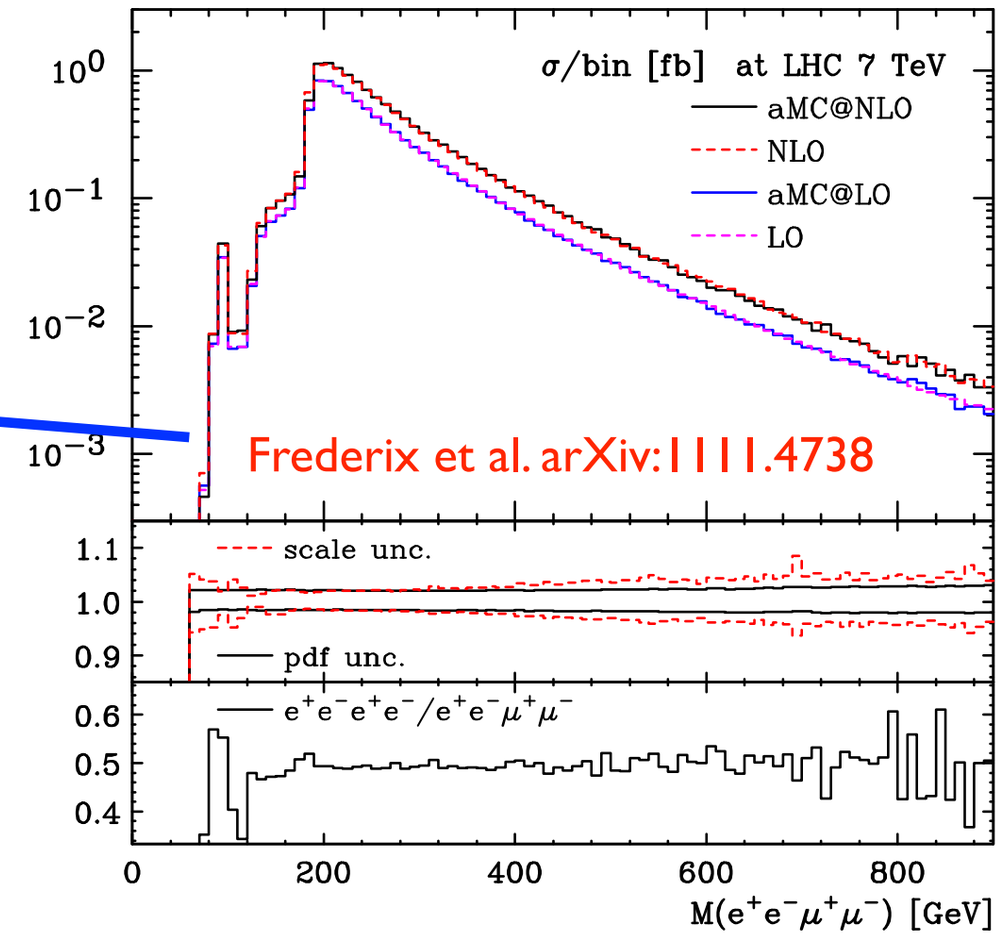
- This procedure is formally correct only for LO events
- At NLO one has both n-body (**S**) and n+1-body (**H**) events
- **S**-events contain Born, Virtual and (MC) counterterms (all have n-body kinematics)
- **H**-events contain Real (n+1) and MC (n) counterterms
- Intermediate contributions from different matrix-elements/kinematics configurations have to be stored in the event file

Reweighting event samples

- Experimentalist's problem: Event generation, shower and detector simulation can be very time-consuming operations
- Theorist's #1 (QCD) problem: computation of uncertainties means to perform different runs with different parameters (scales, PDFs, α_s, masses, …)
- Theorist's #2 (BSM) problem: doing a parameter scan and generating event samples
- Theorist's #3 problem: generating events with slow matrix elements

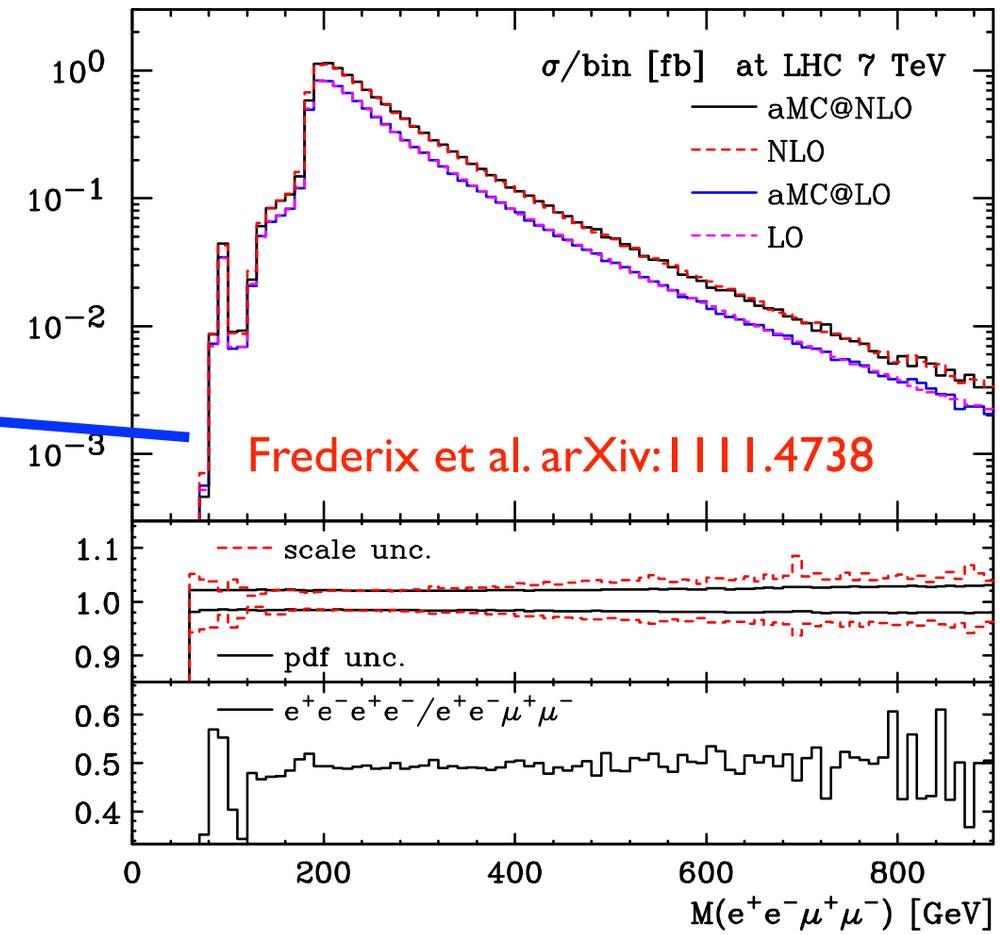
Reweighting event samples

- Experimentalist's problem: Event generation, shower and detector simulation can be very time-consuming operations
- Theorist's #1 (QCD) problem: computation of uncertainties means to perform different runs with different parameters (scales, PDFs, α_s, masses, …)
- Theorist's #2 (BSM) problem: doing a parameter scan and generating event samples
- Theorist's #3 problem: generating events with slow matrix elements

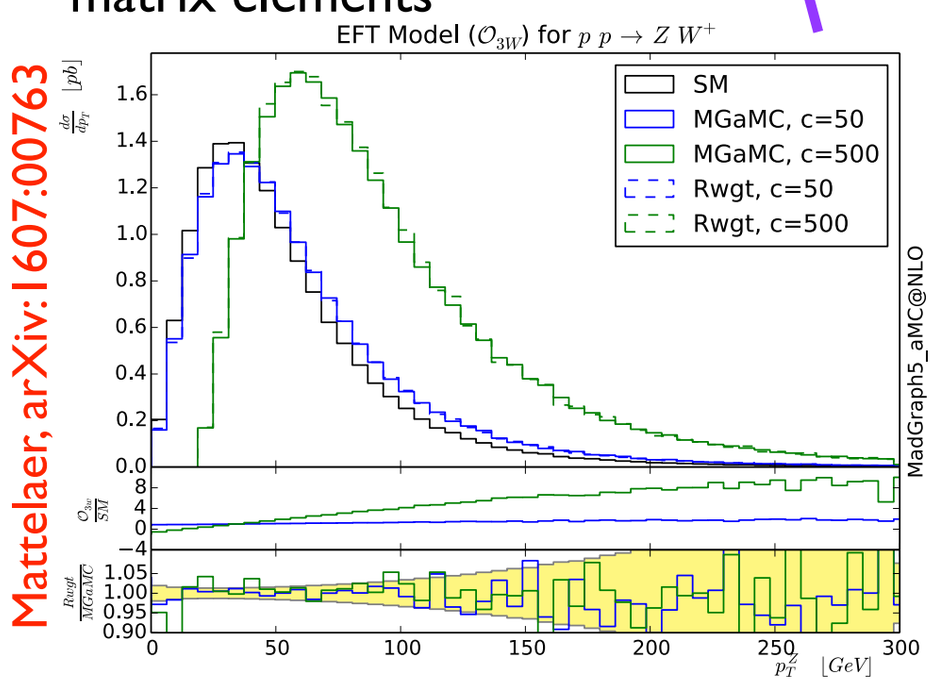


Reweighting event samples

- Experimentalist's problem: Event generation, shower and detector simulation can be very time-consuming operations
- Theorist's #1 (QCD) problem: computation of uncertainties means to perform different runs with different parameters (scales, PDFs, α_s, masses, …)
- Theorist's #2 (BSM) problem: doing a parameter scan and generating event samples
- Theorist's #3 problem: generating events with slow matrix elements



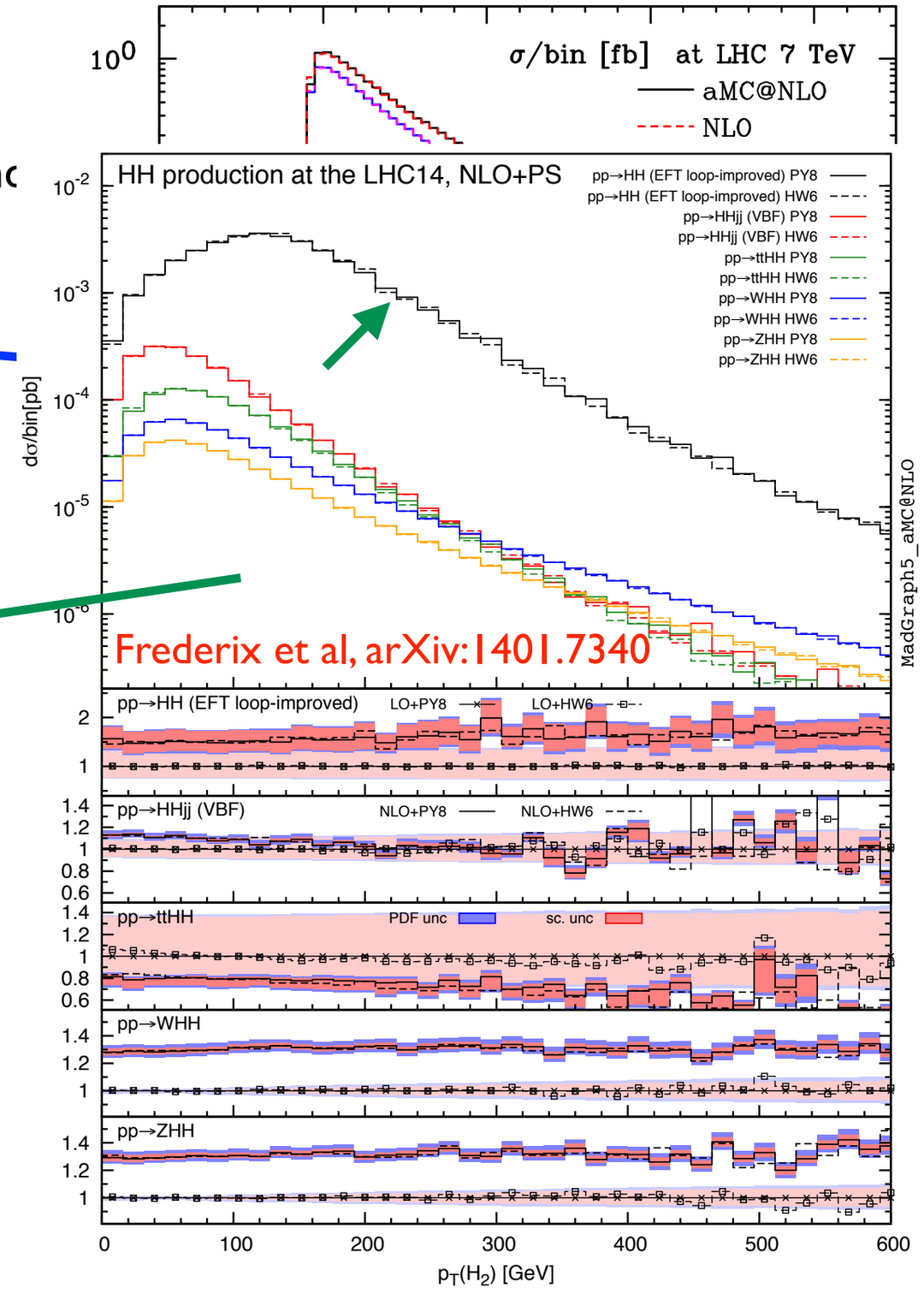
Frederix et al. arXiv:1111.4738

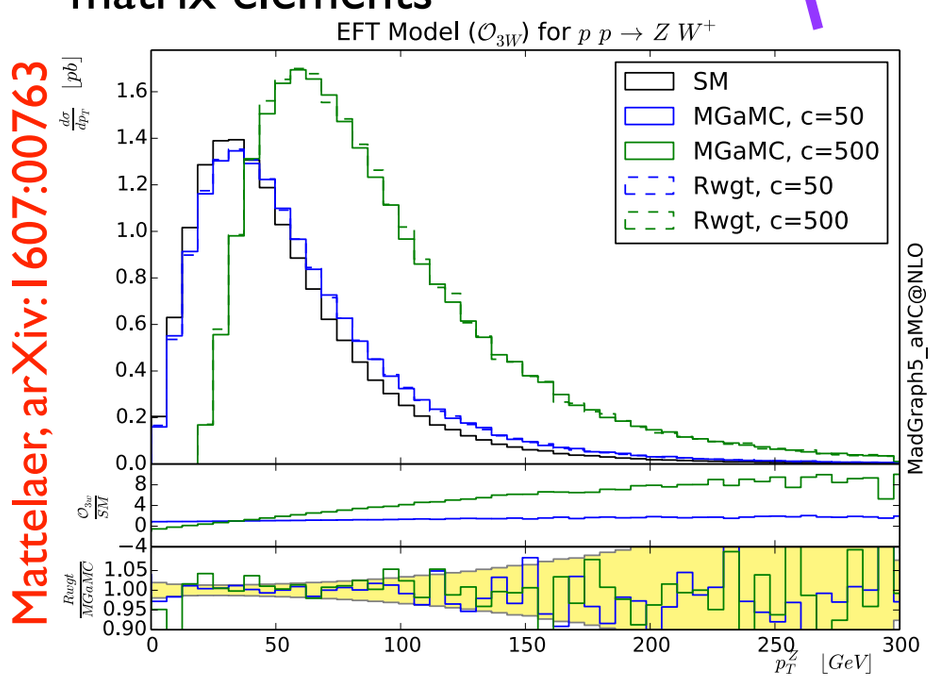


Mattelaer, arXiv:1607:00763

Reweighting event samples

- Experimentalist's problem: Event generation, shower and detector simulation can be very time-consuming operations
- Theorist's #1 (QCD) problem: computation of uncertainties means to perform different runs with different parameters (scales, PDFs, α_s, masses, …)
- Theorist's #2 (BSM) problem: doing a parameter scan and generating event samples
- Theorist's #3 problem: generating events with slow matrix elements





Mattelaer, arXiv:1607:00763

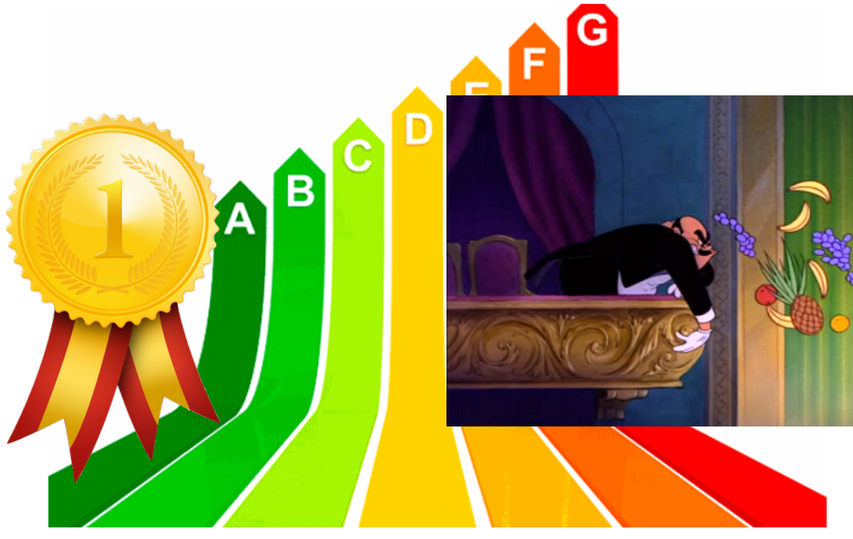
Automated reweighting in MadGraph5_aMC@NLO

Mattelaer, arXiv:1607:00763

- The three kind of reweighting (uncertainties, parameter scan and with a different matrix element) are all automated in MG5_aMC
- Both a LO and NLO reweightings can be used
  For parameter variations, LO is often as good as NLO
- More information can be found on
  https://cp3.irmp.ucl.ac.be/projects/madgraph/wiki/Reweight
- Bugs may still be there, let us know…

Farewell…

- We hope you managed to know a bit more MadGraph5_aMC@NLO, and possibly enjoyed it
- Don't hesitate to contact us for questions, requests, …
- The most effective way is via the LaunchPad page
  - Bugs: https://bugs.launchpad.net/mg5amcnlo
  - Questions: https://answers.launchpad.net/mg5amcnlo
- Again, please let us know your comments about the tutorial!



Farewell…

- We hope you managed to know a bit more MadGraph5_aMC@NLO, and possibly enjoyed it
- Don't hesitate to contact us for questions, requests, …
- The most effective way is via the LaunchPad page
  - Bugs: https://bugs.launchpad.net/mg5amcnlo
  - Questions: https://answers.launchpad.net/mg5amcnlo
- Again, please let us know your comments about the tutorial!

Thanks for coming!