Event Generation with MadGraph 5

Johan Alwall (歐友涵)

National Taiwan University

2nd Taipei MG/FR School, Sept 4-8, 2013

Outline of lectures

- Background:
  - ➡ New Physics at hadron colliders
  - ➡ Monte Carlo integration and generation
  - ➡ Simulation of collider events
- Simulations with MadGraph 5:
  - ➡ Computing the Matrix Element in MadGraph
  - ➡ Features of MadGraph 5
- Jet Matching:
  - ➡ Parton Showers
  - ➡ MLM Matching with MadGraph and Pythia

Aims for these lectures

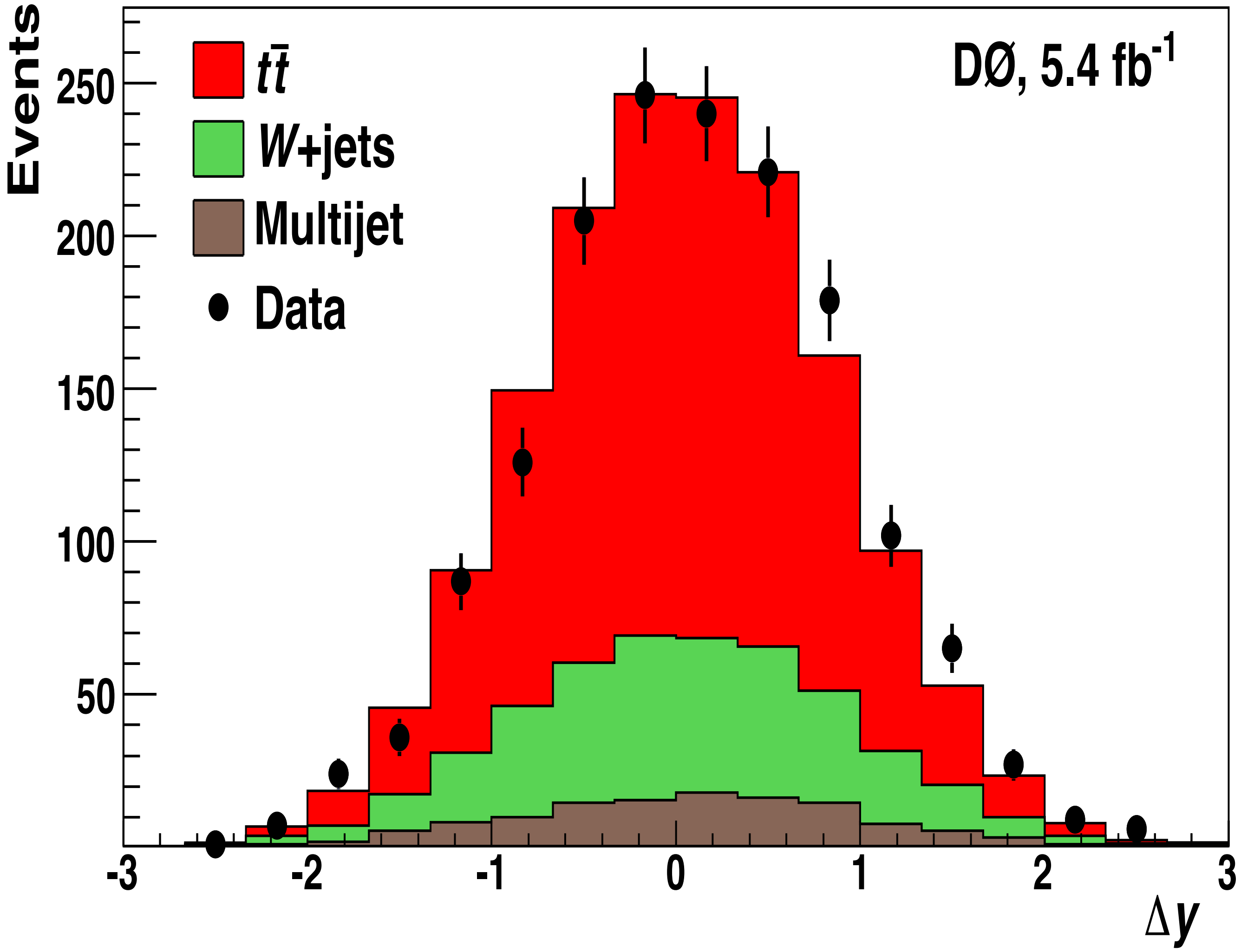
- Get you acquainted with the concepts and tools used in event simulation at hadron colliders
- Answer as many of your questions as I can (so please ask questions!)

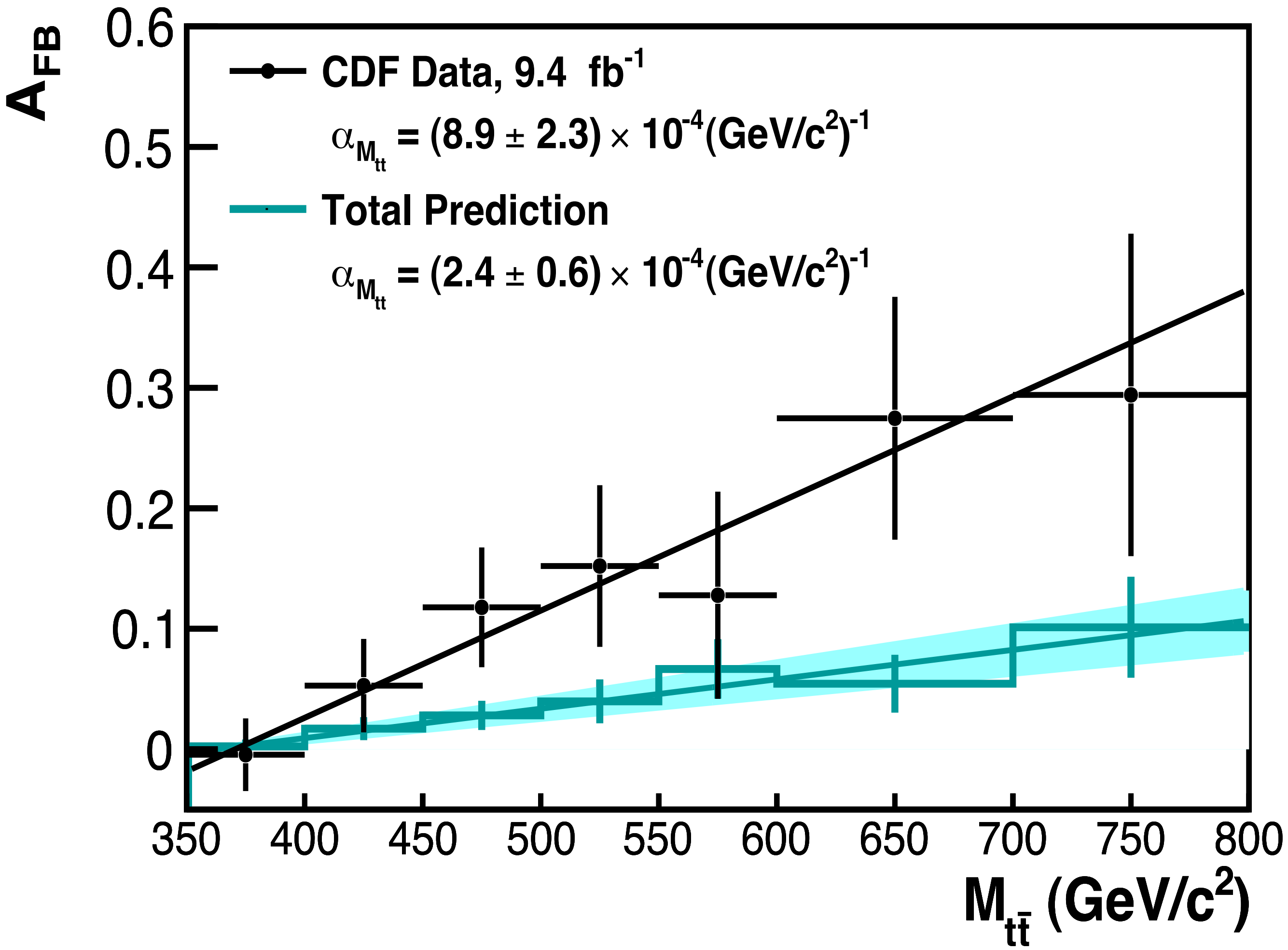
New Physics at hadron colliders

- The LHC has taken over from the Tevatron!
- Significant luminocities
  - ➡ Tevatron collected >10 fb^{-1} in the last 10 years
  - ➡ Fantastic legacy, including several interesting excesses!
  - ➡ LHC has collected 23 fb^{-1} in its 8 TeV run!
  - ➡ Ever-more stringent tests of the SM!
  - ➡ Found (what looks like) the Higgs boson in July 2012!
- How interpret excesses? How determine Standard Model backgrounds?
  - ➡ Monte Carlo simulation! (combined with data-driven methods)

Example: top-antitop asymmetry at Tevatron

CDF collaboration, arXiv:1211.1003, 1101.0034
DØ collaboration, arXiv: 1107.4995



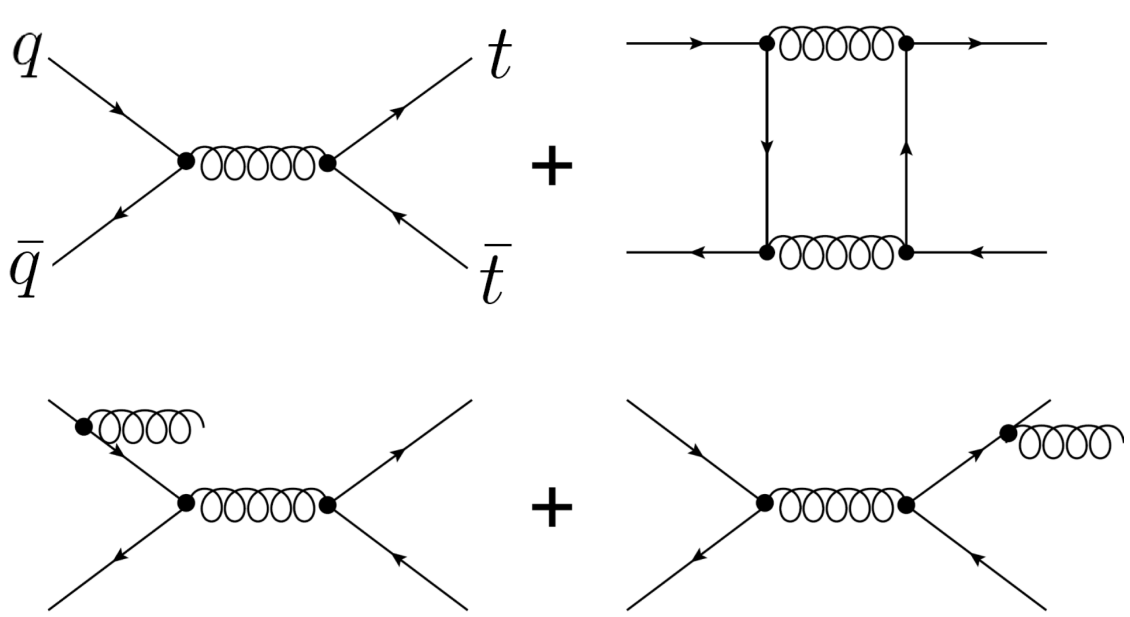


Example: top-antitop asymmetry at Tevatron

- First: Look for Standard Model explanations

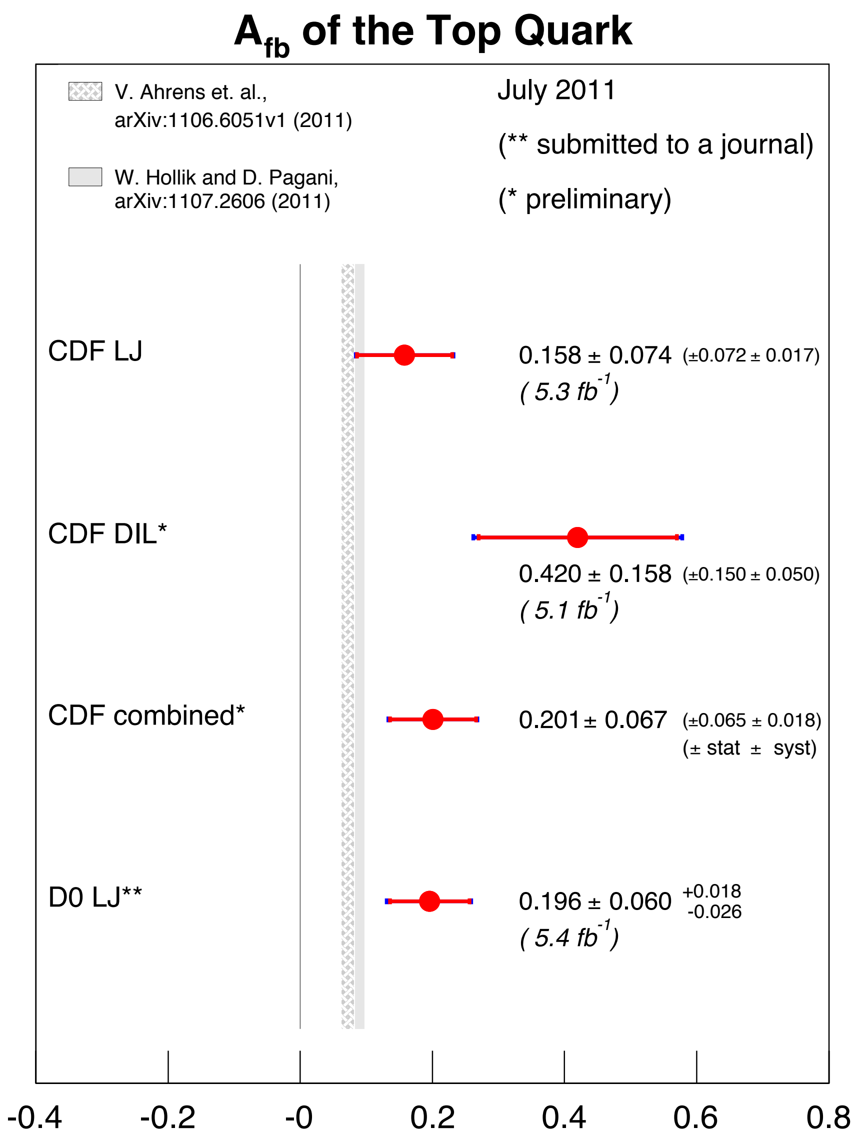
Example: top-antitop asymmetry at Tevatron

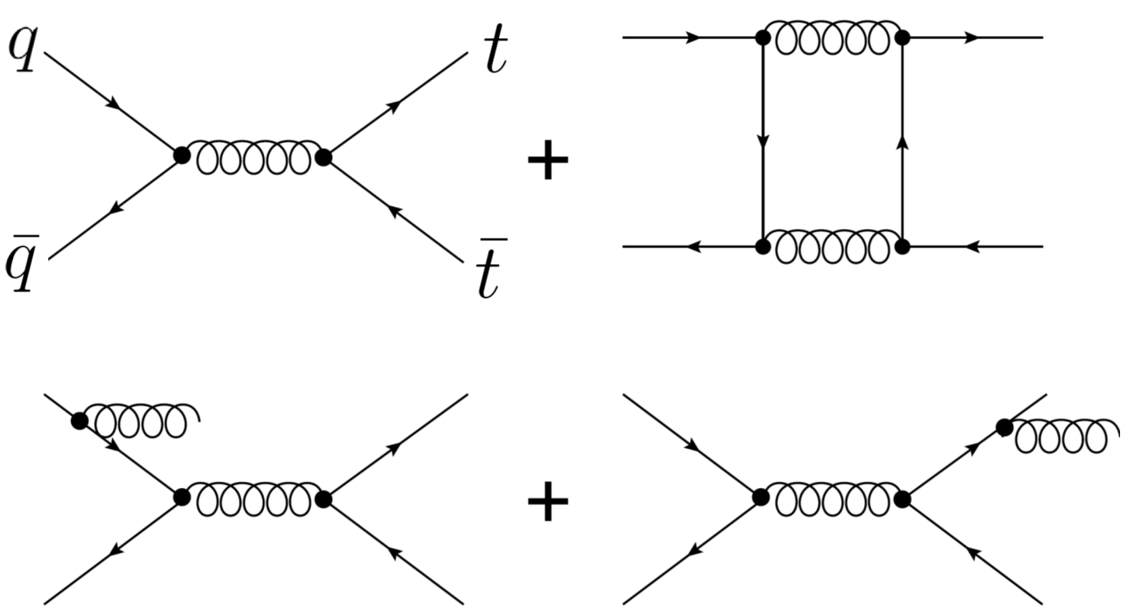
- First: Look for Standard Model explanations



Example: top-antitop asymmetry at Tevatron

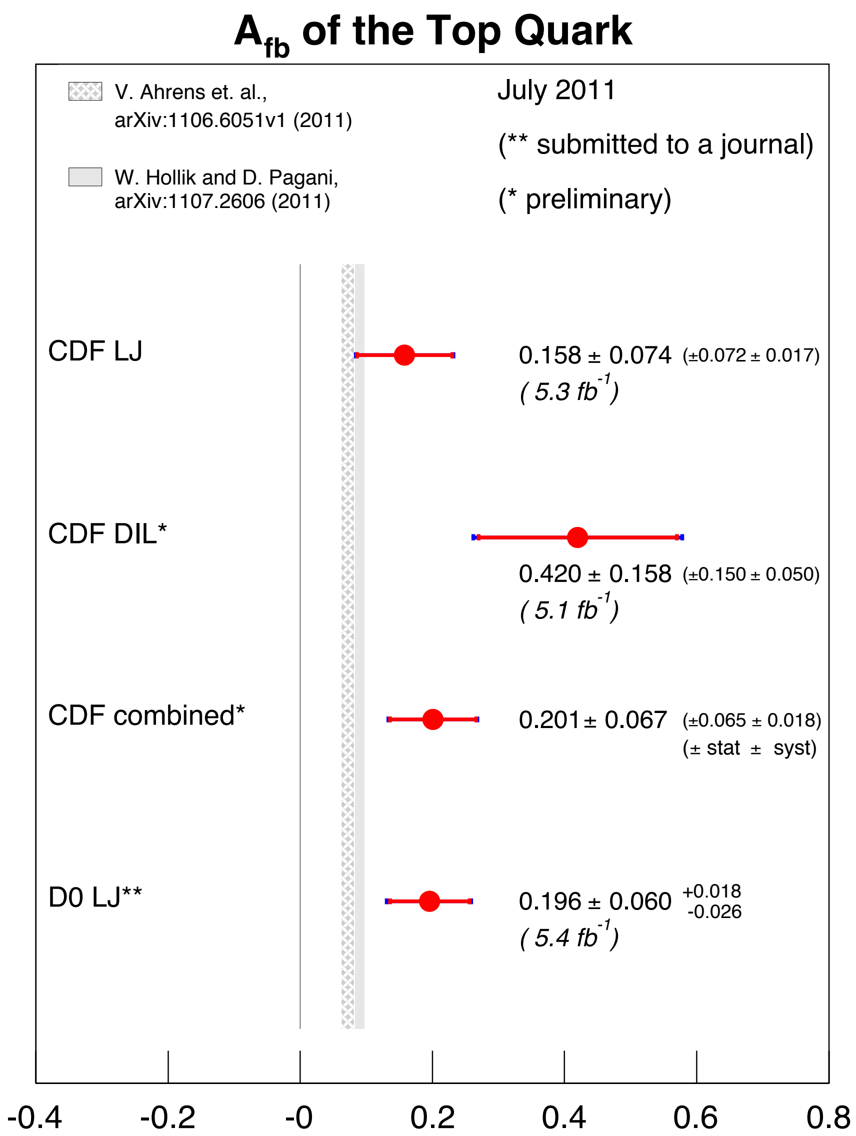
- First: Look for Standard Model explanations

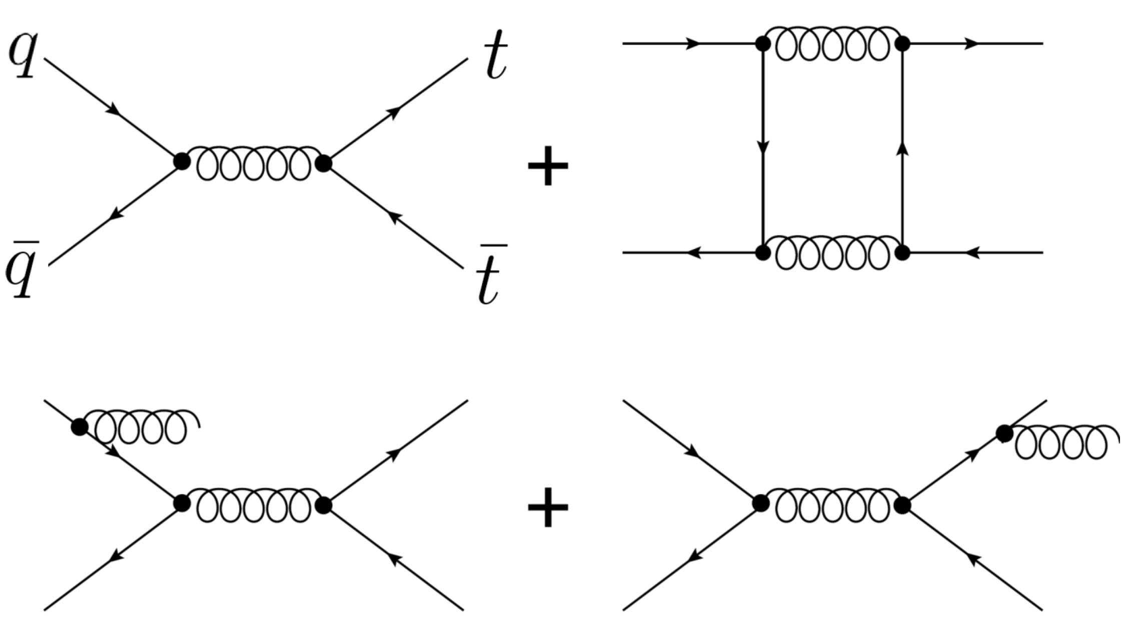




Example: top-antitop asymmetry at Tevatron

- First: Look for Standard Model explanations

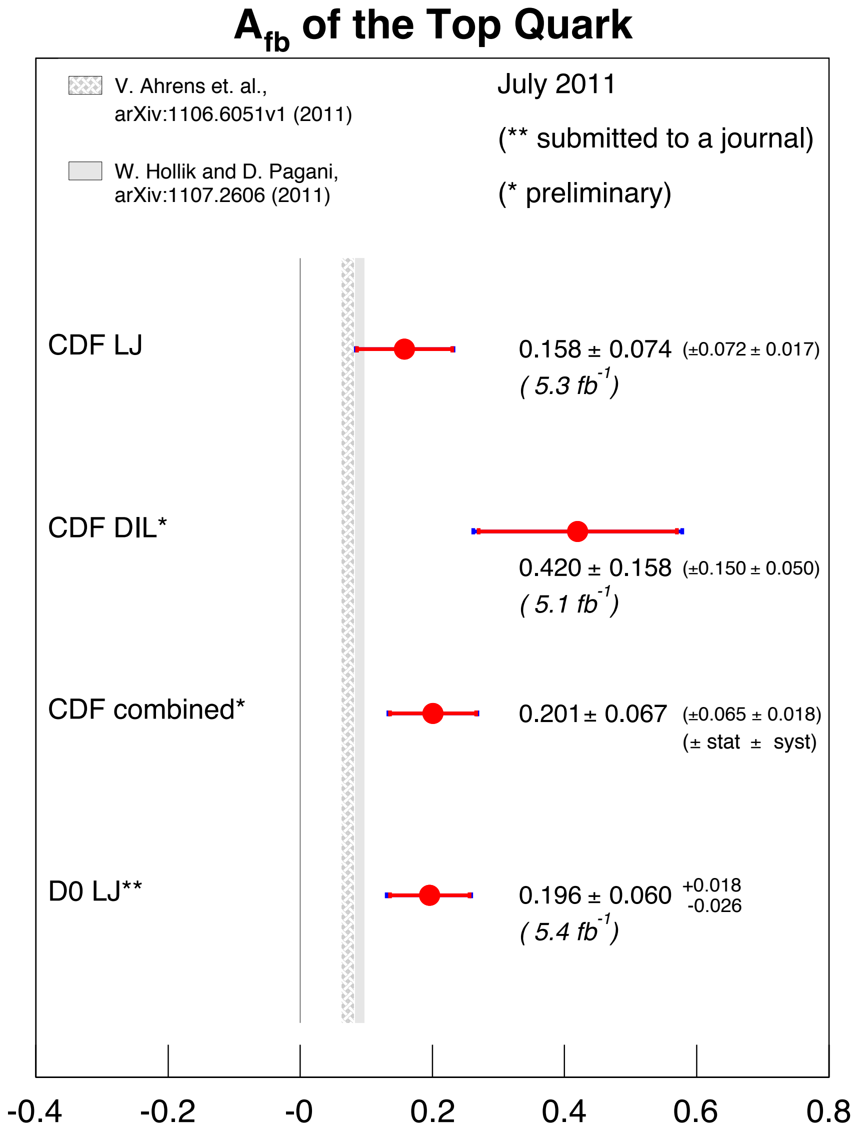


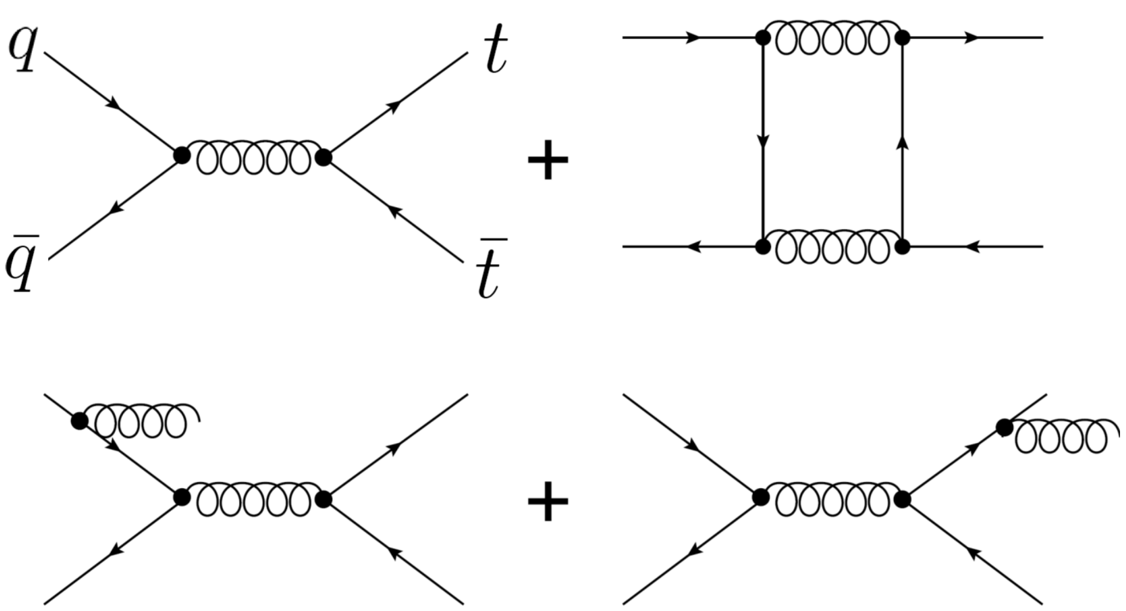


➡ Need NLO and EW contributions

Example: top-antitop asymmetry at Tevatron

- First: Look for Standard Model explanations





➡ Need NLO and EW contributions

➡ Reduces discrepancy with SM

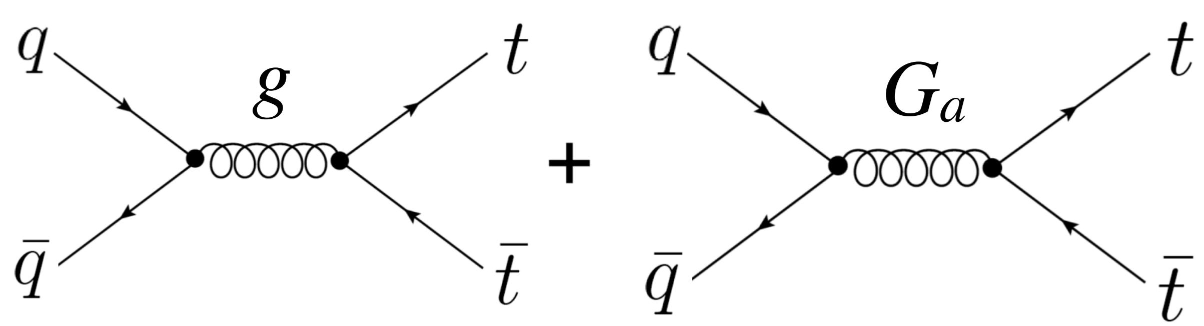
Example: top-antitop asymmetry at Tevatron

- Second: Look for possible New Physics contributions

Example: top-antitop asymmetry at Tevatron

- Second: Look for possible New Physics contributions

S-channel "gluon" with axial vector couplings and mass above the collider limit



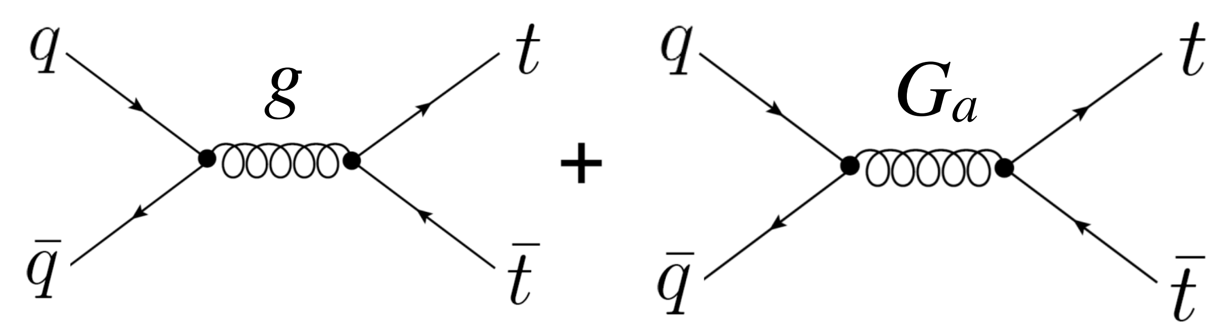
$M_G = 2\ TeV$

CDF collaboration, arXiv:1211.1003

Example: top-antitop asymmetry at Tevatron

- Second: Look for possible New Physics contributions

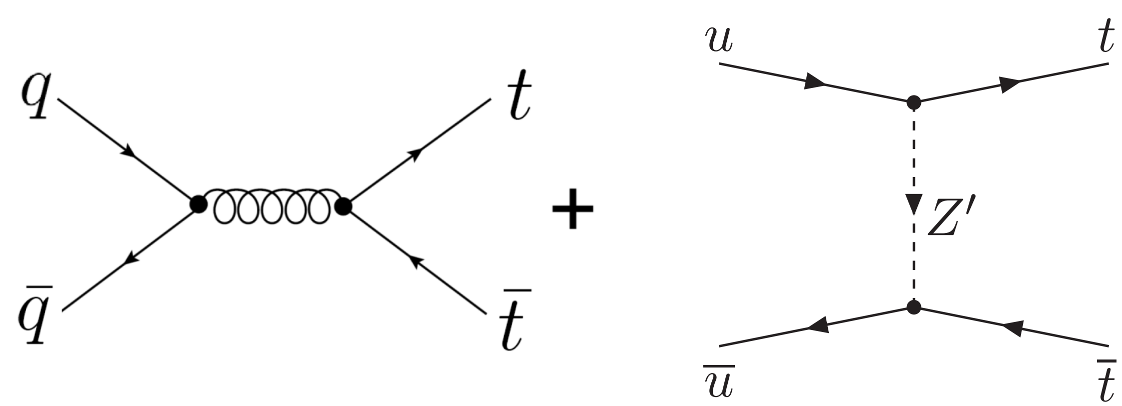
S-channel "gluon" with axial vector couplings and mass above the collider limit



$M_G = 2\ TeV$

CDF collaboration, arXiv:1211.1003

T-channel Z' with flavor-changing u-t coupling



$M_Z = 350\text{-}450\ \mathrm{GeV}$

Drobnak et al 1209.4354, Álvarez, Leskow 1209.4872

Example: top-antitop asymmetry at Tevatron

- Check if the model can explain the data!

Example: top-antitop asymmetry at Tevatron

- Check if the model can explain the data! How?

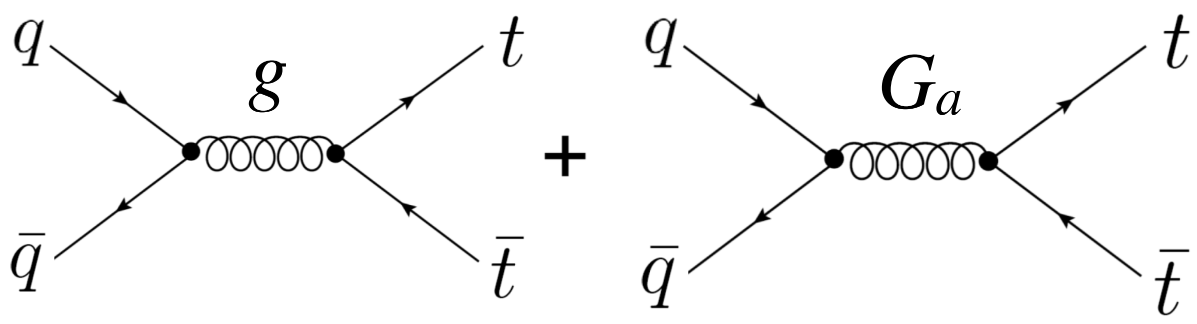
Example: top-antitop asymmetry at Tevatron

- Check if the model can explain the data! How? Monte Carlo!

Example: top-antitop asymmetry at Tevatron

- Check if the model can explain the data! How? Monte Carlo!

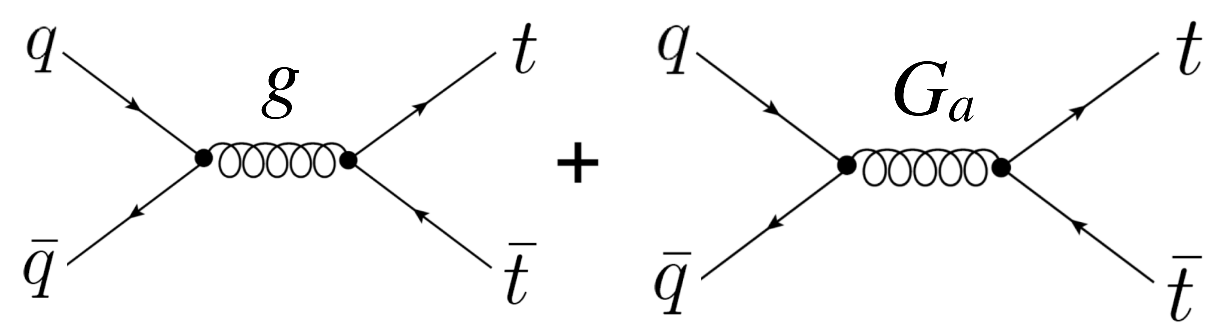
Axigluon

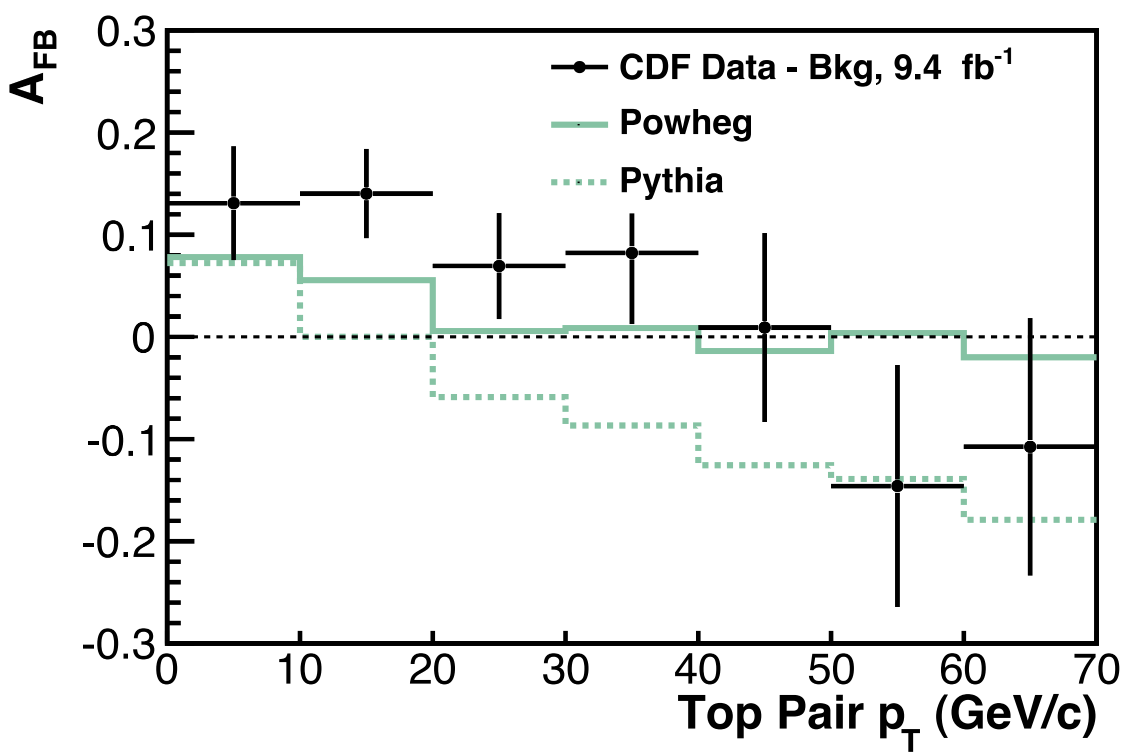


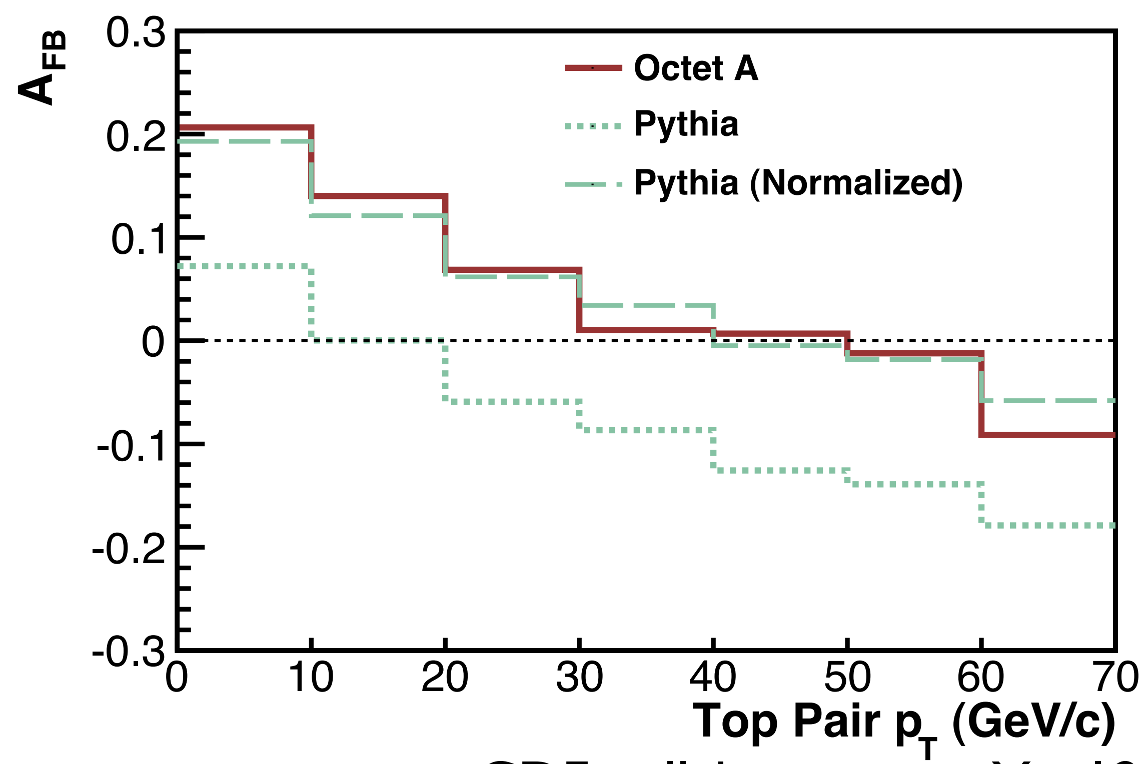
Example: top-antitop asymmetry at Tevatron

- Check if the model can explain the data! How? Monte Carlo!

Axigluon





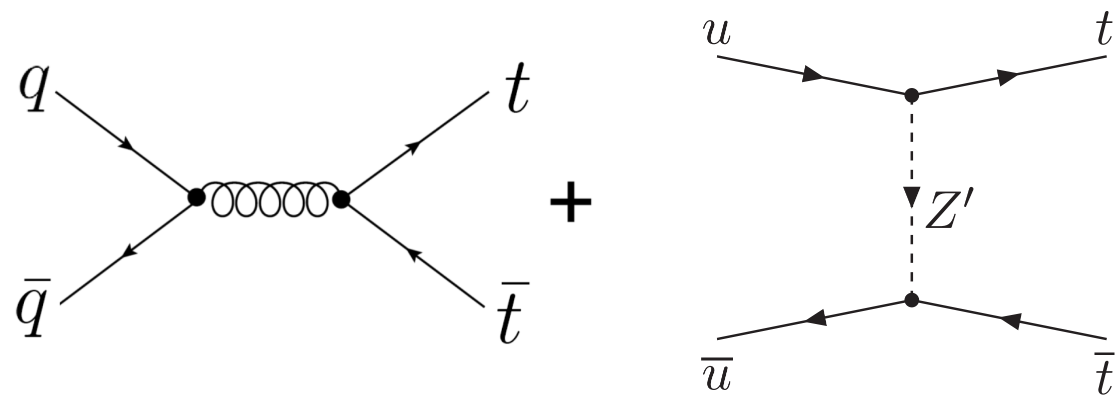


CDF collaboration, arXiv:1211.1003

Example: top-antitop asymmetry at Tevatron

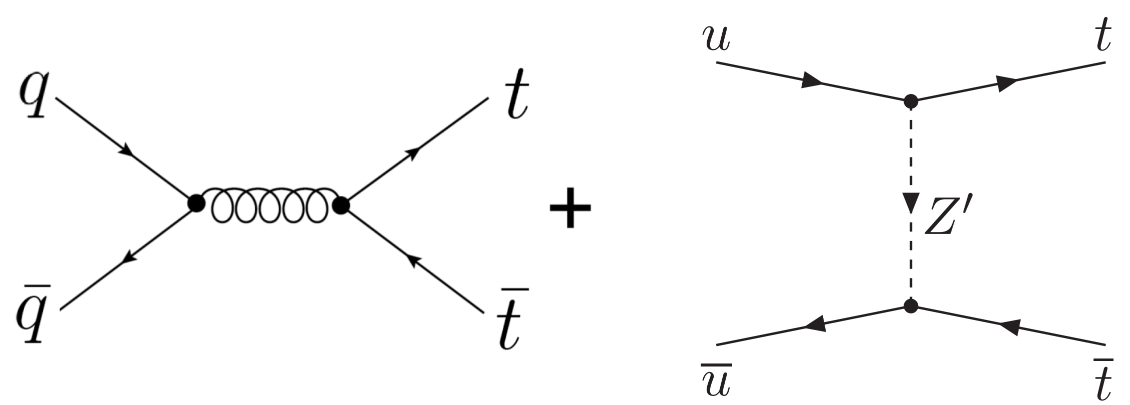
Example: top-antitop asymmetry at Tevatron

Flavor-changing Z'

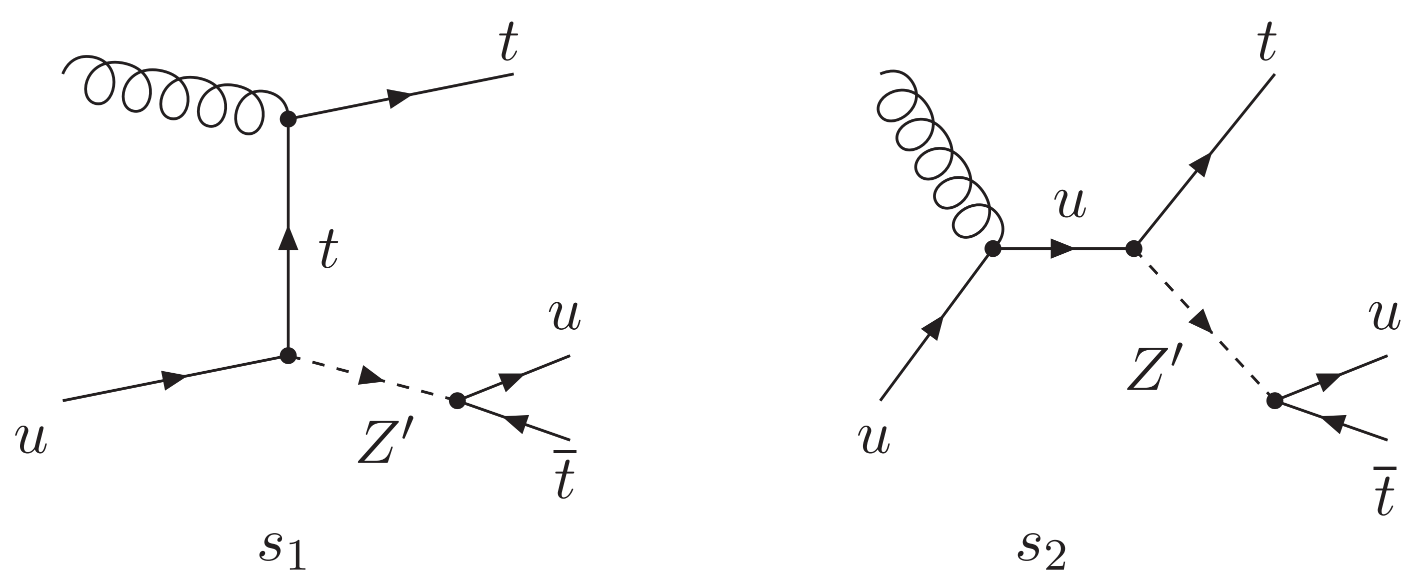


Example: top-antitop asymmetry at Tevatron

Flavor-changing Z’

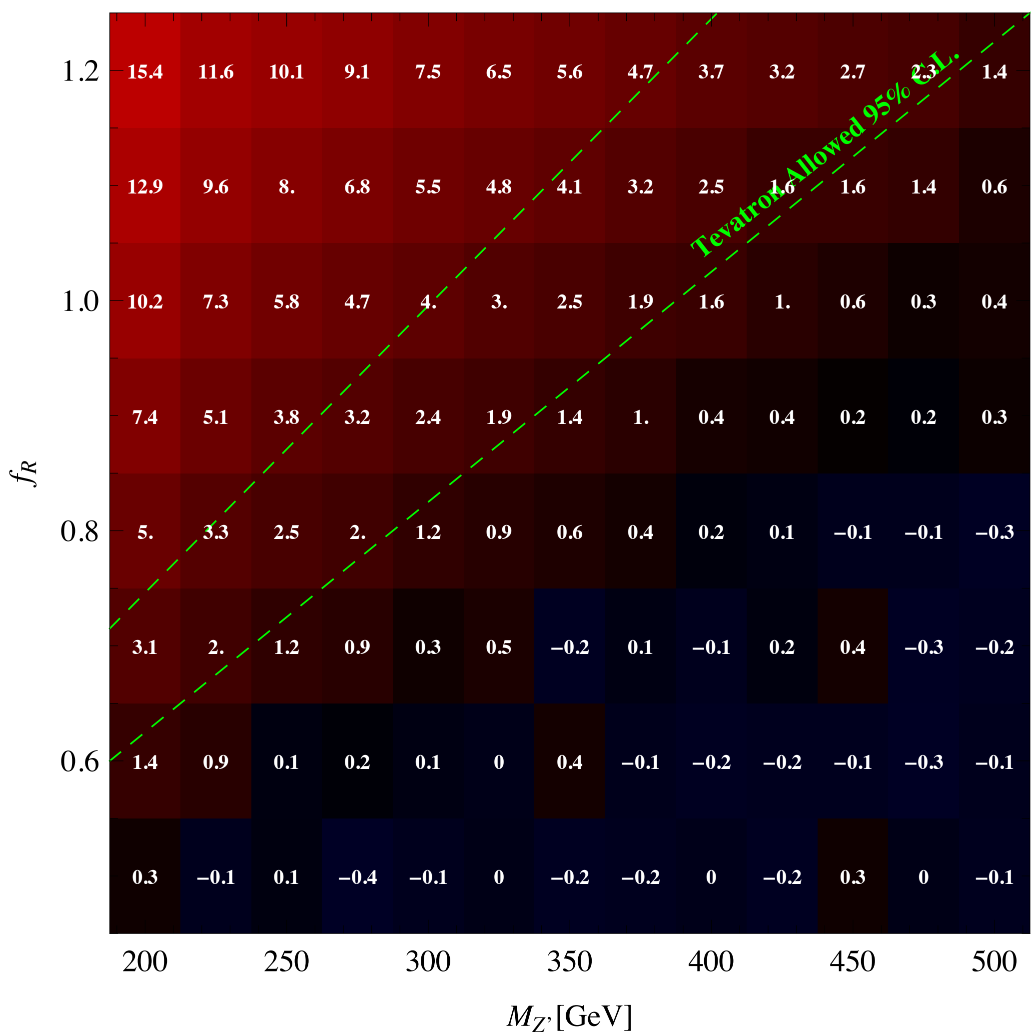


➡ Consequence:

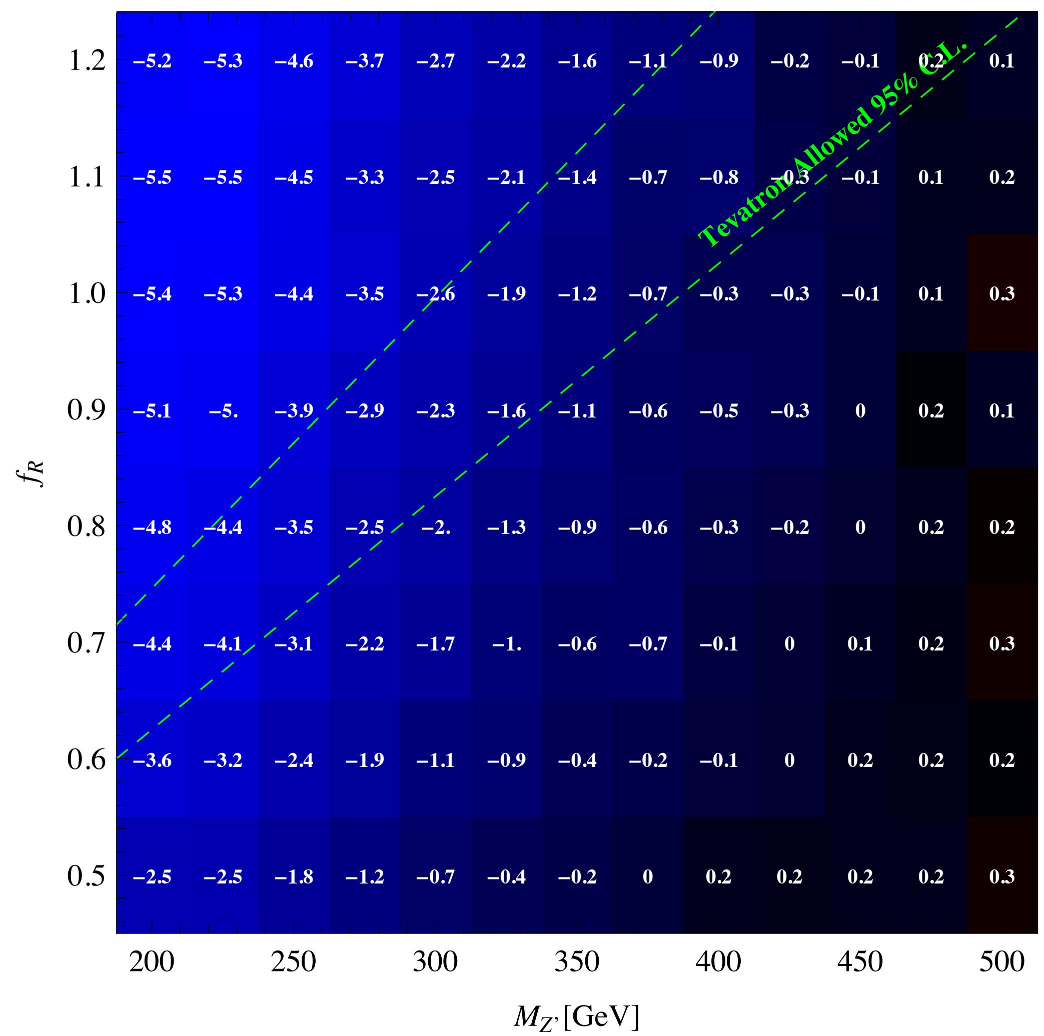


Example: top-antitop asymmetry at Tevatron

Flavor-changing Z’



t-channel charge asymmetry



s-channel charge asymmetry

Example: top-antitop asymmetry at Tevatron

Example: top-antitop asymmetry at Tevatron

- Now, think of ways to test the model at the LHC!

Example: top-antitop asymmetry at Tevatron

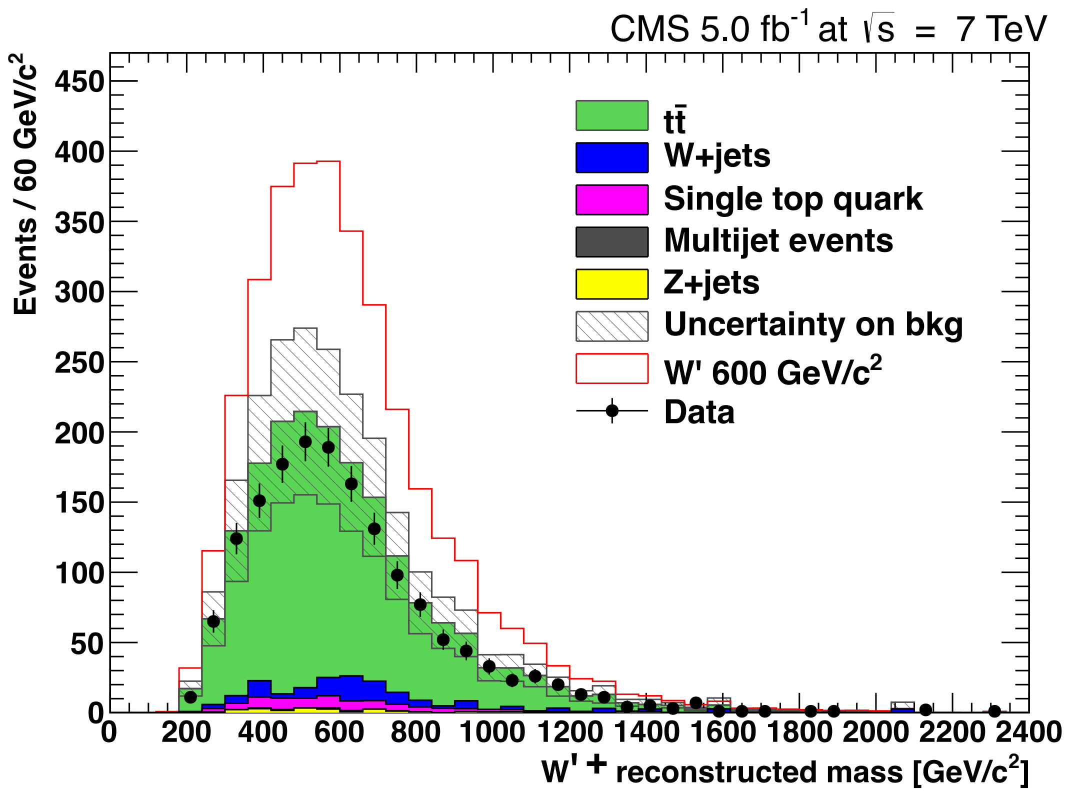
- Now, think of ways to test the model at the LHC!
  - ➡ Charge asymmetry A_C : $A_C = \frac{N(\Delta|y| > 0) - N(\Delta|y| < 0)}{N(\Delta|y| > 0) + N(\Delta|y| < 0)}.$

Example: top-antitop asymmetry at Tevatron

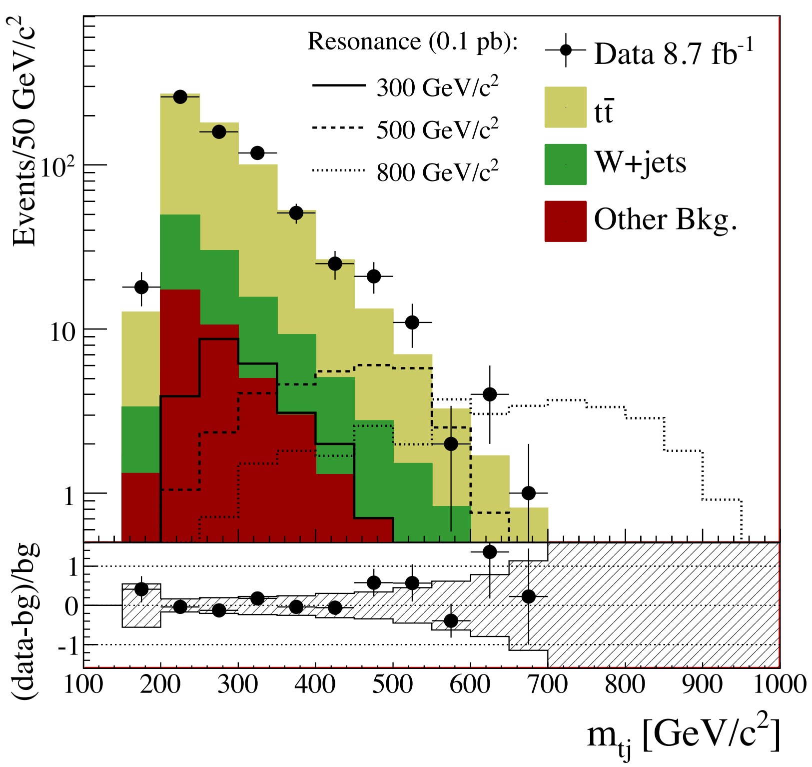
- Now, think of ways to test the model at the LHC!
  - ➡ Charge asymmetry A_C: $A_C = \frac{N(\Delta|y| > 0) - N(\Delta|y| < 0)}{N(\Delta|y| > 0) + N(\Delta|y| < 0)}.$
  - ➡ t+jet resonances

Example: top-antitop asymmetry at Tevatron

- Now, think of ways to test the model at the LHC!
  - ➡ Charge asymmetry A_C : $A_C = \dfrac{N(\Delta|y| > 0) - N(\Delta|y| < 0)}{N(\Delta|y| > 0) + N(\Delta|y| < 0)}.$
  - ➡ t+jet resonances



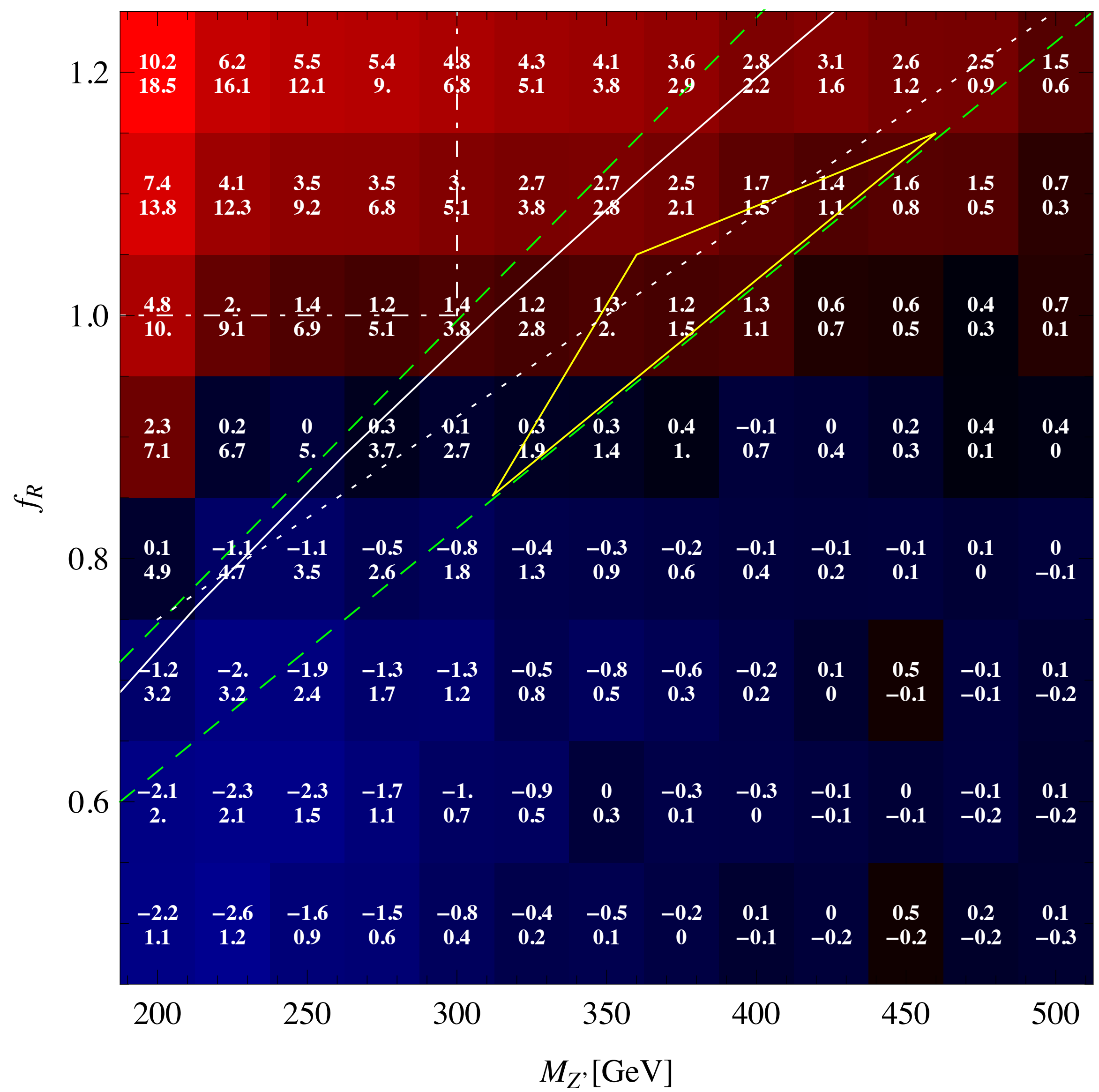
CMS collaboration, arXiv:1206.3921



CDF collaboration, arXiv:1203.3894

Example: top-antitop asymmetry at Tevatron

Flavor-changing Z'



Explaining/modeling excesses

Explaining/modeling excesses

1. An excess is discovered in data

Explaining/modeling excesses

1. An excess is discovered in data
2. Exhaust SM explanations for the excess

Explaining/modeling excesses

1. An excess is discovered in data
2. Exhaust SM explanations for the excess
3. Think of possible new physics explanations
   - ➡ Within or outside of conventional/high scale models

Explaining/modeling excesses

1. An excess is discovered in data
2. Exhaust SM explanations for the excess
3. Think of possible new physics explanations
   - ➡ Within or outside of conventional/high scale models
4. Find range of model parameters that can explain excess
   - ➡ Typically, using Monte Carlo simulations

Explaining/modeling excesses

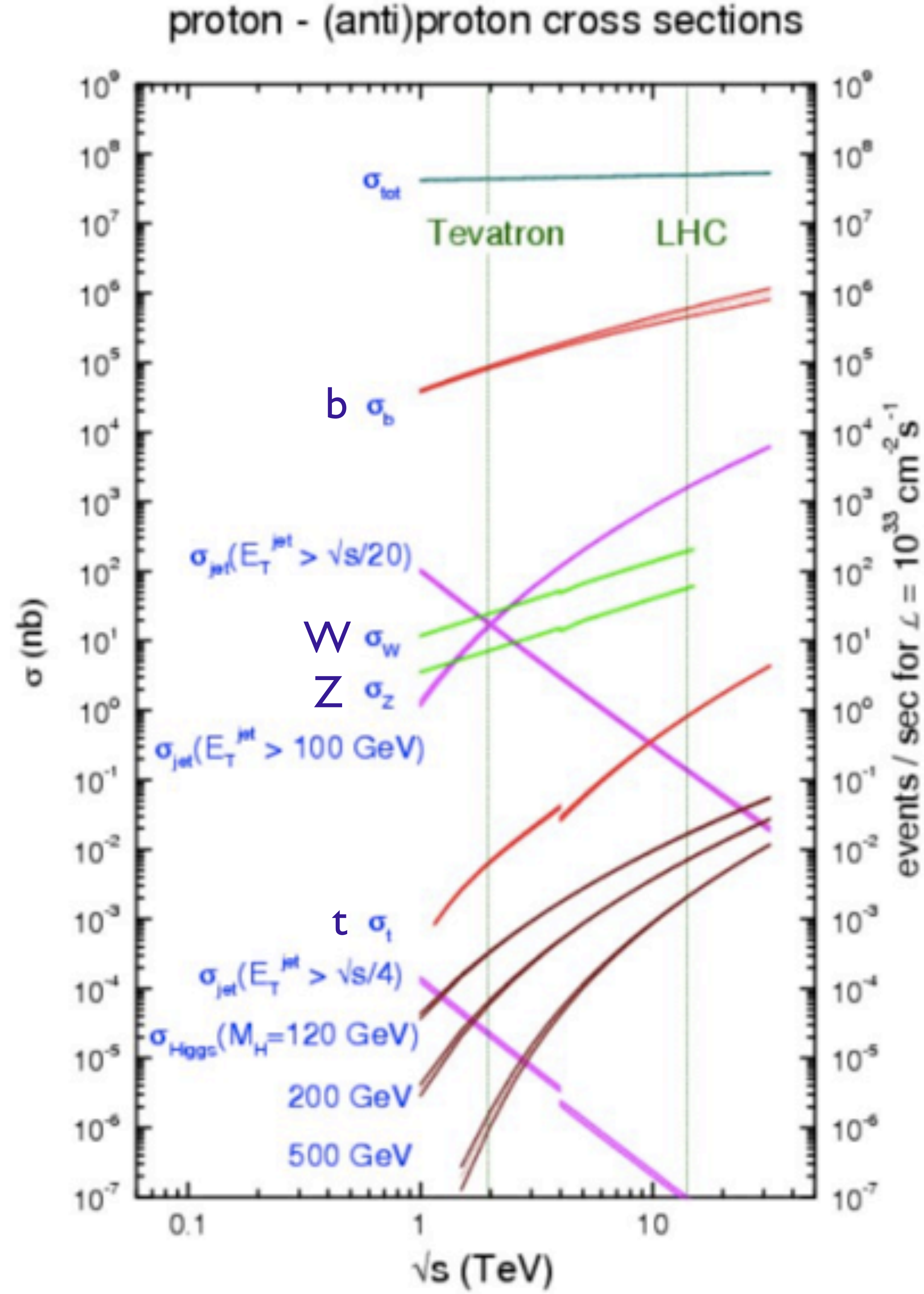
1. An excess is discovered in data
2. Exhaust SM explanations for the excess
3. Think of possible new physics explanations
   - ➡ Within or outside of conventional/high scale models
4. Find range of model parameters that can explain excess
   - ➡ Typically, using Monte Carlo simulations
5. Find other observables (collider as well as flavor/EWP/cosmology) where the explanation can be verified/falsified
   - ➡ Note that indirect constraints (flavor/EWP/cosmology) typically modified by additional particles in the spectrum

Processes at Hadron Colliders

First: Understand our processes!

Cross sections at a collider depend on:

- Coupling strength
- Coupling to what?
  (light quarks, gluons, heavy quarks, EW gauge bosons?)
- Mass
- Single production/pair production

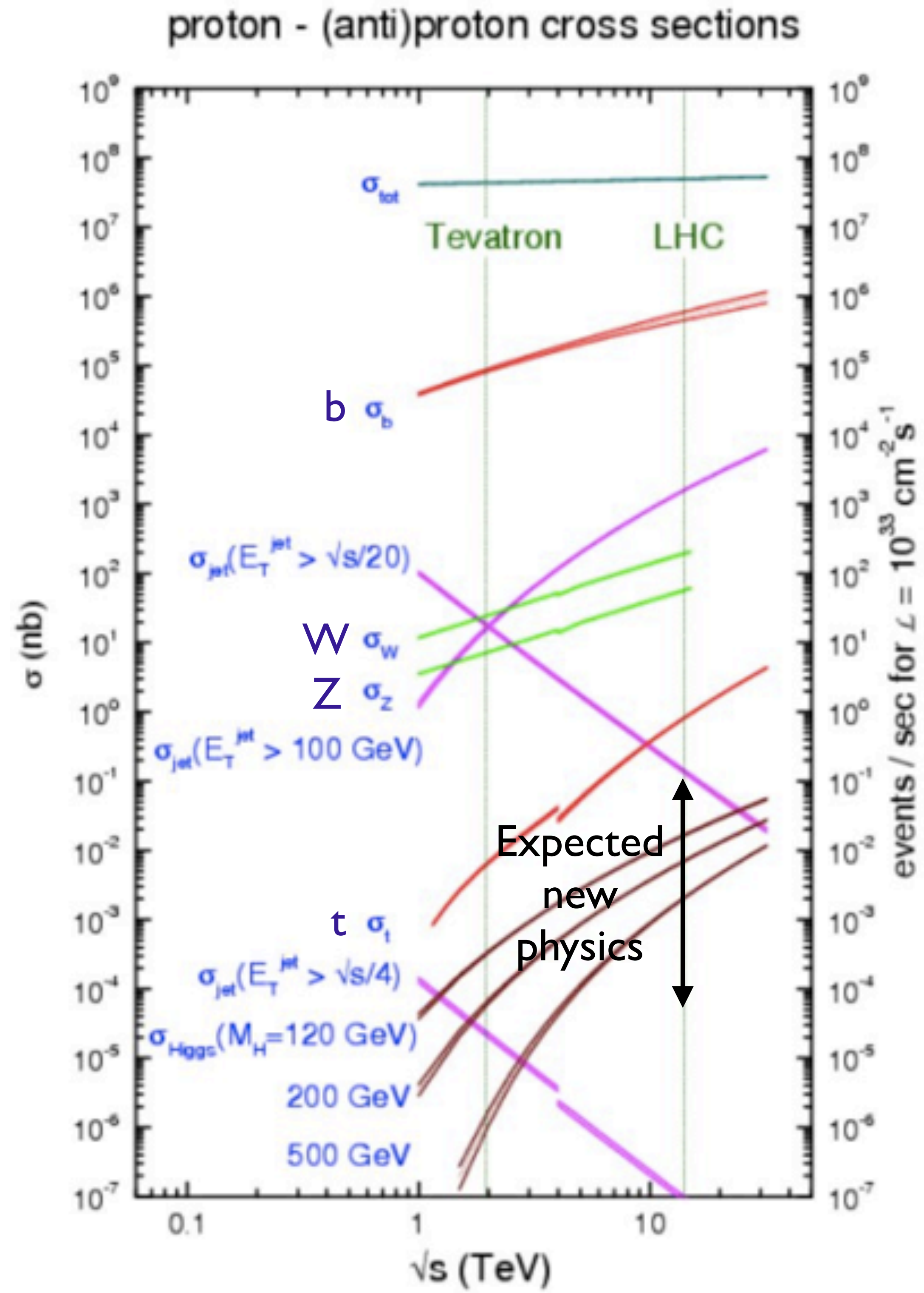


Processes at Hadron Colliders

First: Understand our processes!

Cross sections at a collider depend on:

- Coupling strength
- Coupling to what?
  (light quarks, gluons, heavy quarks, EW gauge bosons?)
- Mass
- Single production/pair production



Master formula

Master formula

$$\hat{\sigma}_{ab\to X}(\hat{s},\ldots)$$

Parton level
cross section

- Parton level cross section from matrix element

Master formula

$$\hat{\sigma}_{ab\to X}(\hat{s},\ldots)\, f_a(x_1) f_b(x_2)$$

Parton level cross section | Parton density functions

- Parton level cross section from matrix element
- Parton density (or distribution) functions:
  Process independent, determined by particle type

Master formula

$$\int \hat{\sigma}_{ab\to X}(\hat{s},\ldots)\, f_a(x_1) f_b(x_2)\, dx_1 dx_2 d\Phi_{FS}$$

Parton level cross section | Parton density functions | Phase space integral

- Parton level cross section from matrix element
- Parton density (or distribution) functions: Process independent, determined by particle type

Master formula

$$\int \hat{\sigma}_{ab\to X}(\hat{s},\ldots)\, f_a(x_1) f_b(x_2)\, dx_1 dx_2 d\Phi_{FS}$$

Parton level cross section — Parton density functions — Phase space integral

- Parton level cross section from matrix element
- Parton density (or distribution) functions: Process independent, determined by particle type
- $\hat{s} = x_1 x_2 s$ (s = collision energy of the collider)

Master formula

$$\int \hat{\sigma}_{ab\to X}(\hat{s},\ldots) f_a(x_1) f_b(x_2)\, dx_1 dx_2 d\Phi_{FS}$$

Parton level cross section | Parton density functions | Phase space integral

- Parton level cross section from matrix element
- Parton density (or distribution) functions:
  Process independent, determined by particle type
- $\hat{s} = x_1 x_2 s$ (s = collision energy of the collider)
- Difference between colliders given by parton luminocities

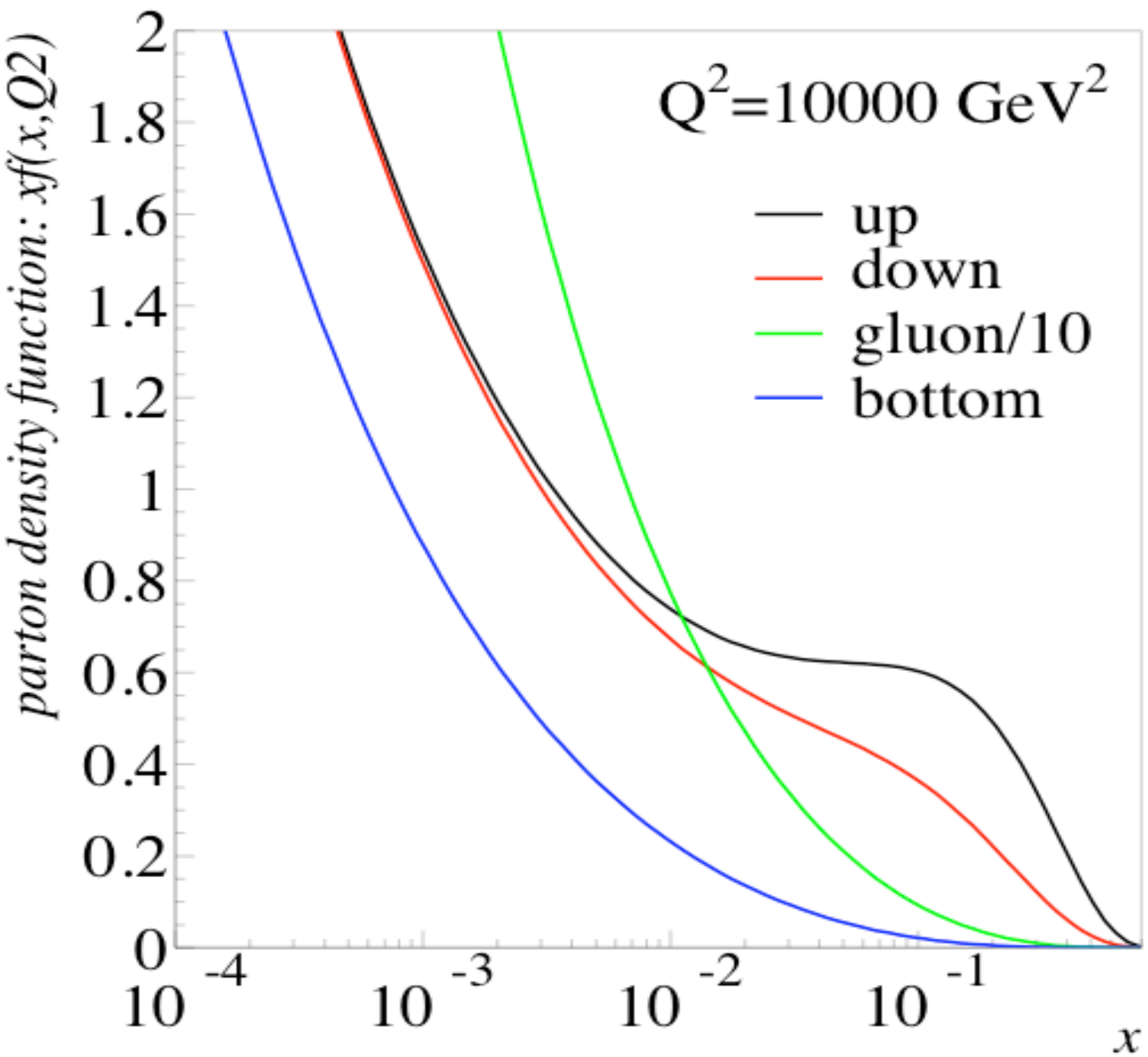
Tevatron vs. the LHC

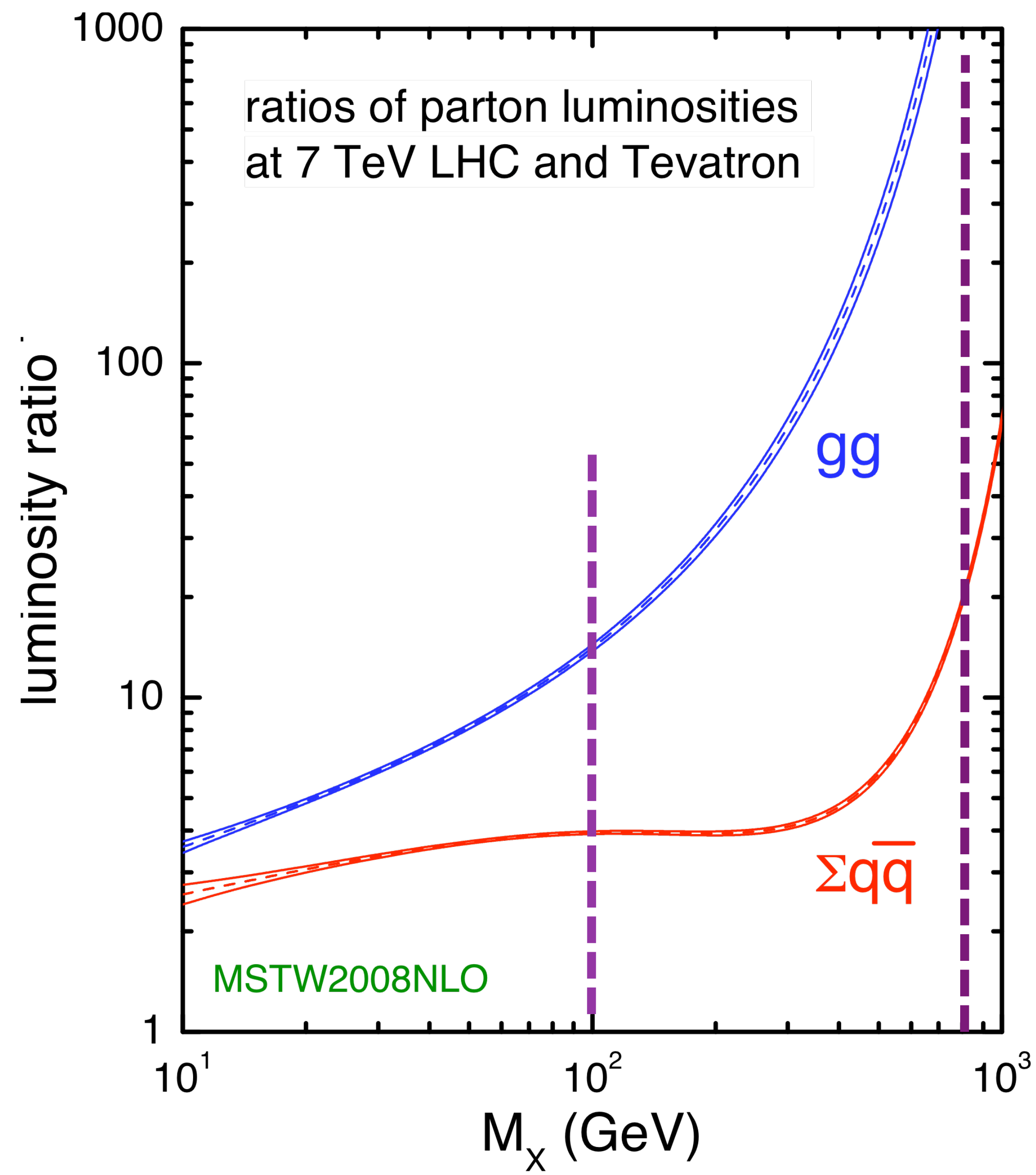
- Tevatron: 2 TeV proton-antiproton collider
  - ➡ Most important: q-$\bar{q}$ annihilation (85% of $t\bar{t}$)
- LHC: 8-14 TeV proton-proton collider
  - ➡ Most important: g-g annihilation (90% of $t\bar{t}$)

Tevatron vs. the LHC

- Tevatron: 2 TeV proton-antiproton collider
  - ➡ Most important: q-$\bar{q}$ annihilation (85% of $t\bar{t}$)
- LHC: 8-14 TeV proton-proton collider
  - ➡ Most important: g-g annihilation (90% of $t\bar{t}$)

Parton densities

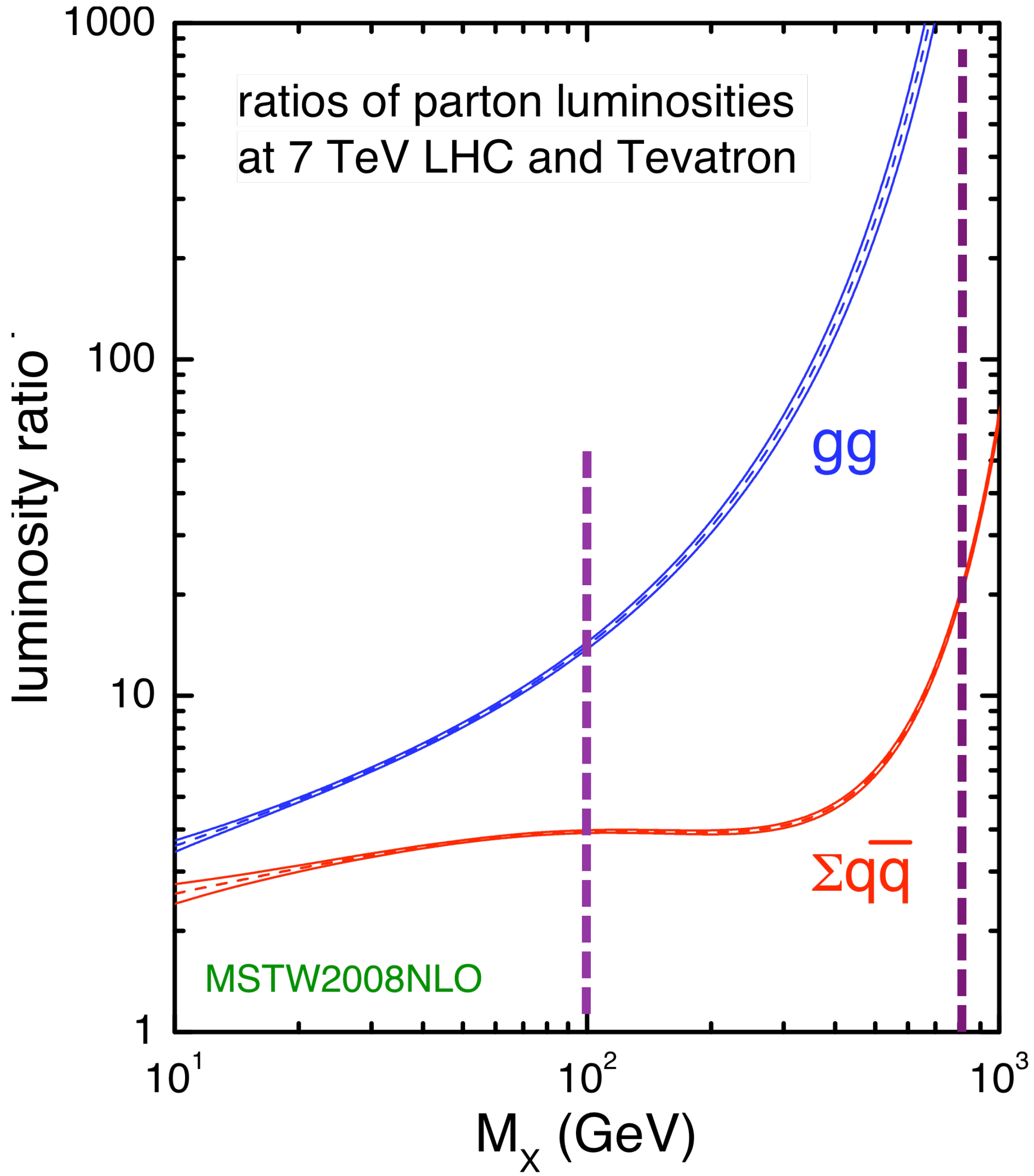


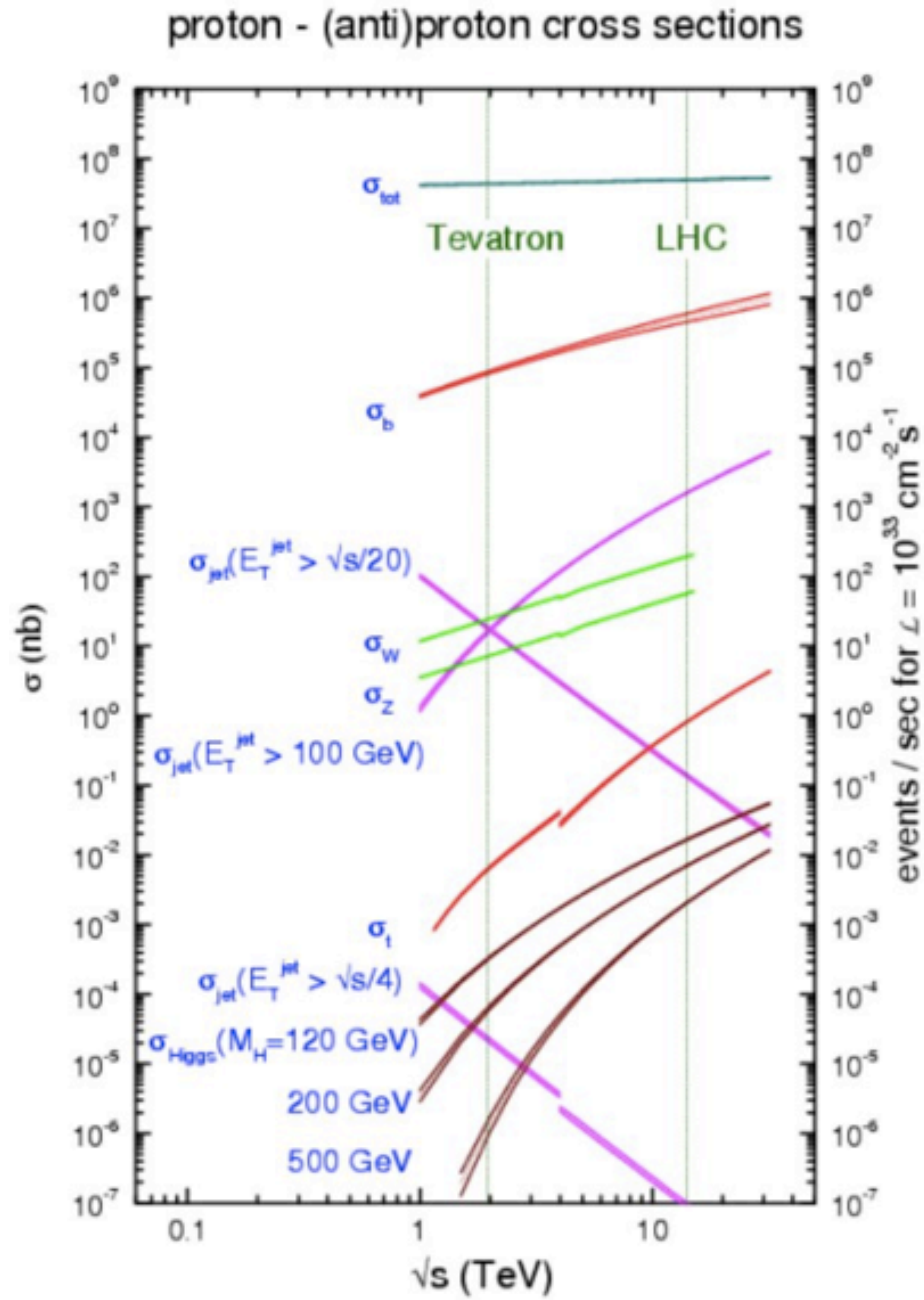


At small x (small $\hat{s}$), gluon domination.
At large x valence quarks

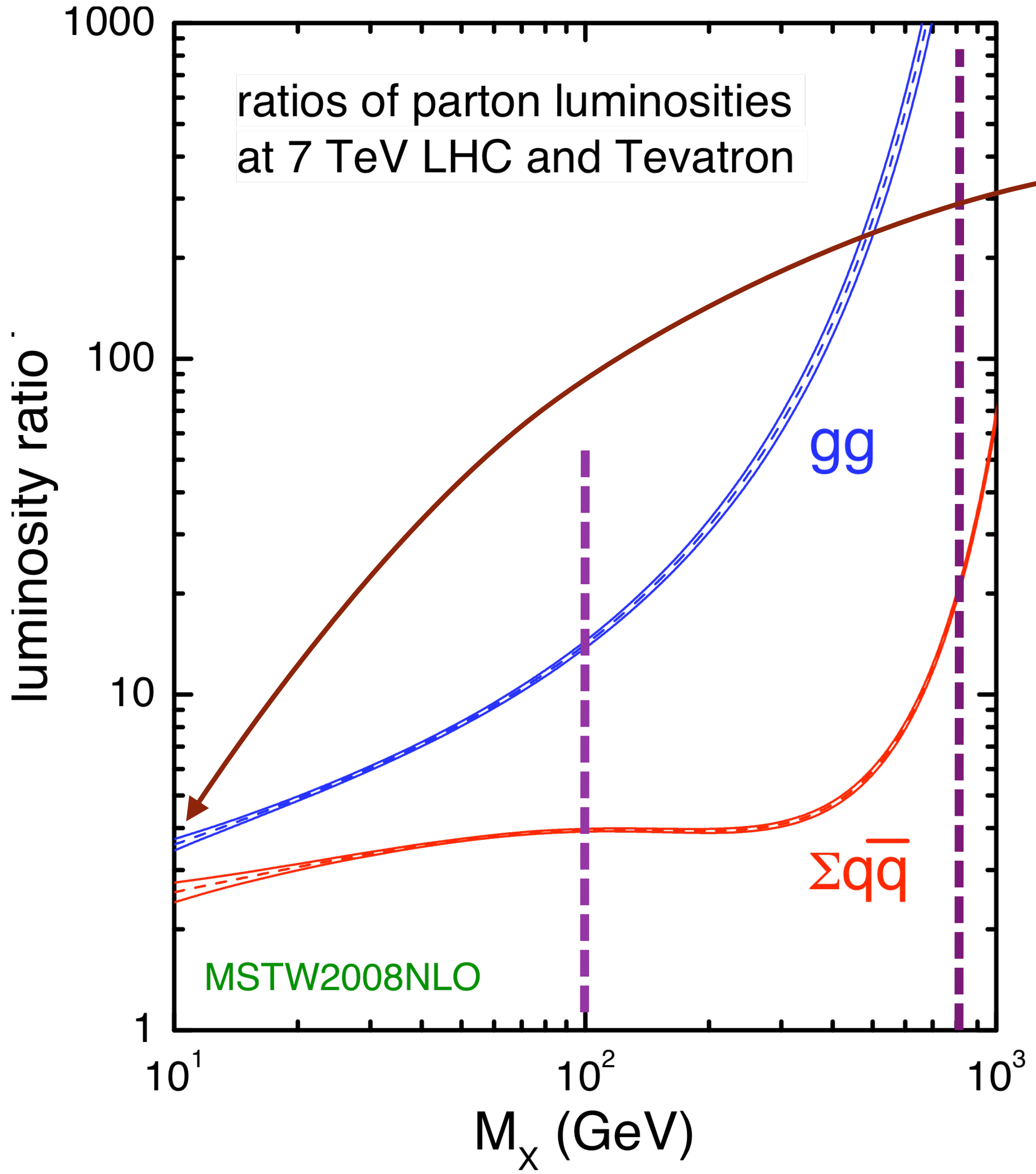
LHC formidable at large mass –
For low mass, Tevatron backgrounds smaller

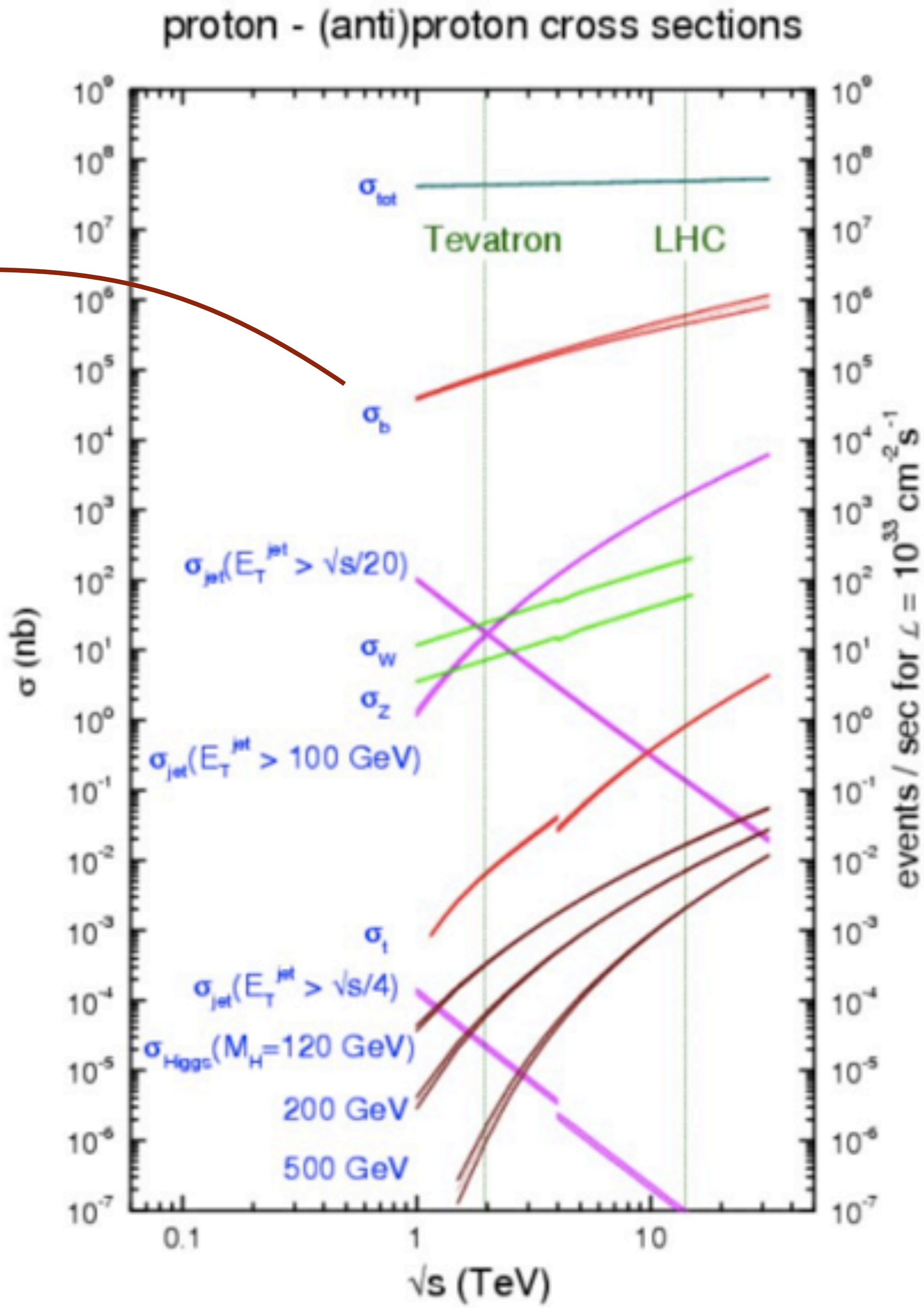
Back to the processes



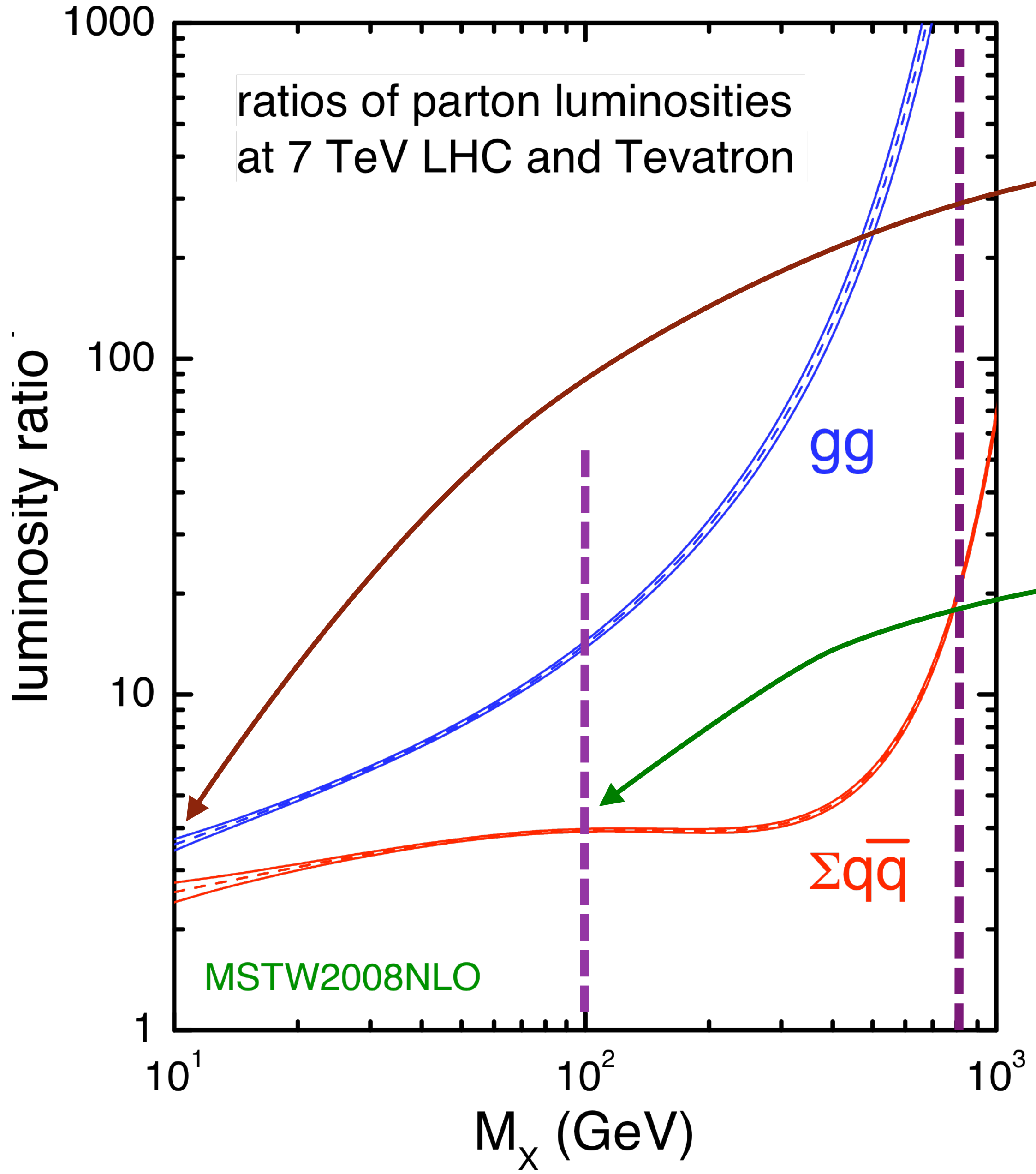


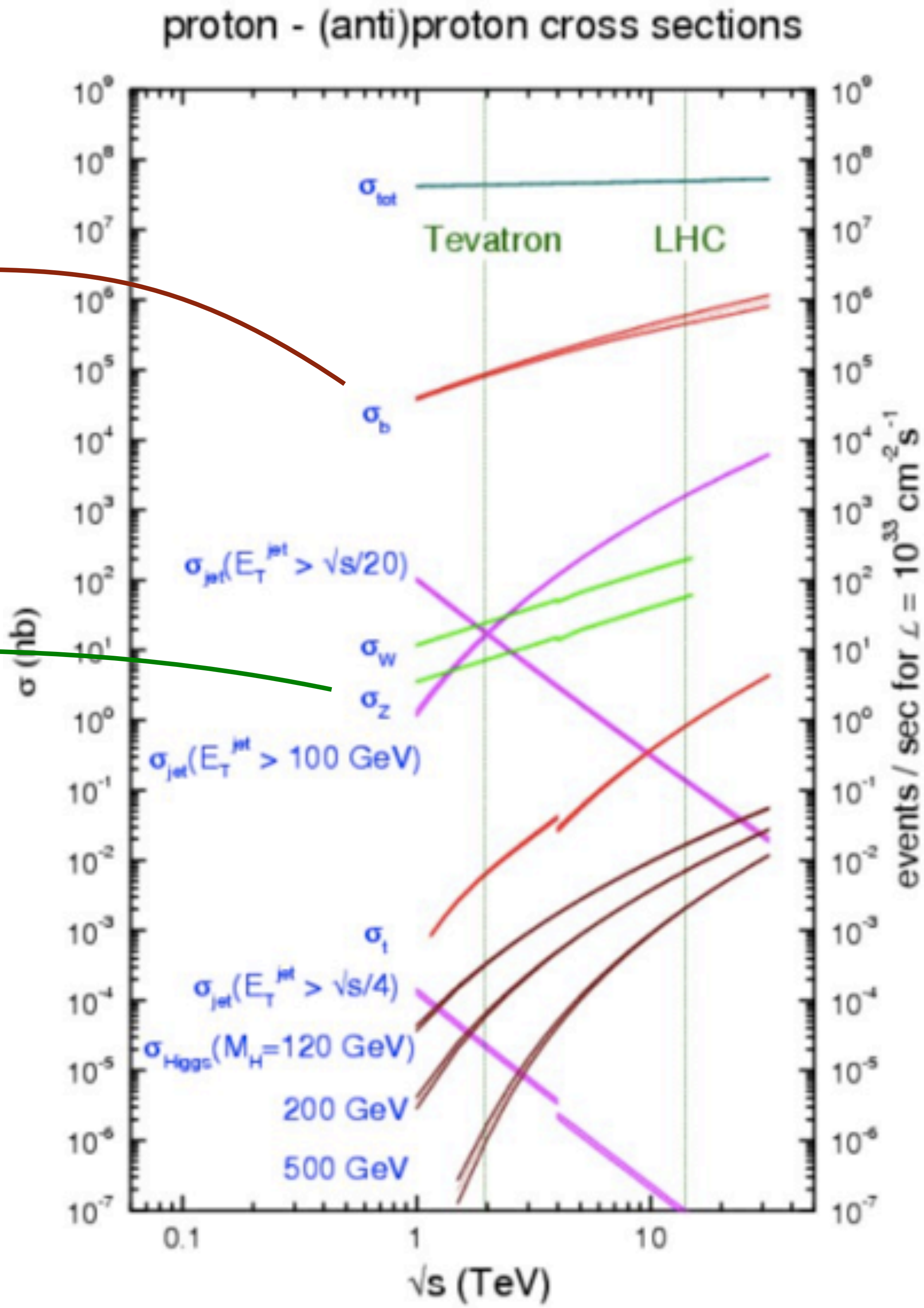
Back to the processes



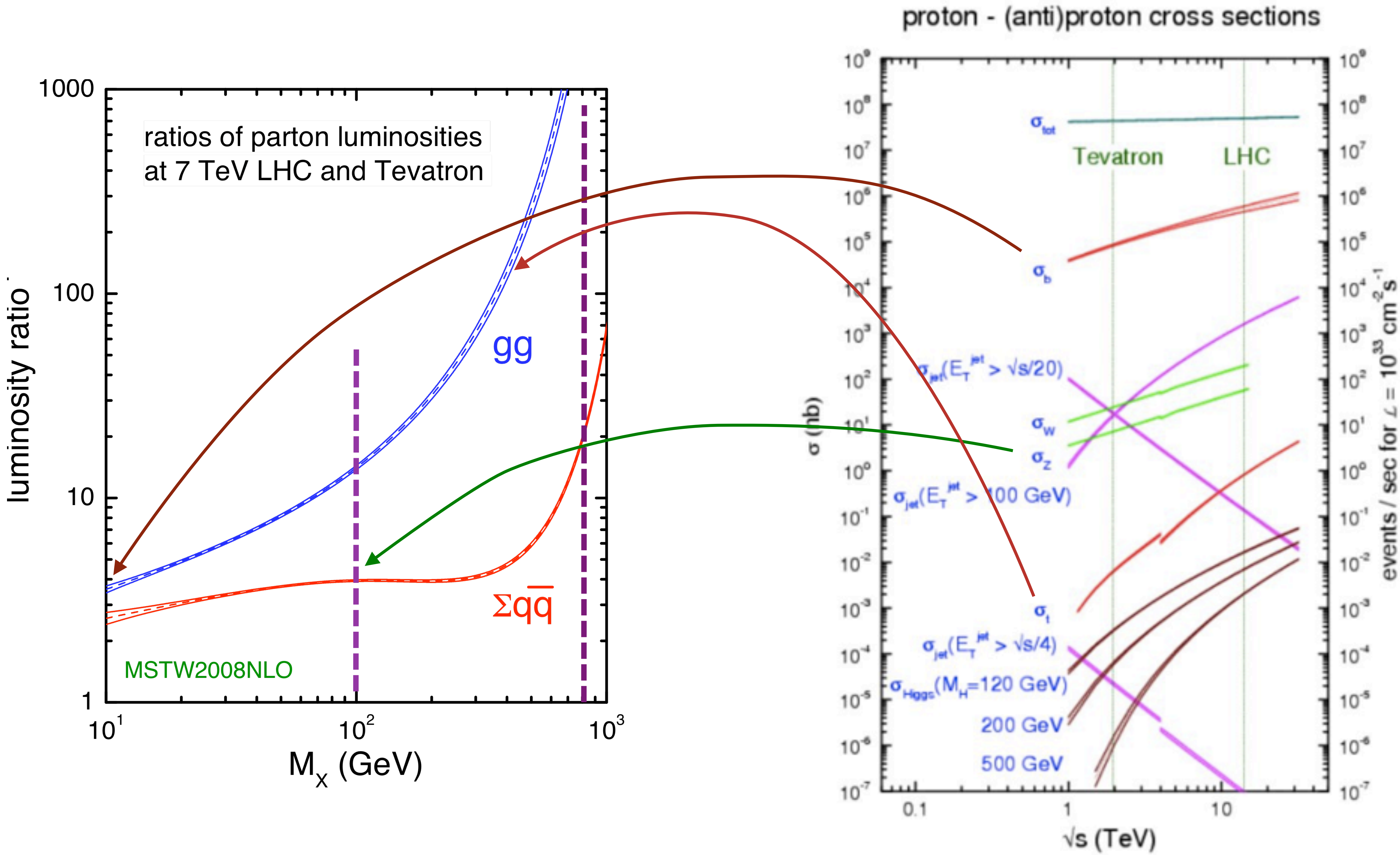


Back to the processes





Back to the processes



Monte Carlo Integration and Generation

Monte Carlo Integration and Generation

Calculations of cross section or decay widths involve integrations over high-dimension phase space of very peaked functions:

Monte Carlo Integration and Generation

Calculations of cross section or decay widths involve integrations over high-dimension phase space of very peaked functions:

$$\sigma = \frac{1}{2s}\int |\mathcal{M}|^2 d\Phi(n)$$

Monte Carlo Integration and Generation

Calculations of cross section or decay widths involve integrations over high-dimension phase space of very peaked functions:

$$\sigma = \frac{1}{2s}\int |\mathcal{M}|^2 d\Phi(n)$$

$$Dim[\Phi(n)] \sim 3n$$

Monte Carlo Integration and Generation

Calculations of cross section or decay widths involve integrations over high-dimension phase space of very peaked functions:

$$\sigma = \frac{1}{2s}\int |\mathcal{M}|^2 d\Phi(n)$$

$$Dim[\Phi(n)] \sim 3n$$

General and flexible method is needed

Integrals as averages

$$I = \int_{x_1}^{x_2} f(x)dx \quad \Rightarrow \quad I_N = (x_2 - x_1)\frac{1}{N}\sum_{i=1}^{N} f(x)$$

$$V = (x_2 - x_1)\int_{x_1}^{x_2} [f(x)]^2 dx - I^2 \quad \Rightarrow \quad V_N = (x_2 - x_1)^2 \frac{1}{N}\sum_{i=1}^{N} [f(x)]^2 - I_N^2$$

Integrals as averages

$$I = \int_{x_1}^{x_2} f(x)dx \quad \Rightarrow \quad I_N = (x_2 - x_1)\frac{1}{N}\sum_{i=1}^{N} f(x)$$

$$V = (x_2 - x_1)\int_{x_1}^{x_2} [f(x)]^2 dx - I^2 \quad \Rightarrow \quad V_N = (x_2 - x_1)^2 \frac{1}{N}\sum_{i=1}^{N}[f(x)]^2 - I_N^2$$

$$I = I_N \pm \sqrt{V_N/N}$$

Integrals as averages

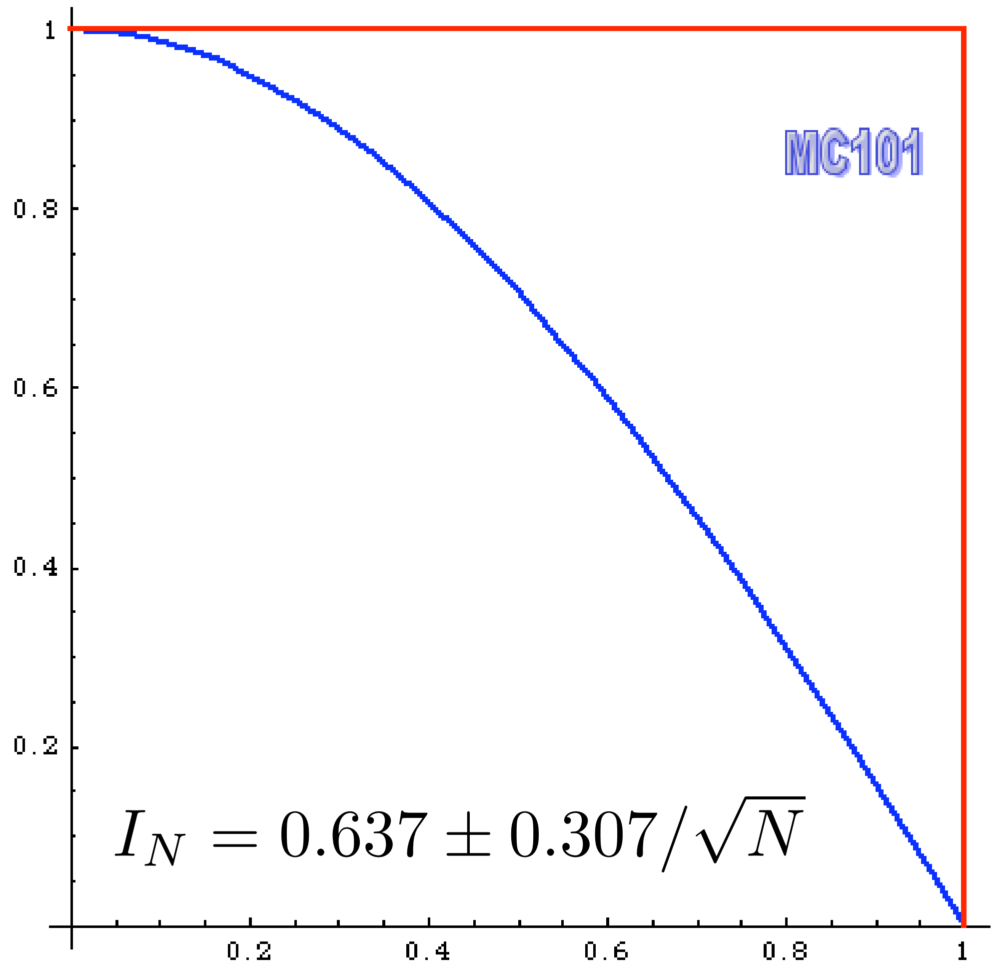
$$I = \int_{x_1}^{x_2} f(x)dx \quad \Rightarrow \quad I_N = (x_2 - x_1)\frac{1}{N}\sum_{i=1}^{N} f(x)$$

$$V = (x_2 - x_1)\int_{x_1}^{x_2} [f(x)]^2 dx - I^2 \quad \Rightarrow \quad V_N = (x_2 - x_1)^2 \frac{1}{N}\sum_{i=1}^{N}[f(x)]^2 - I_N^2$$

$$I = I_N \pm \sqrt{V_N/N}$$

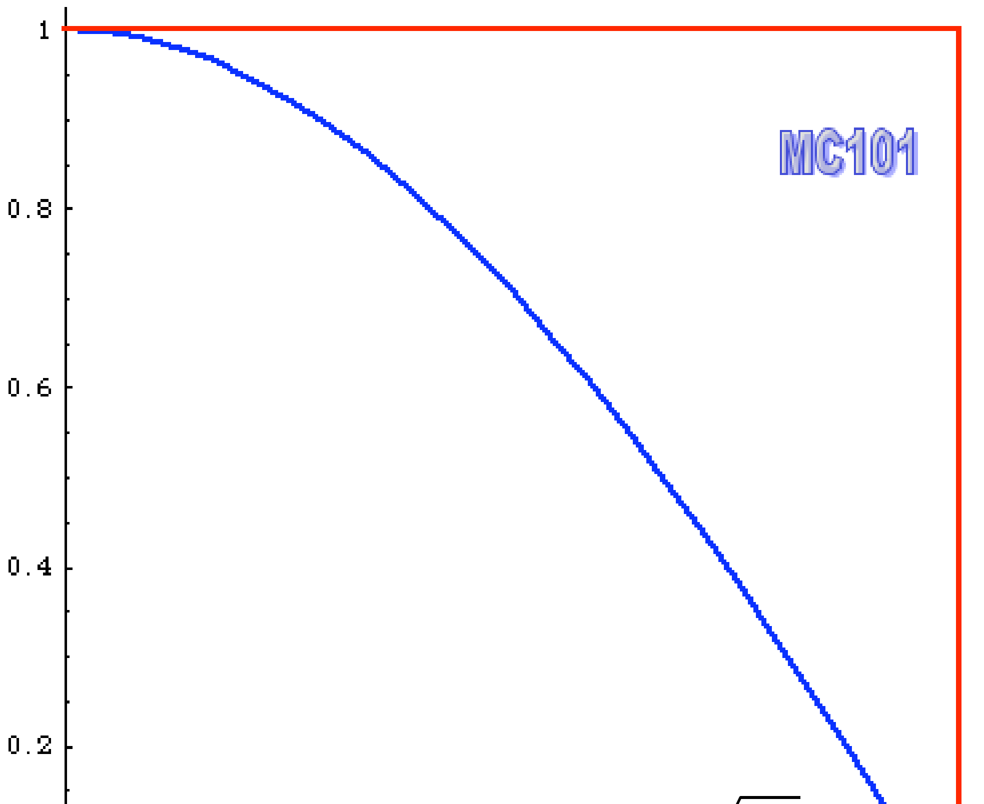
- ☞ Convergence is slow but it can be easily estimated
- ☞ Error does not depend on # of dimensions!
- ☞ Improvement by minimizing V_N.
- ☞ Optimal/Ideal case: $f(x)=C \Rightarrow V_N=0$

Importance Sampling



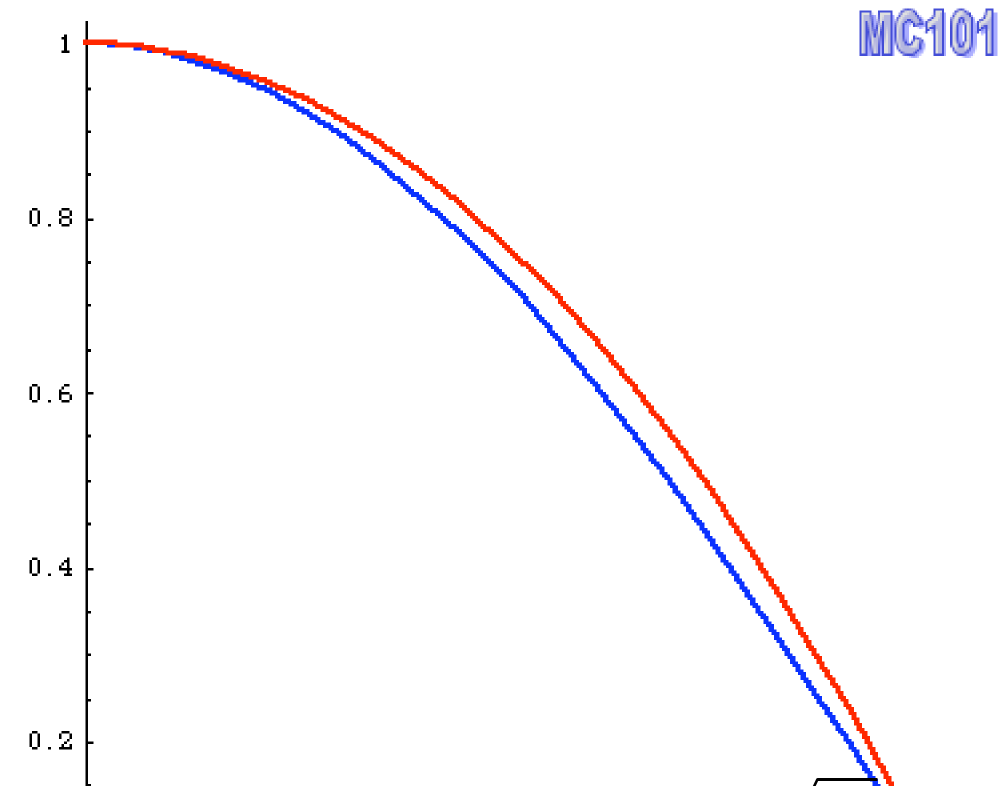
$$I = \int_0^1 dx \cos \frac{\pi}{2} x$$

Importance Sampling



$I_N = 0.637 \pm 0.307/\sqrt{N}$

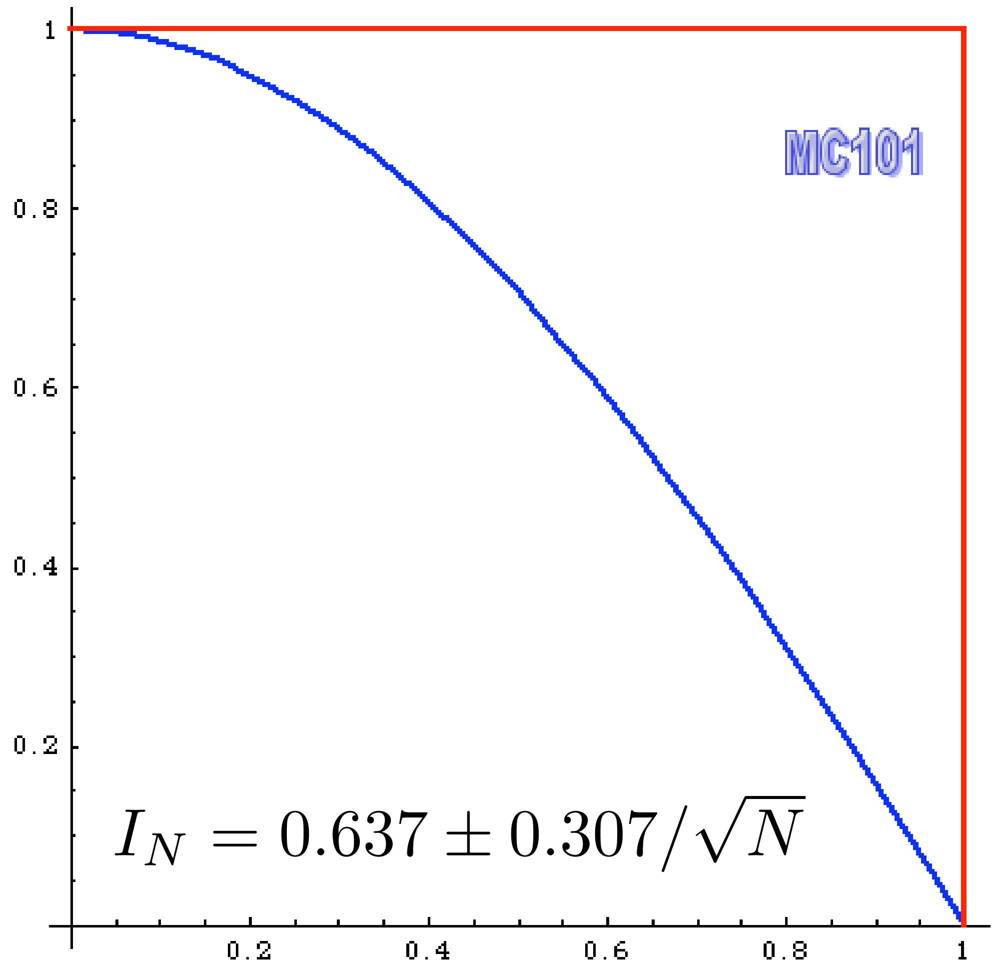
$$I = \int_0^1 dx \cos \frac{\pi}{2} x$$

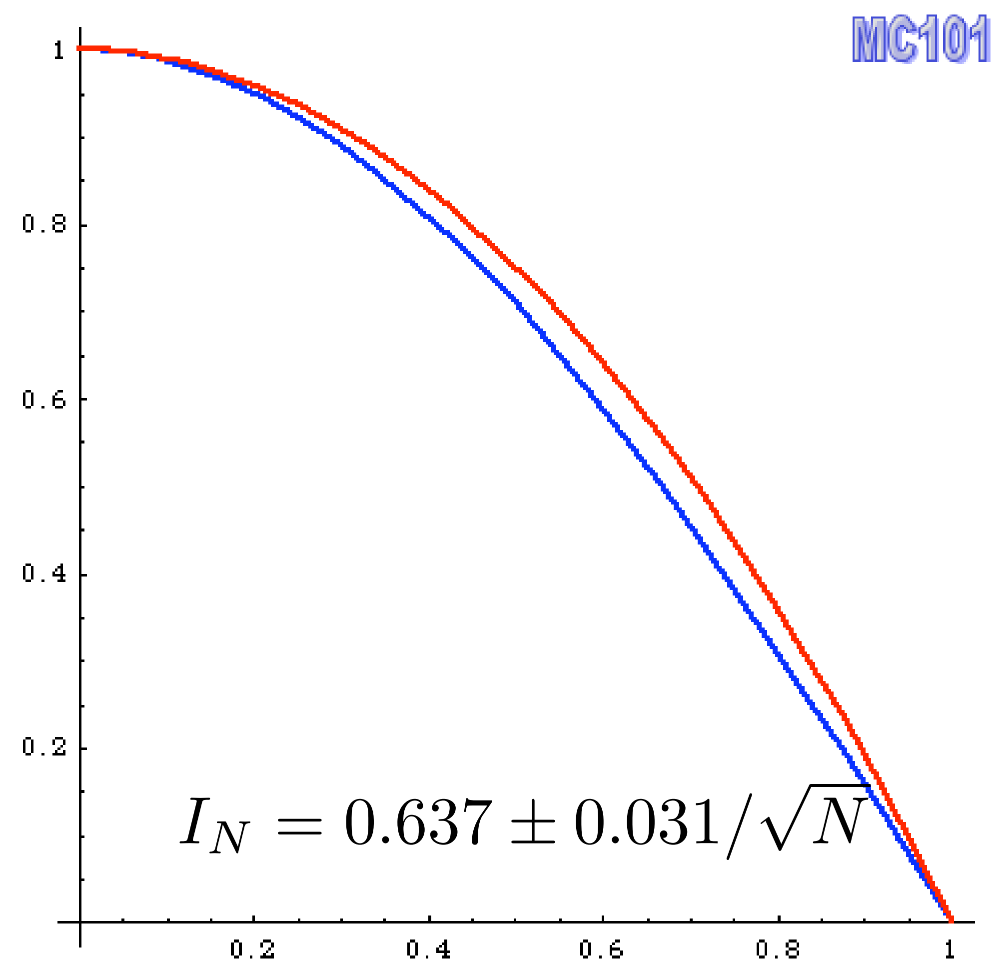


$I_N = 0.637 \pm 0.031/\sqrt{N}$

$$I = \int_0^1 dx (1 - x^2) \frac{\cos \frac{\pi}{2} x}{1 - x^2}$$

Importance Sampling



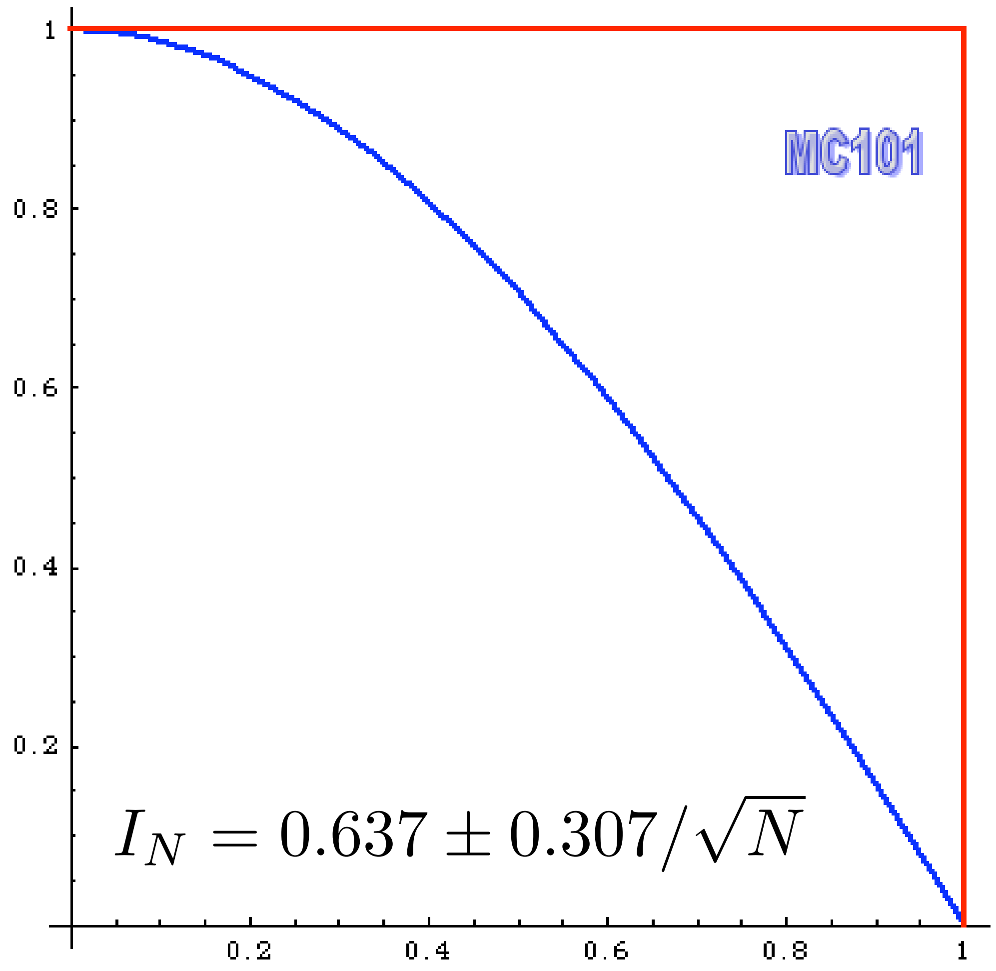


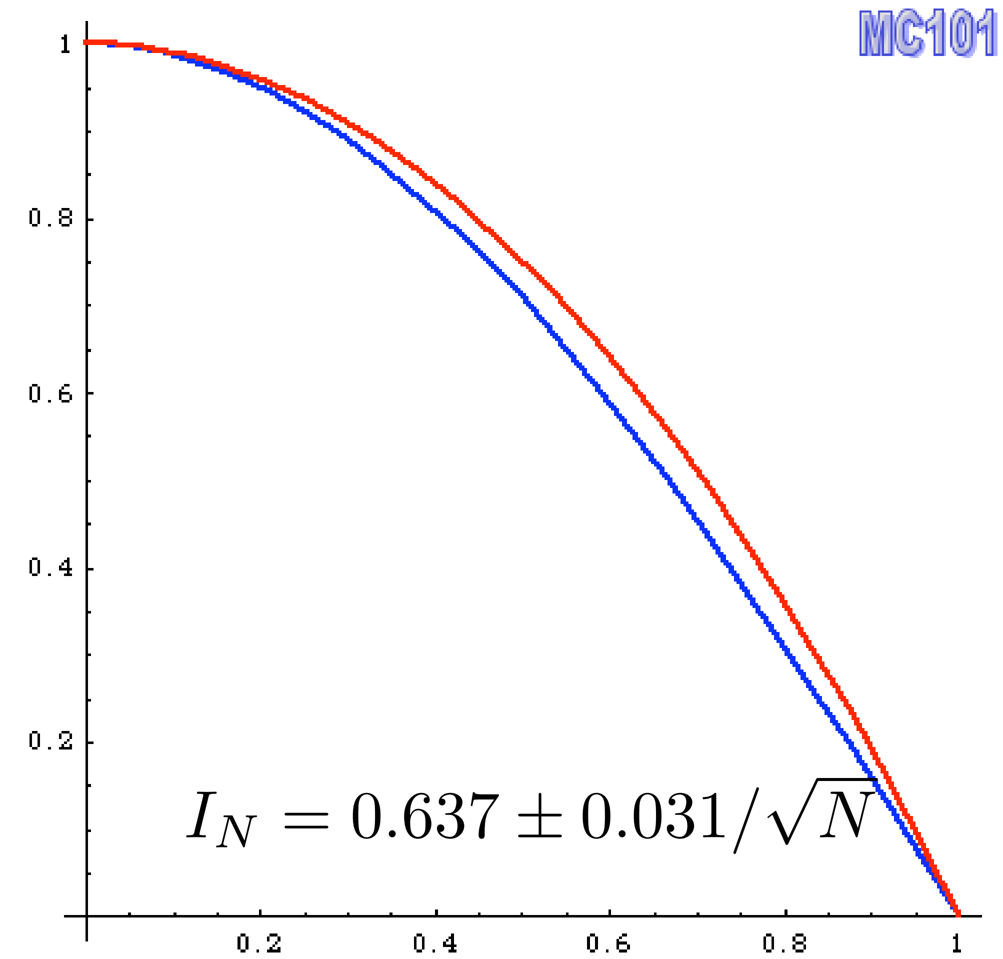
$$I = \int_0^1 dx \cos \frac{\pi}{2} x$$

$$I = \int_0^1 dx (1 - x^2) \frac{\cos \frac{\pi}{2} x}{1 - x^2}$$

$$= \int_{\xi_1}^{\xi_2} d\xi \frac{\cos \frac{\pi}{2} x[\xi]}{1 - x[\xi]^2}$$

Importance Sampling





$$I = \int_0^1 dx \cos \frac{\pi}{2} x$$

$$I = \int_0^1 dx (1 - x^2) \frac{\cos \frac{\pi}{2} x}{1 - x^2}$$

$$= \int_{\xi_1}^{\xi_2} d\xi \frac{\cos \frac{\pi}{2} x[\xi]}{1 - x[\xi]^2} \Rightarrow \simeq 1$$

Importance Sampling

but... you need to know a lot about f(x)!

Importance Sampling

but... you need to know a lot about f(x)!

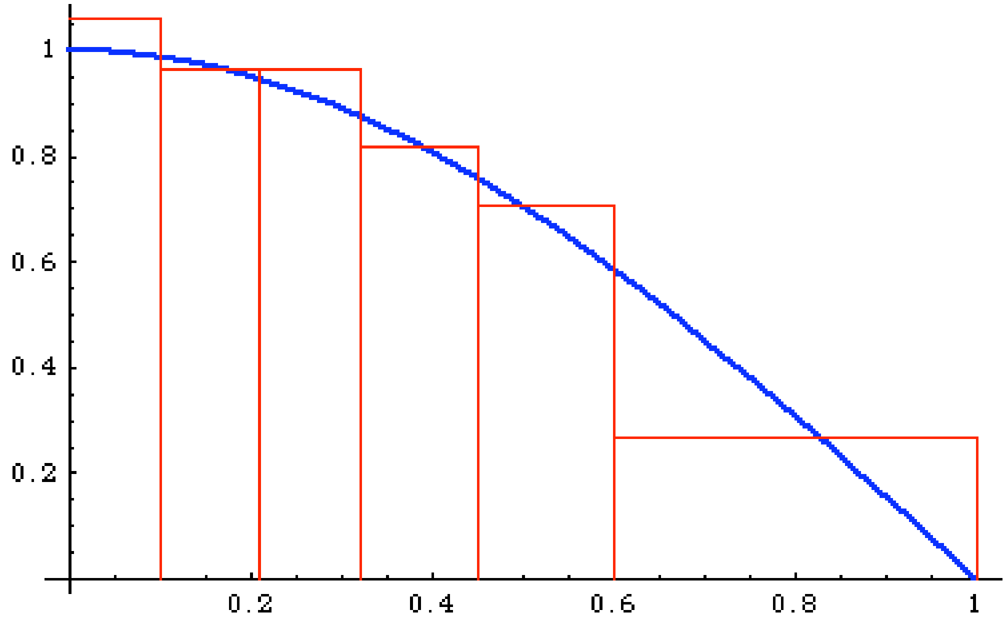
Alternative: learn during the run and build a step-function approximation p(x) of f(x) ⇒ VEGAS

Importance Sampling

but... you need to know a lot about f(x)!

Alternative: learn during the run and build a step-function approximation p(x) of f(x) ➡ VEGAS

MC101

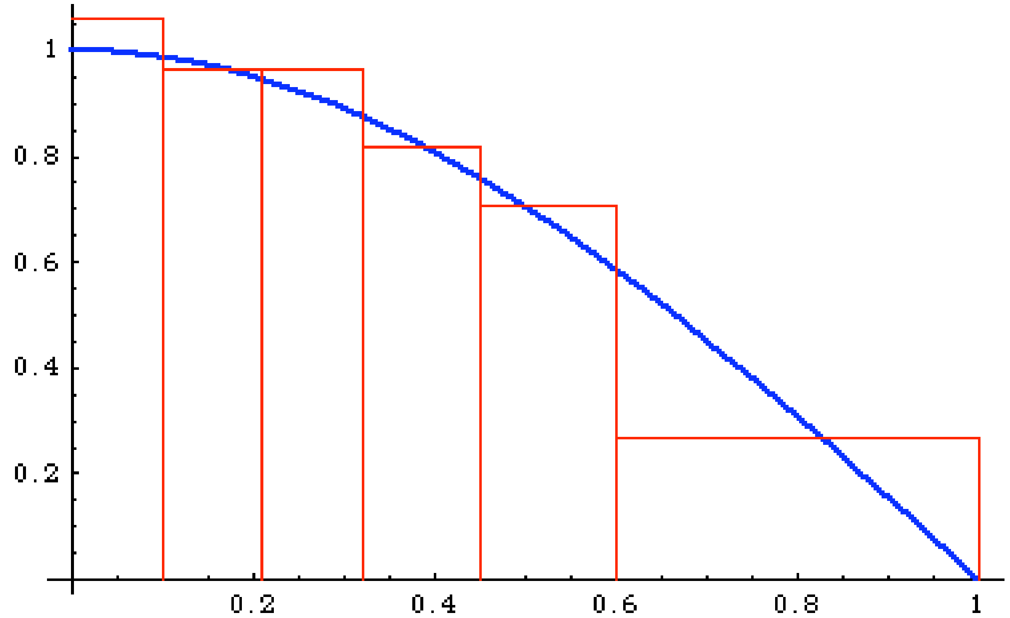


Importance Sampling

but... you need to know a lot about f(x)!

Alternative: learn during the run and build a step-function approximation p(x) of f(x) ⇨ VEGAS

MC101



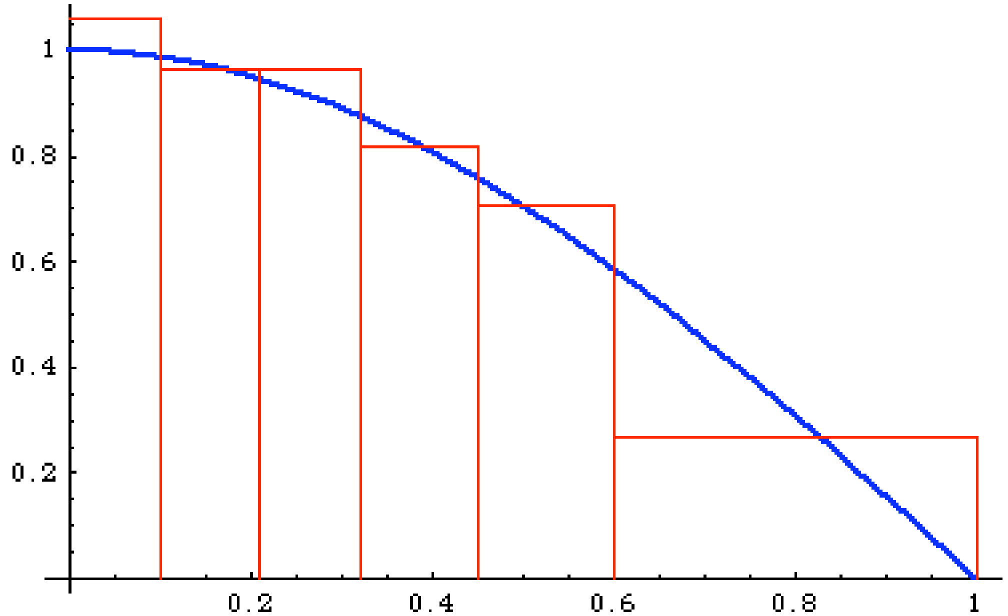
many bins where f(x) is large

Importance Sampling

but... you need to know a lot about f(x)!

Alternative: learn during the run and build a step-function approximation p(x) of f(x) ⇒ VEGAS

MC101



many bins where f(x) is large

$$p(x) = \frac{1}{N_b \Delta x_i}, \quad x_i - \Delta x_i < x < x_i$$

Importance Sampling

can be generalized to n dimensions:

$$p(\vec{x}) = p(x) \cdot p(y) \cdot p(z) \ldots$$

Importance Sampling

can be generalized to n dimensions:

$$p(\vec{x}) = p(x) \cdot p(y) \cdot p(z) \ldots$$

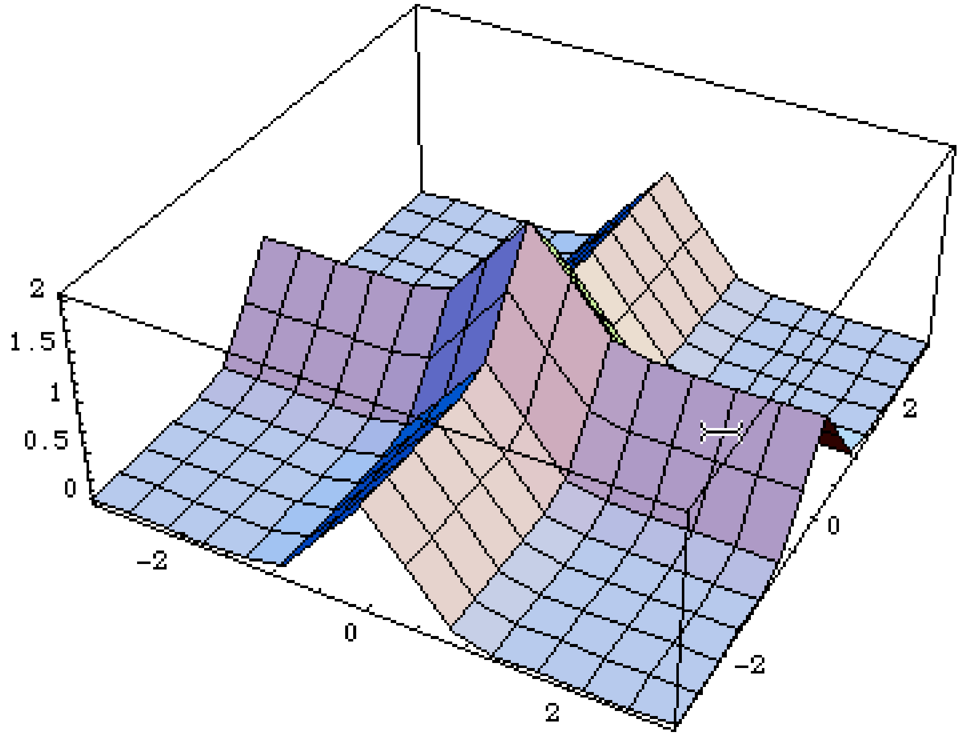
but the peaks of $f(\vec{x})$ need to be "aligned" to the axis!

Importance Sampling

can be generalized to n dimensions:

$$p(\vec{x}) = p(x) \cdot p(y) \cdot p(z) \ldots$$

but the peaks of $f(\vec{x})$ need to be "aligned" to the axis!



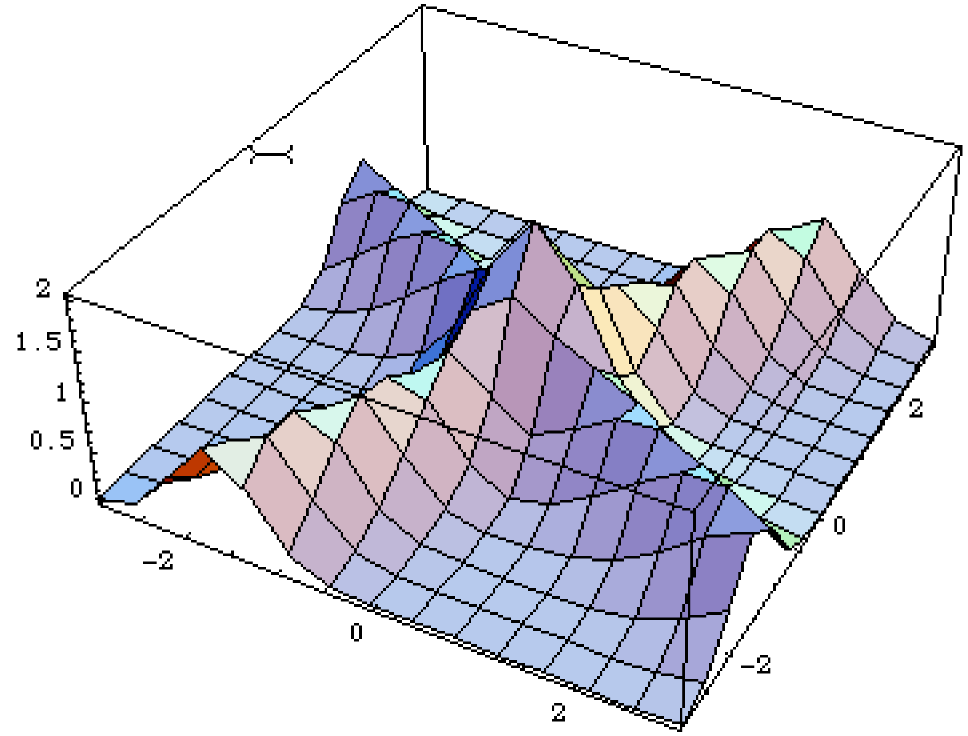
This is ok...

Importance Sampling

can be generalized to n dimensions:

$$p(\vec{x}) = p(x) \cdot p(y) \cdot p(z) \ldots$$

but the peaks of $f(\vec{x})$ need to be "aligned" to the axis!



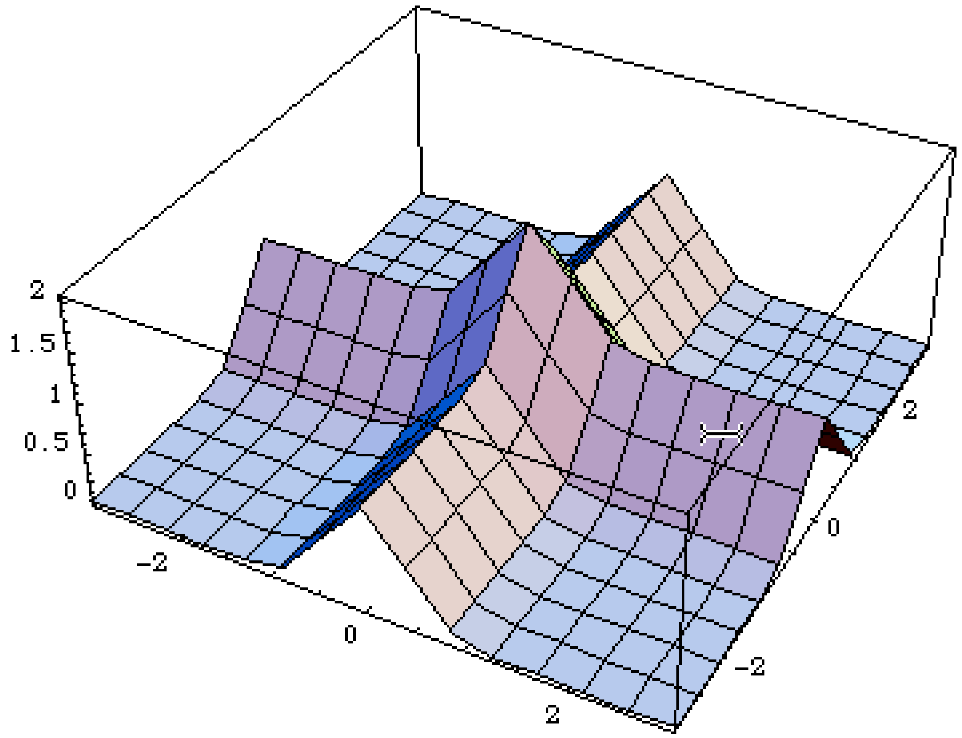
This is not ok...

Importance Sampling

can be generalized to n dimensions:

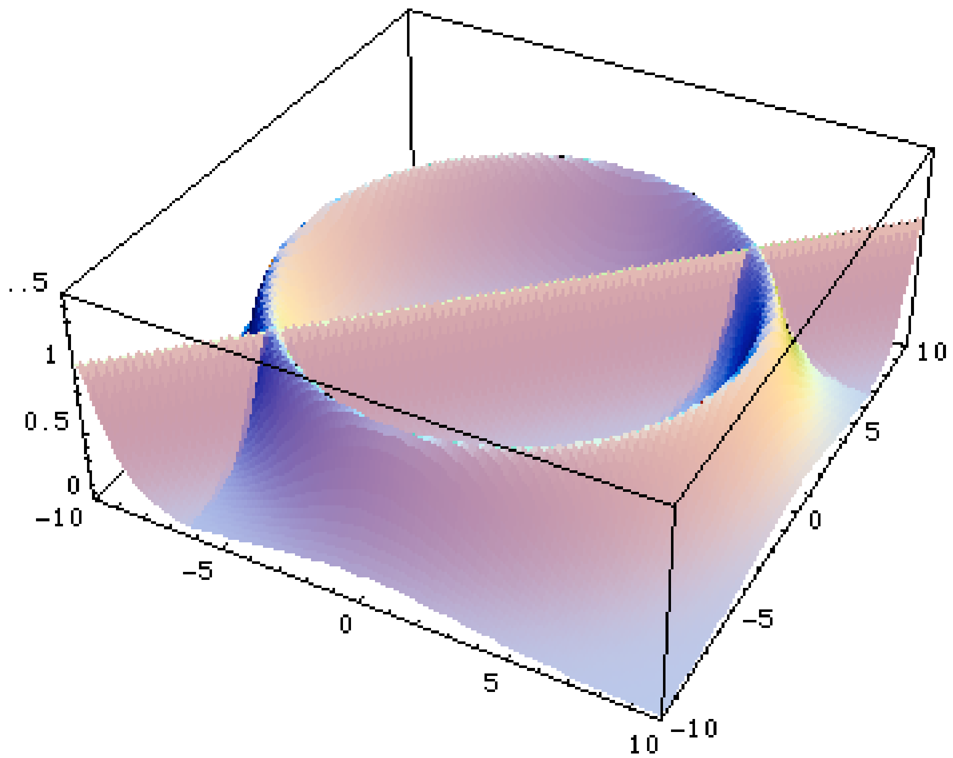
$$p(\vec{x}) = p(x) \cdot p(y) \cdot p(z) \ldots$$

but the peaks of $f(\vec{x})$ need to be "aligned" to the axis!



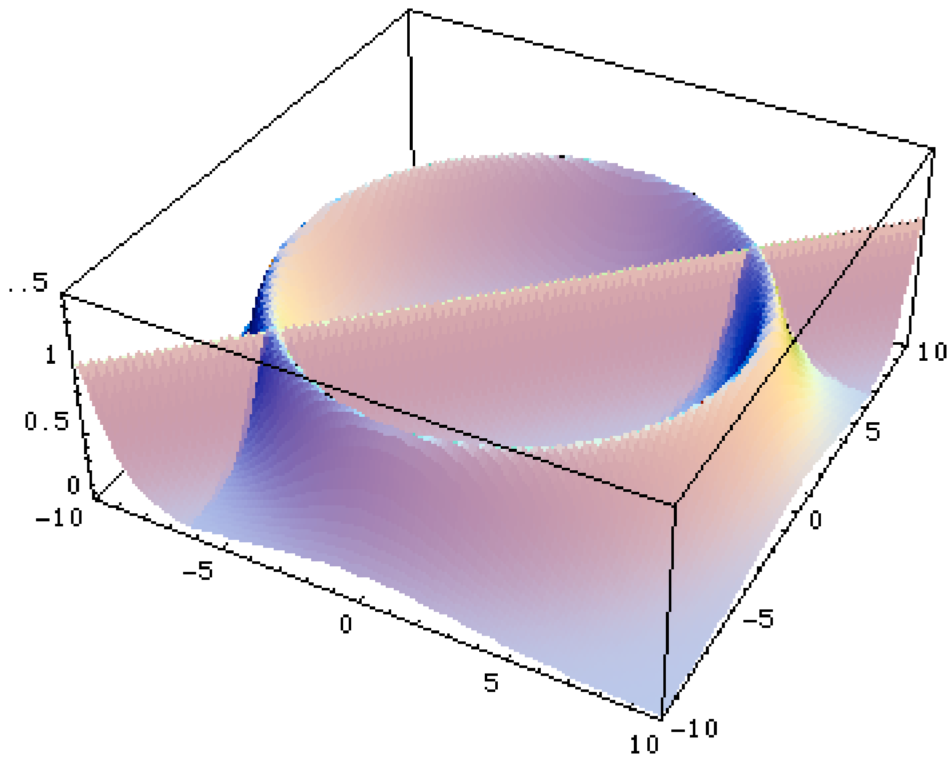
but it is sufficient to make a change of variables!

Multi-channel



What do we do if there is no transformation that aligns all integrand peaks to the chosen axes? Vegas is bound to fail!

Multi-channel



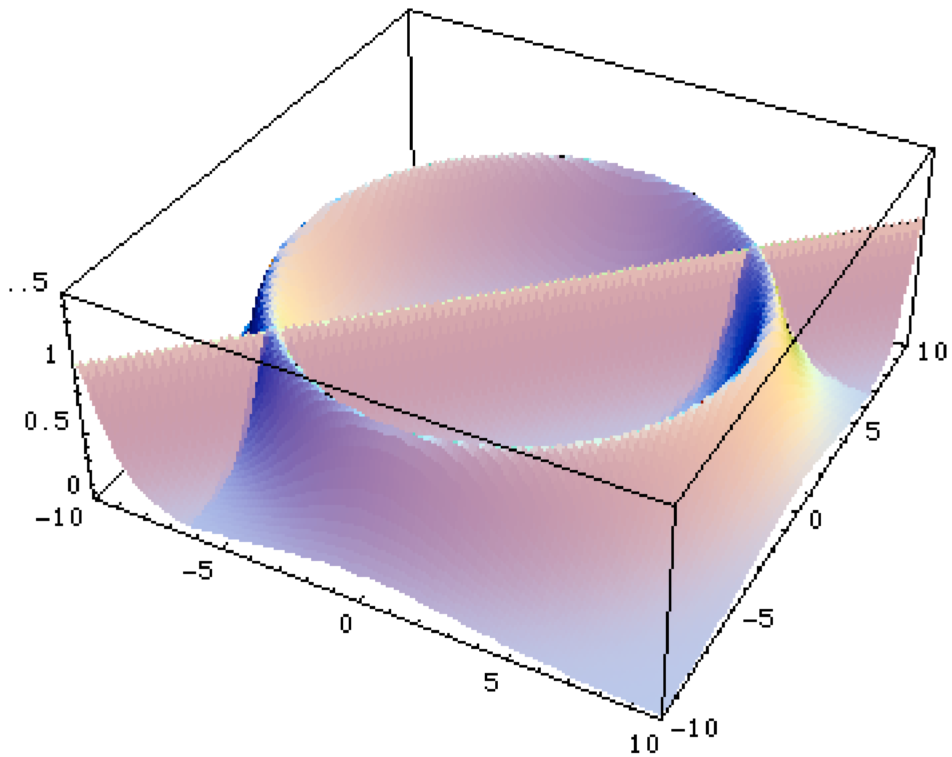
What do we do if there is no transformation that aligns all integrand peaks to the chosen axes? Vegas is bound to fail!

Solution: use different transformations = channels

$$p(x) = \sum_{i=1}^{n} \alpha_i p_i(x) \qquad \text{with} \qquad \sum_{i=1}^{n} \alpha_i = 1$$

with each $p_i(x)$ taking care of one "peak" at the time

Multi-channel

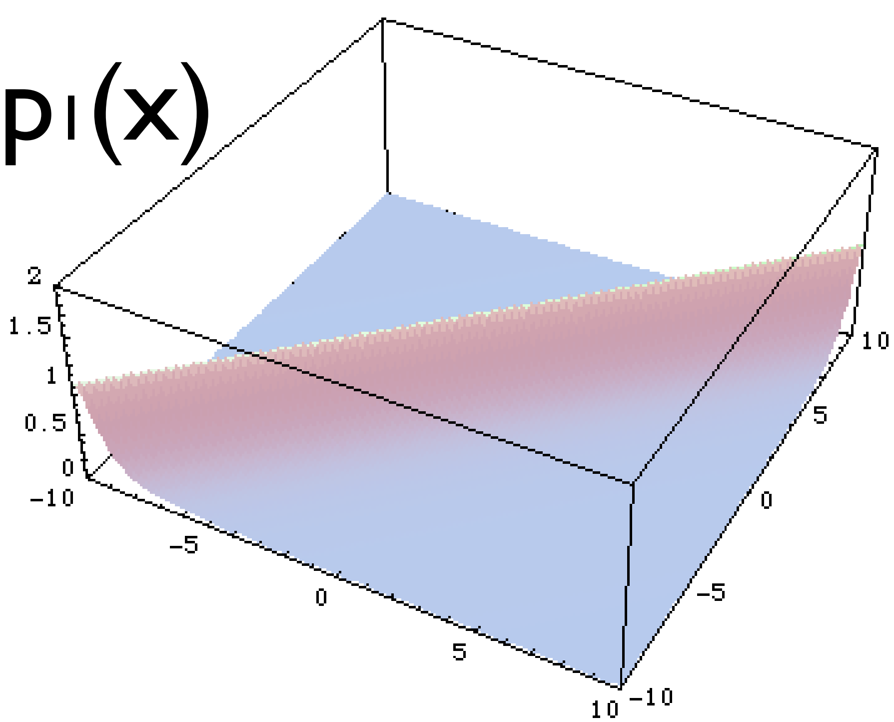


$$p(x) = \sum_{i=1}^{n} \alpha_i p_i(x)$$

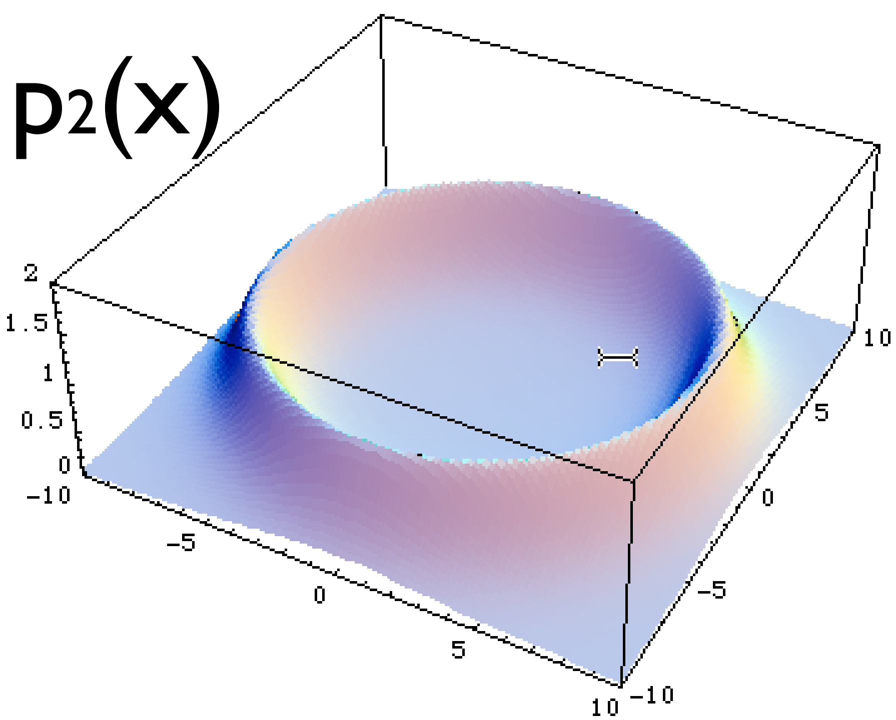
with

$$\sum_{i=1}^{n} \alpha_i = 1$$

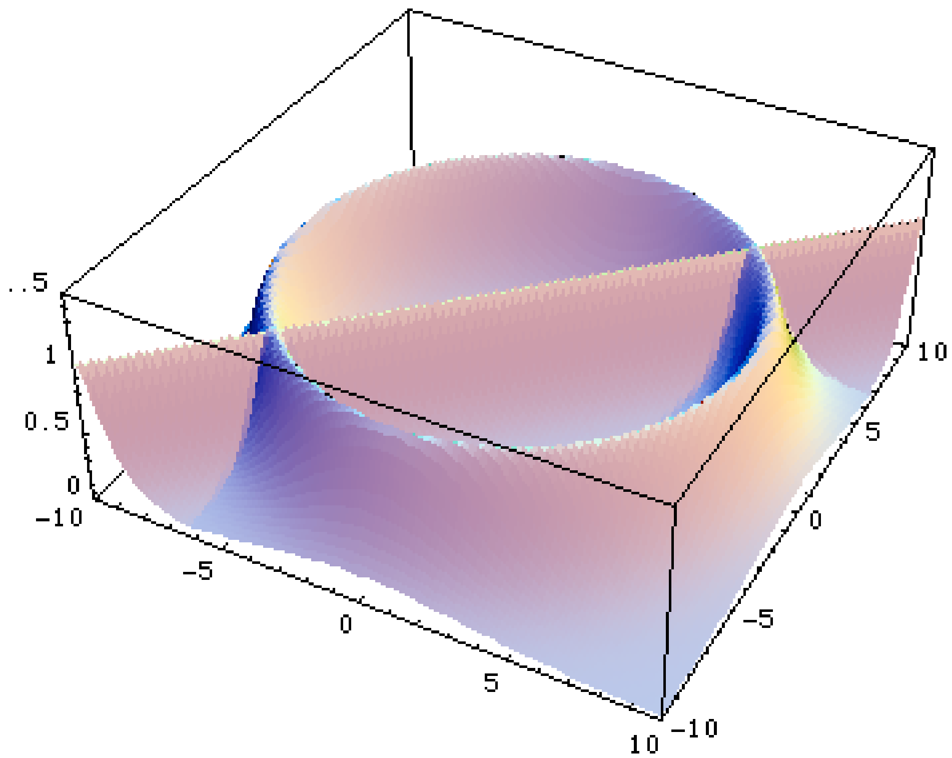
p1(x)



p2(x)



Multi-channel



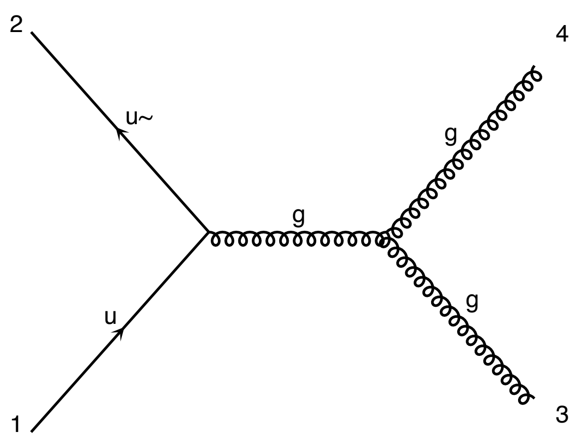
$$p(x) = \sum_{i=1}^{n} \alpha_i p_i(x)$$

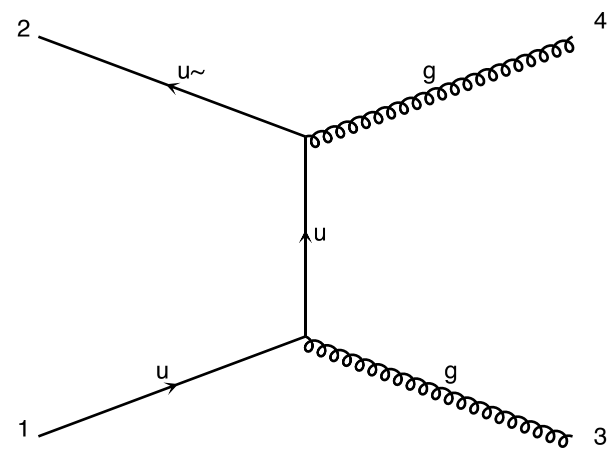
with $$\sum_{i=1}^{n} \alpha_i = 1$$

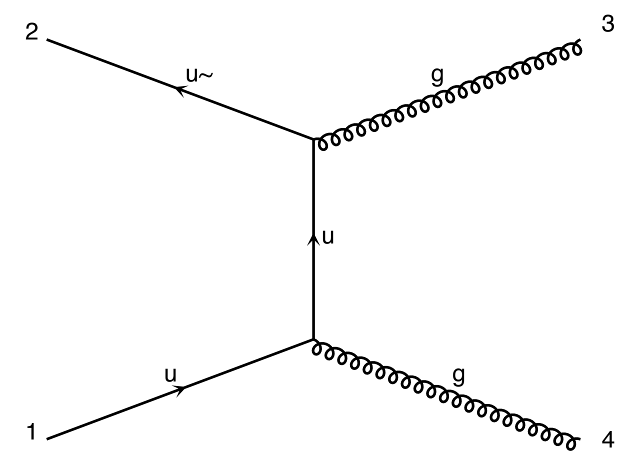
Then,

$$I = \int f(x)dx = \sum_{i=1}^{n} \alpha_i \int \frac{f(x)}{p(x)} p_i(x)dx$$

Example: QCD 2 → 2 production







$$\propto \frac{1}{\hat{s}} = \frac{1}{(p_1 + p_2)^2} \qquad \propto \frac{1}{\hat{t}} = \frac{1}{(p_1 - p_3)^2} \qquad \propto \frac{1}{\hat{u}} = \frac{1}{(p_1 - p_4)^2}$$

Three very different pole structures contributing to the same matrix element.

Multi-channel based on single diagrams

Consider the integration of an amplitude |M|^2 at tree level which many contributing diagrams. We would like to have a basis of functions,

$$f = \sum_{i=1}^{n} f_i \quad \text{with} \quad f_i \geq 0\,, \quad \forall \;\; i\,,$$

such that:

1. we know how to integrate each one of them,
2. they describe all possible peaks,

giving us the combined integral

$$I = \int d\vec{\Phi} f(\vec{\Phi}) = \sum_{i=1}^{n} \int d\vec{\Phi}\, g_i(\vec{\Phi})\, \frac{f_i(\vec{\Phi})}{g_i(\vec{\Phi})} = \sum_{i=1}^{n} I_i\,,$$

Multi-channel based on single diagrams

Consider the integration of an amplitude |M|^2 at tree level which many contributing diagrams. We would like to have a basis of functions,

$$f = \sum_{i=1}^{n} f_i \quad \text{with} \quad f_i \geq 0\,, \quad \forall \;\; i\,,$$

such that:

1. we know how to integrate each one of them,
2. they describe all possible peaks,

giving us the combined integral

$$I = \int d\vec{\Phi} f(\vec{\Phi}) = \sum_{i=1}^{n} \int d\vec{\Phi}\, g_i(\vec{\Phi})\, \frac{f_i(\vec{\Phi})}{g_i(\vec{\Phi})} = \sum_{i=1}^{n} I_i\,,$$

Does such a basis exist?

Multi-channel based on single diagrams*

YES! $$f_i = \frac{|A_i|^2}{\sum_i |A_i|^2} |A_{\rm tot}|^2$$

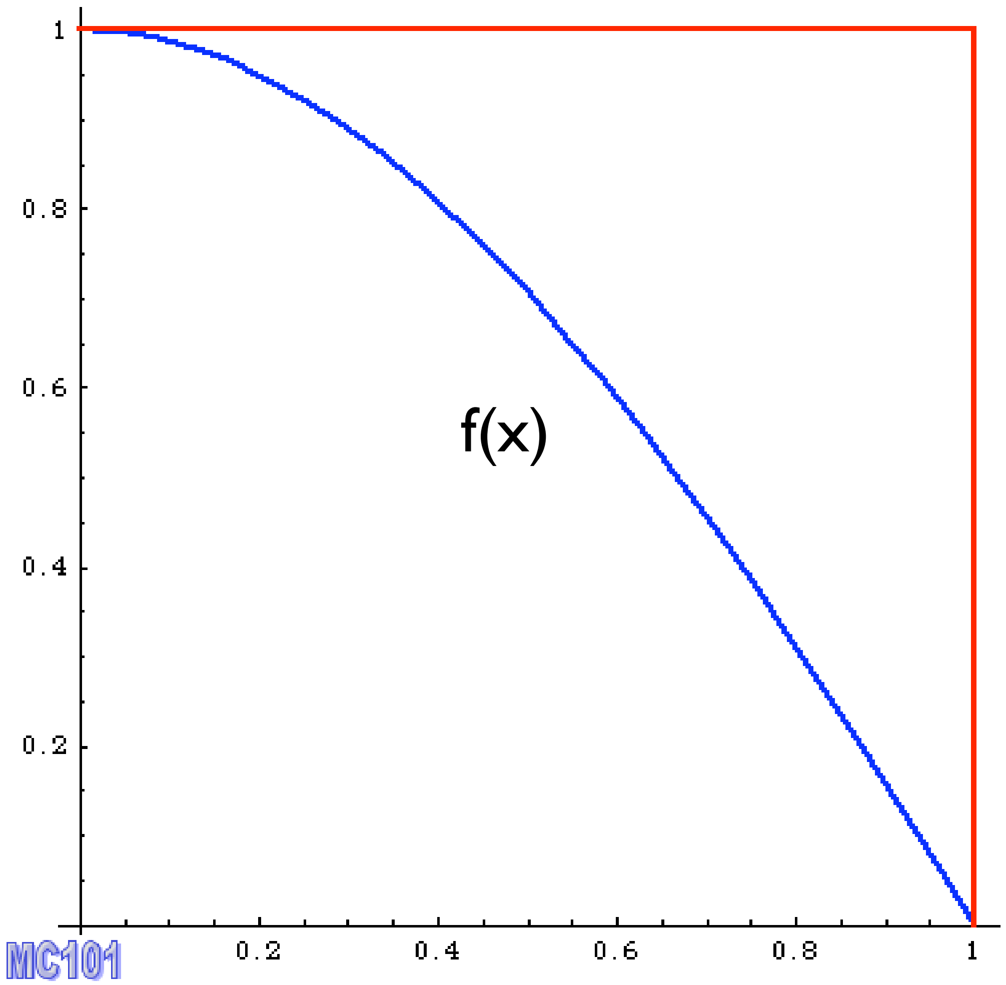
Multi-channel based on single diagrams*

YES! $$f_i = \frac{|A_i|^2}{\sum_i |A_i|^2} |A_{\rm tot}|^2$$

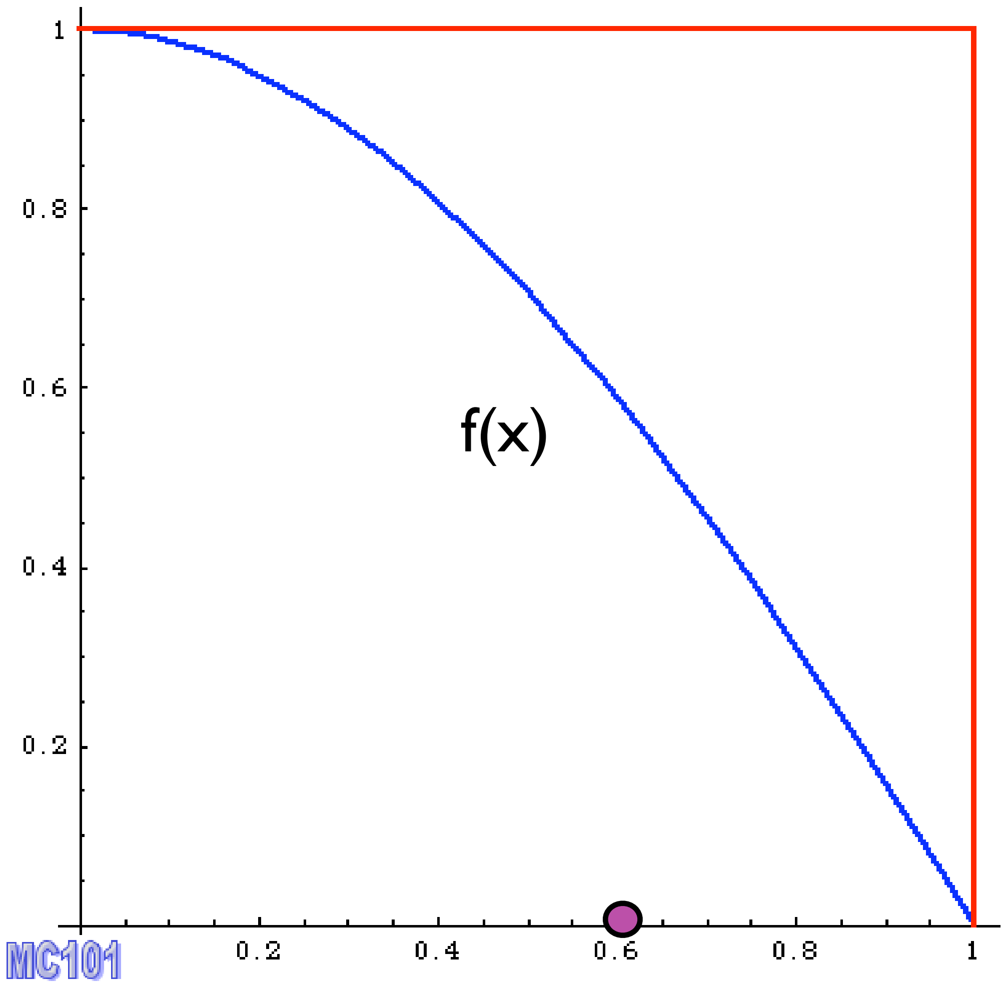
- Key Idea
  - Any single diagram is "easy" to integrate (pole structures/ suitable integration variables known from the propagators)
  - Divide integration into pieces, based on diagrams
  - All other peaks taken care of by denominator sum
- Get N independent integrals
  - Errors add in quadrature so no extra cost
  - "Weight" functions already calculated during $|M|^2$ calculation
  - Parallel in nature
- What about interference?
  - Never creates "new" peaks, so we're OK!

*Method used in MadGraph

Monte Carlo Event Generation

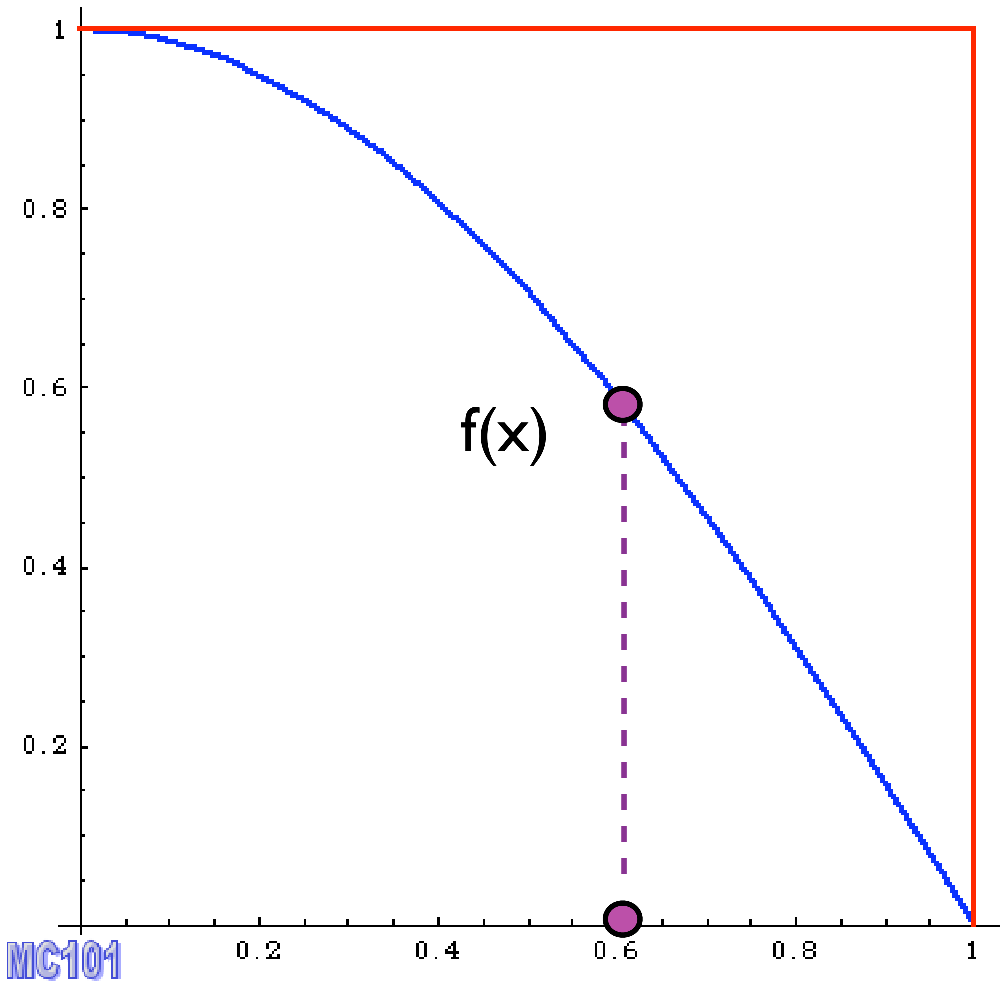


Monte Carlo Event Generation



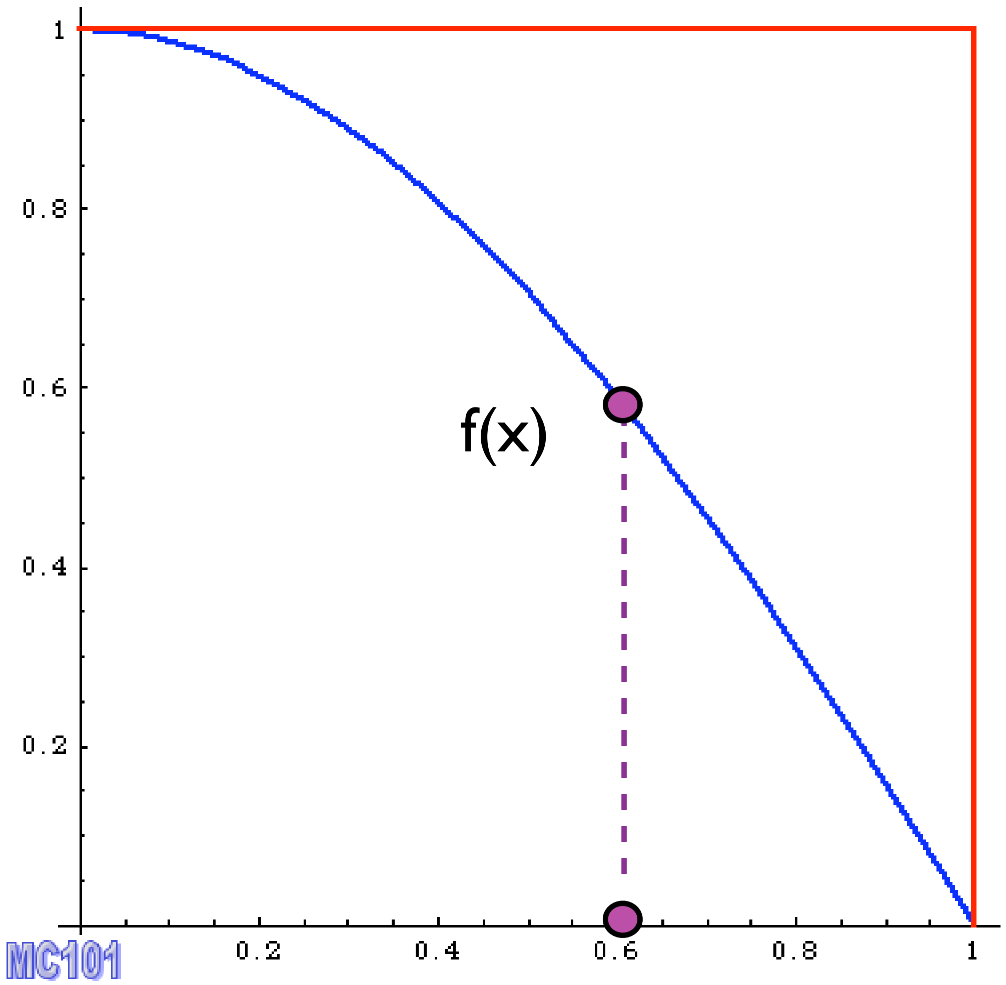
1. pick x

Monte Carlo Event Generation



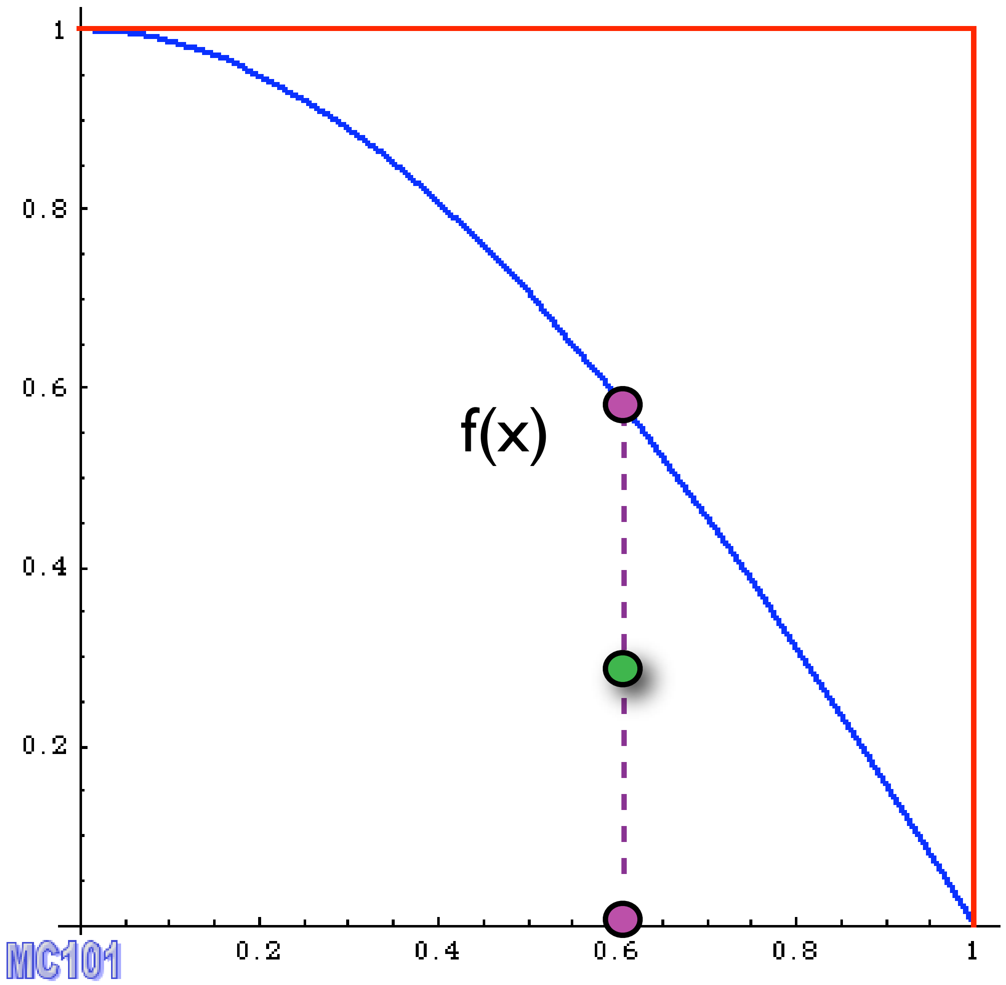
1. pick x
2. calculate f(x)

Monte Carlo Event Generation



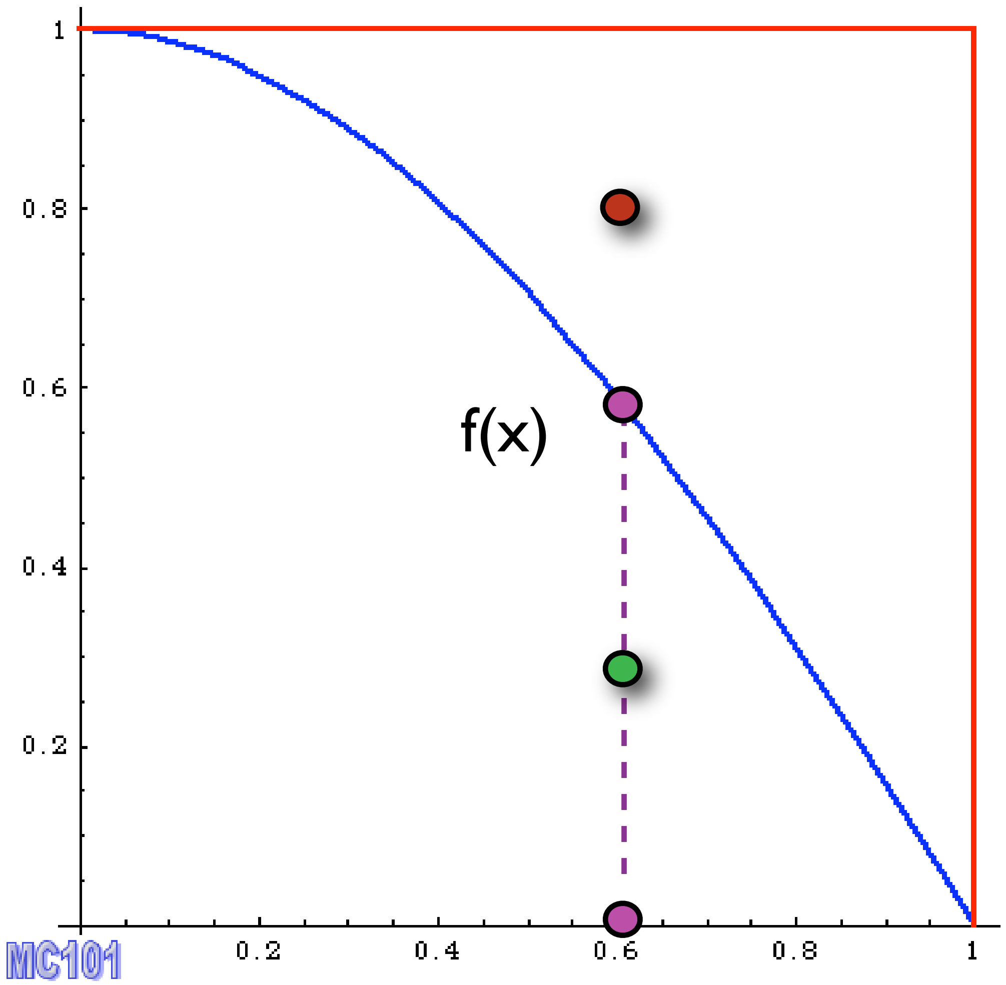
1. pick x
2. calculate f(x)
3. pick 0<y<fmax

Monte Carlo Event Generation



1. pick x
2. calculate f(x)
3. pick 0<y<fmax
4. Compare:
   if f(x)>y accept event,

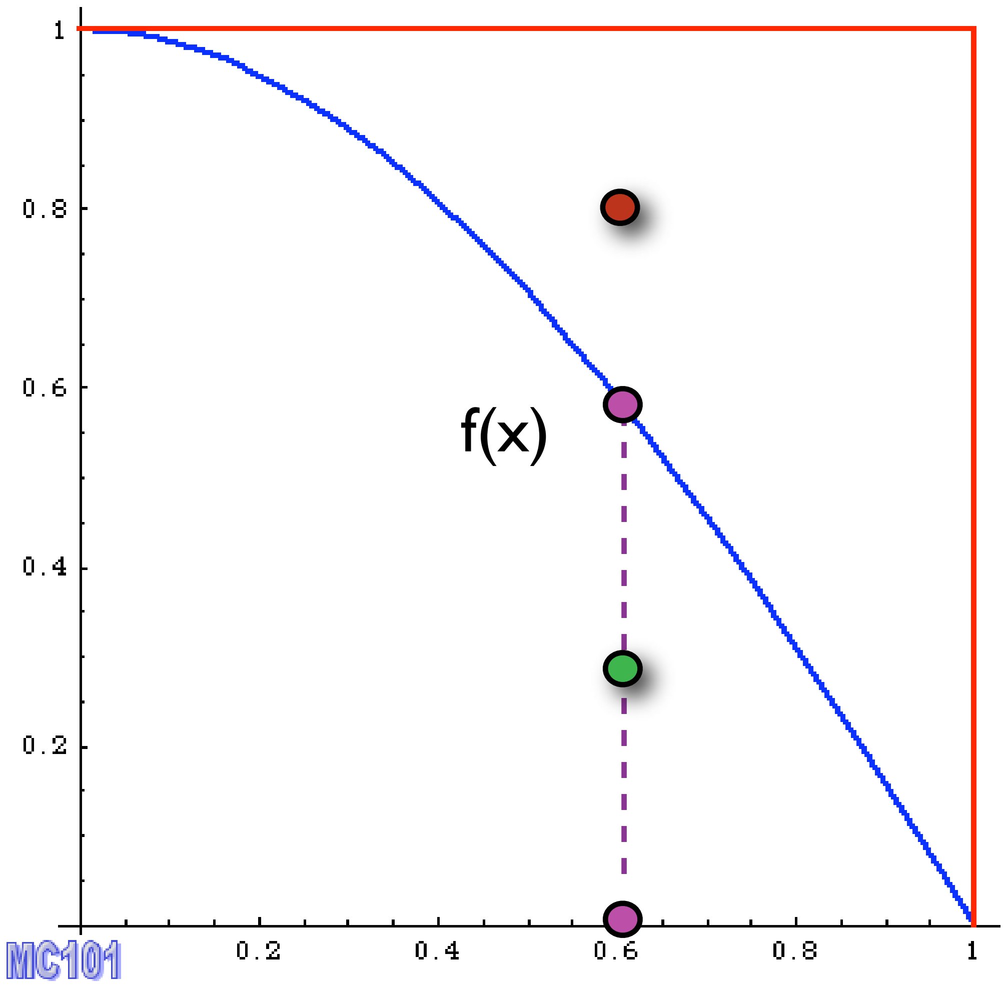
Monte Carlo Event Generation



1. pick x
2. calculate f(x)
3. pick 0<y<fmax
4. Compare:
   if f(x)>y accept event,

   else reject it.

Monte Carlo Event Generation

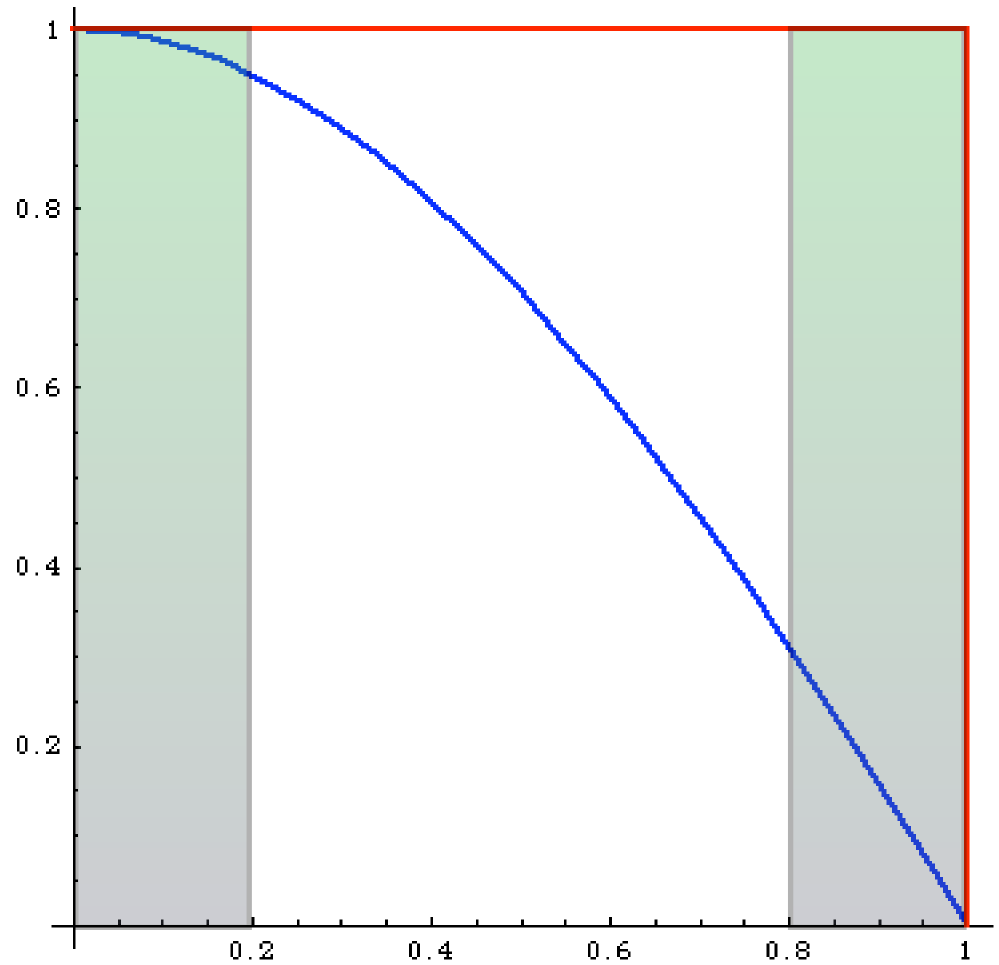


1. pick x
2. calculate f(x)
3. pick 0<y<fmax
4. Compare:
   if f(x)>y accept event,

   else reject it.

$$I = \frac{\text{accepted}}{\text{total tries}} = \text{efficiency}$$

Event generation

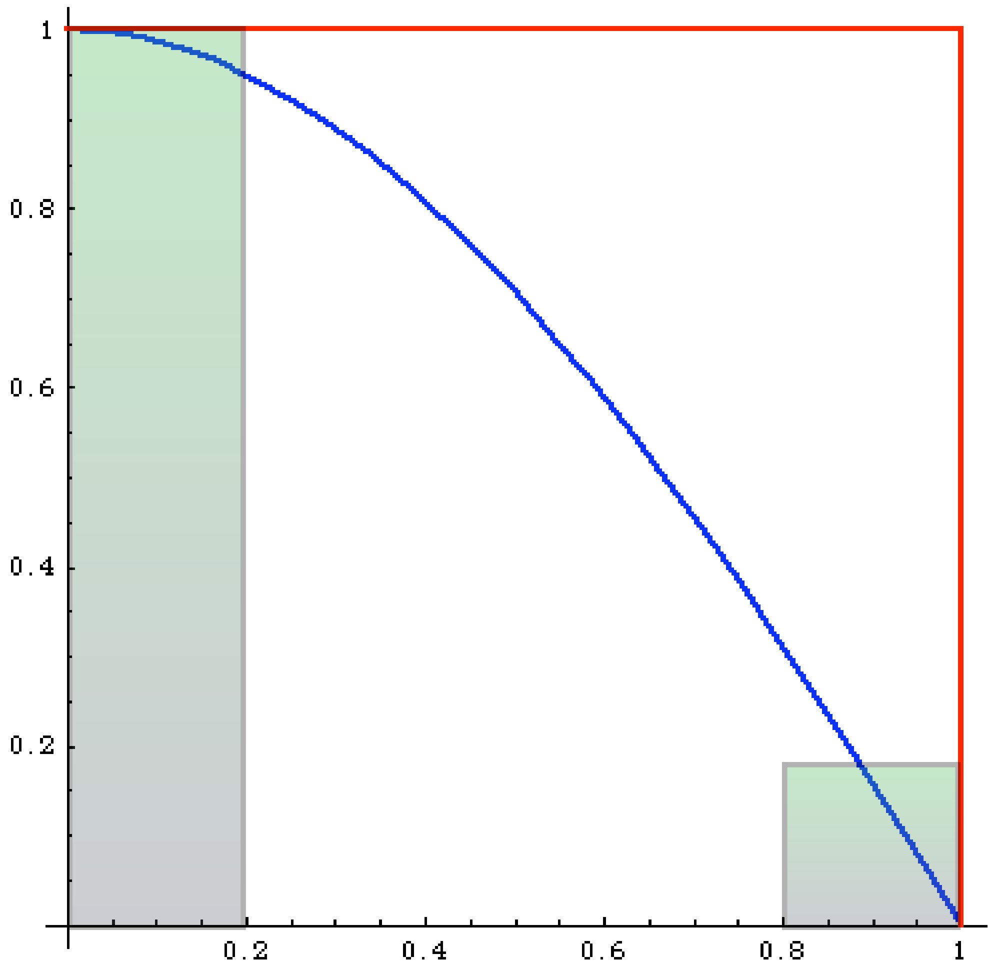


What's the difference between weighted and unweighted?

Weighted:

Same # of events in areas of phase space with very different probabilities: events must have different weights

Event generation



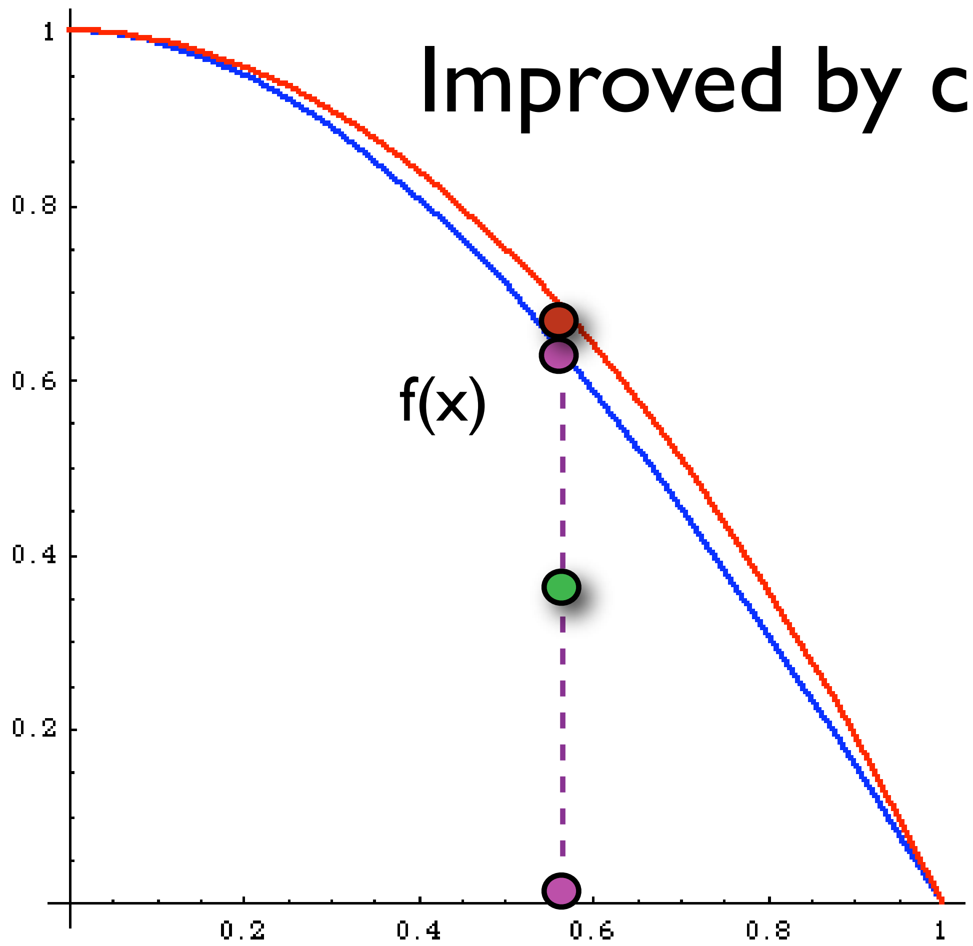
What's the difference between weighted and unweighted?

Unweighted:

events is proportional to the probability of areas of phase space:
events have all the same weight ("unweighted")

Events distributed as in nature

Event generation



Improved by combining with importance sampling:

1. pick x distributed as p(x)
2. calculate f(x) and p(x)
3. pick 0<y<1
4. Compare:
   if f(x)>y p(x) accept event,

else reject it.

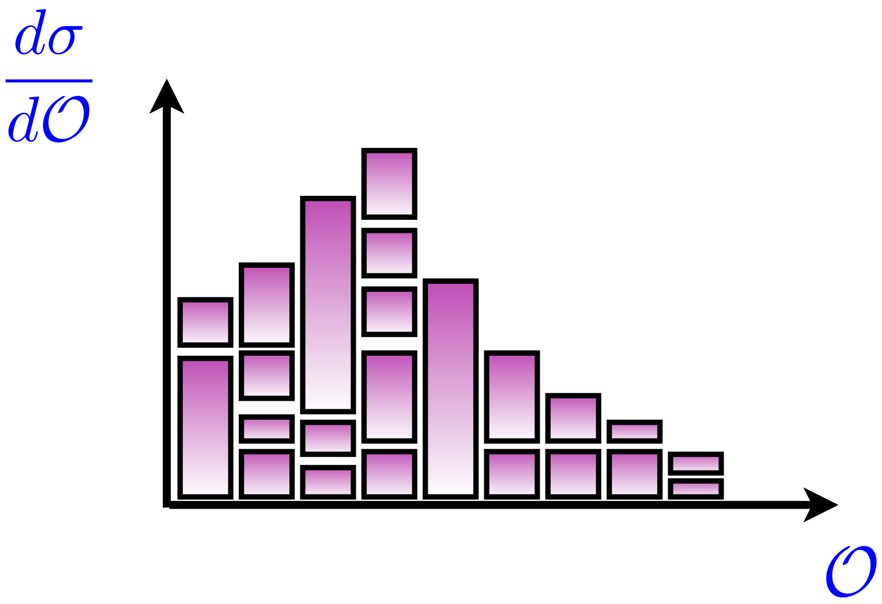
much better efficiency!!!

Event generation

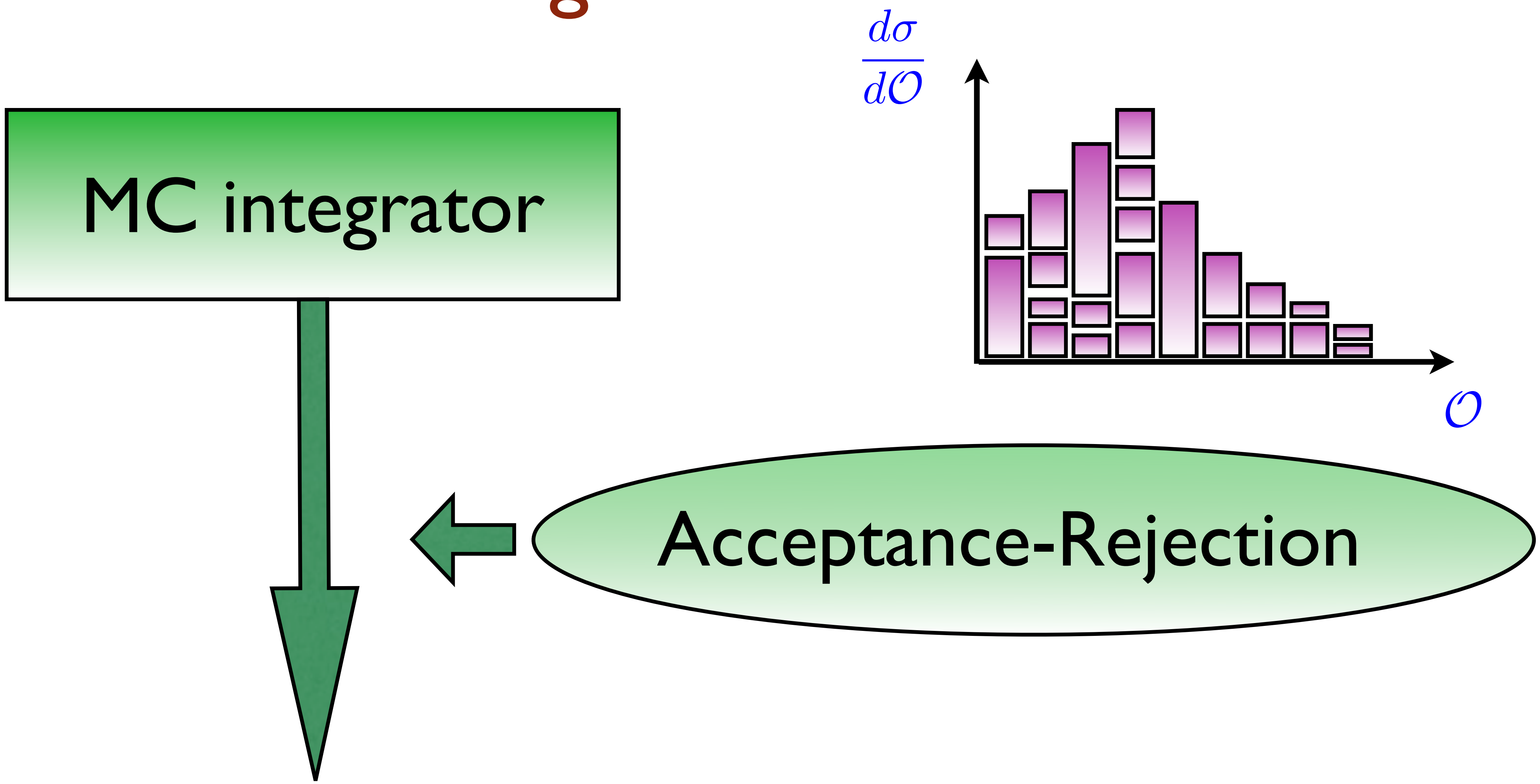
MC integrator

Event generation

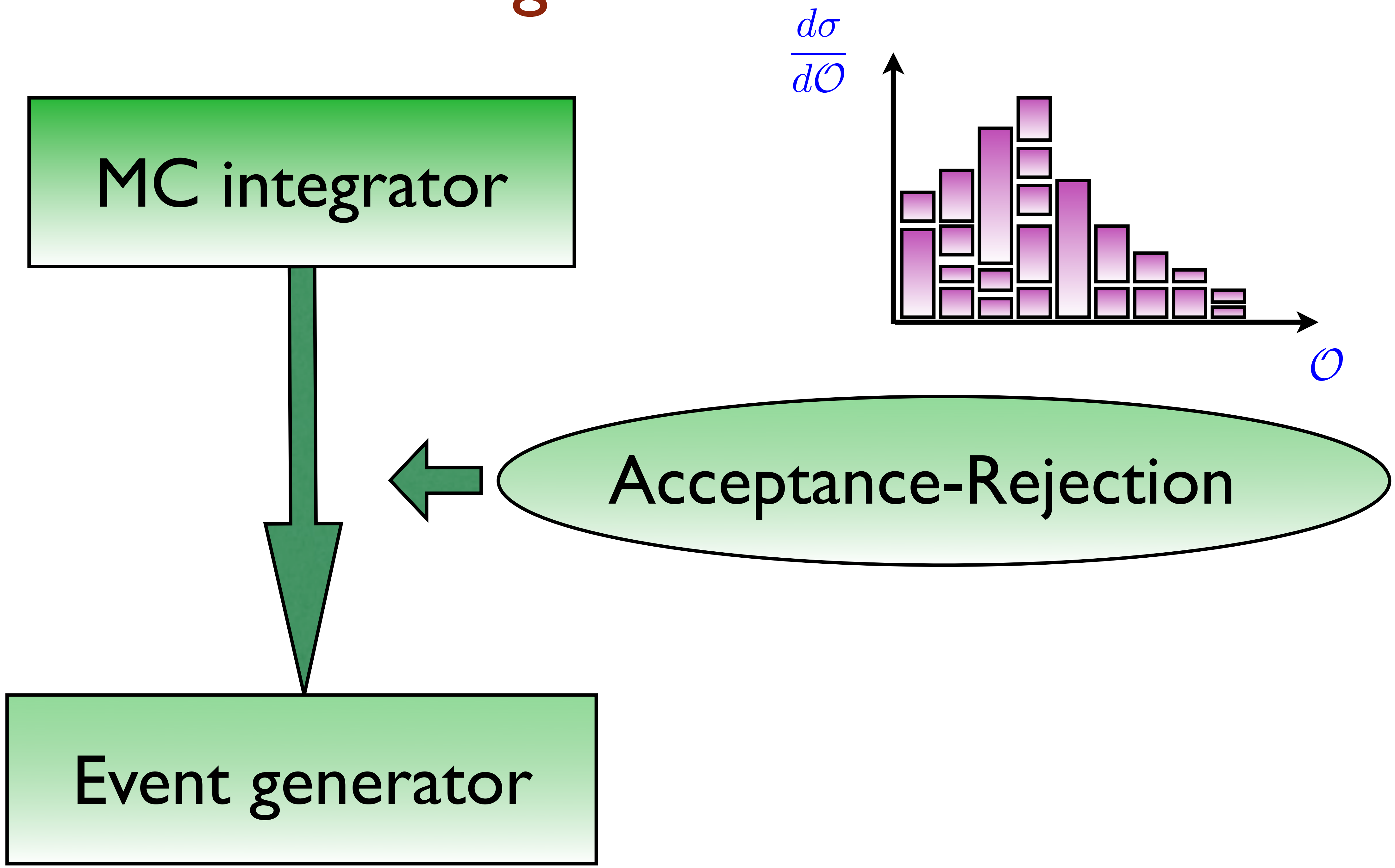
MC integrator



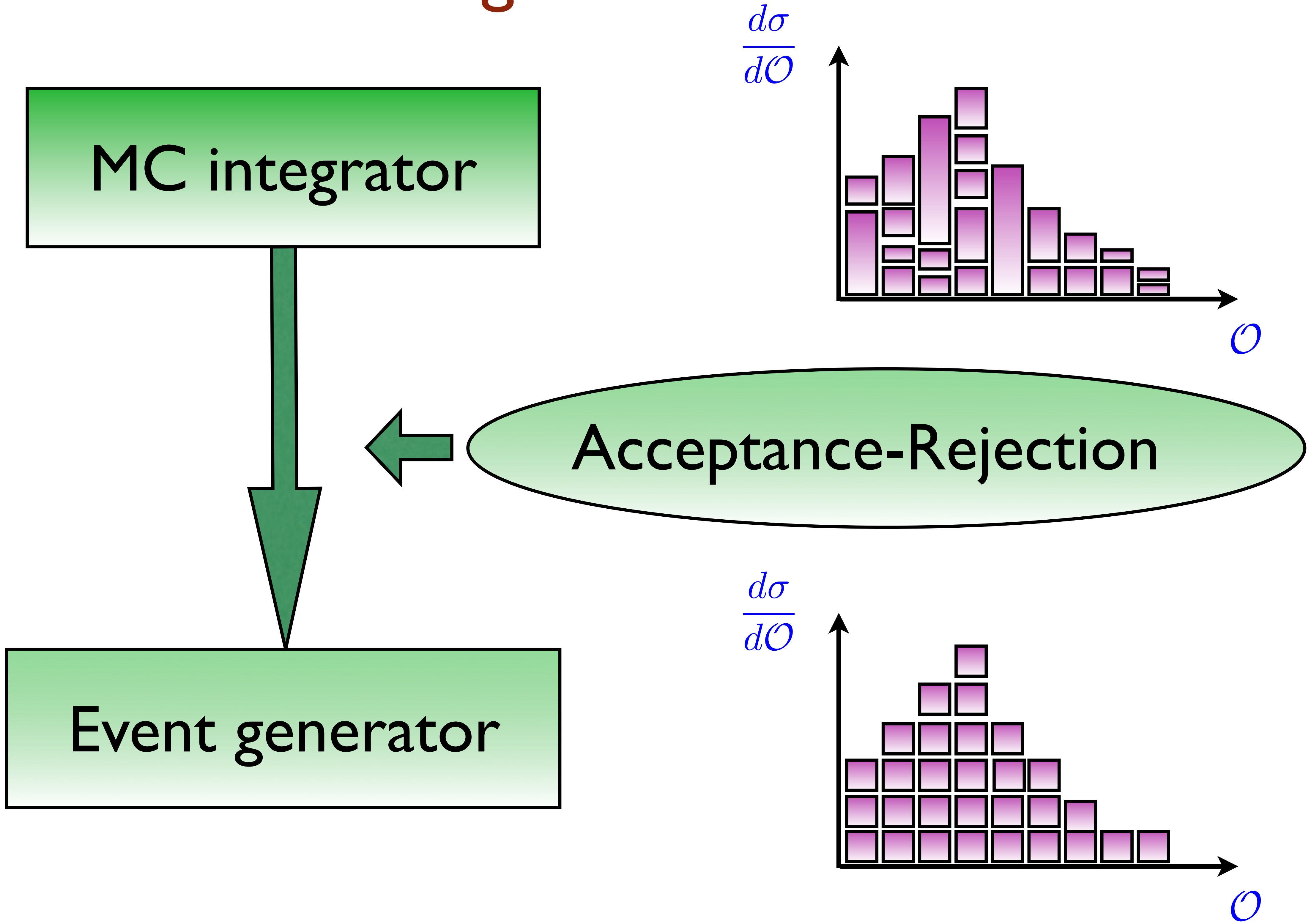
Event generation



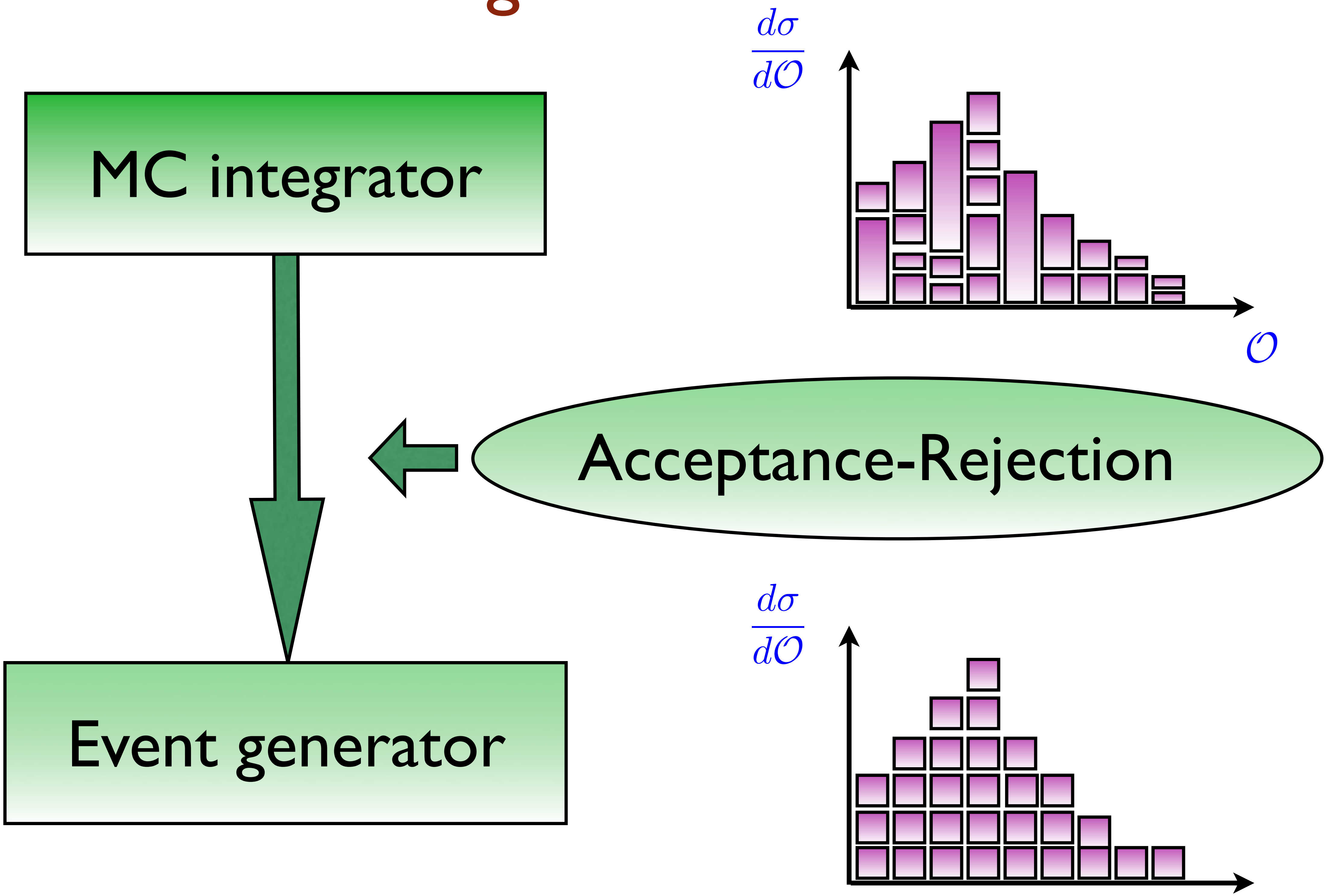
Event generation



Event generation

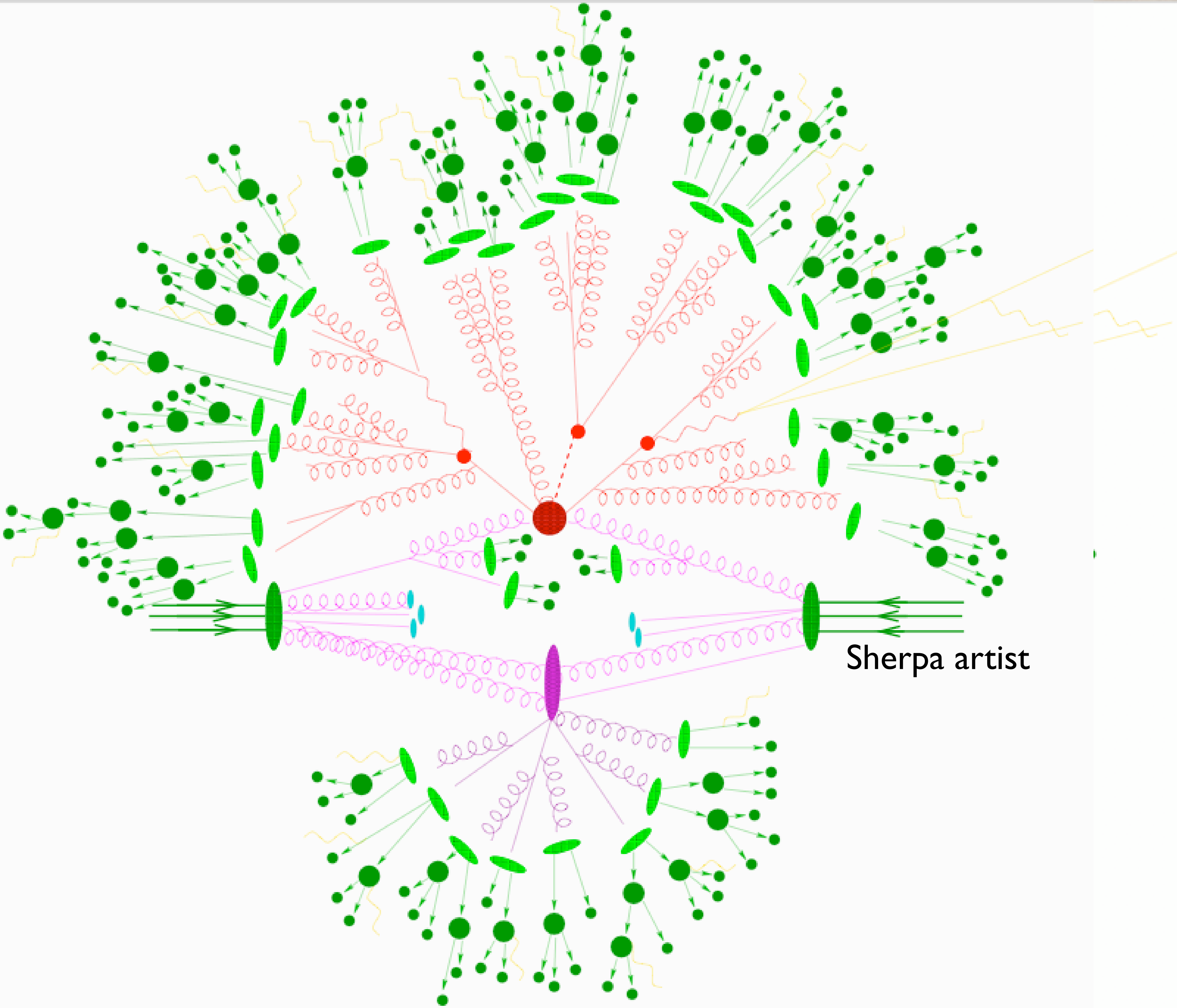


Event generation



☞ **This is possible only if f(x)<∞ AND has definite sign!**

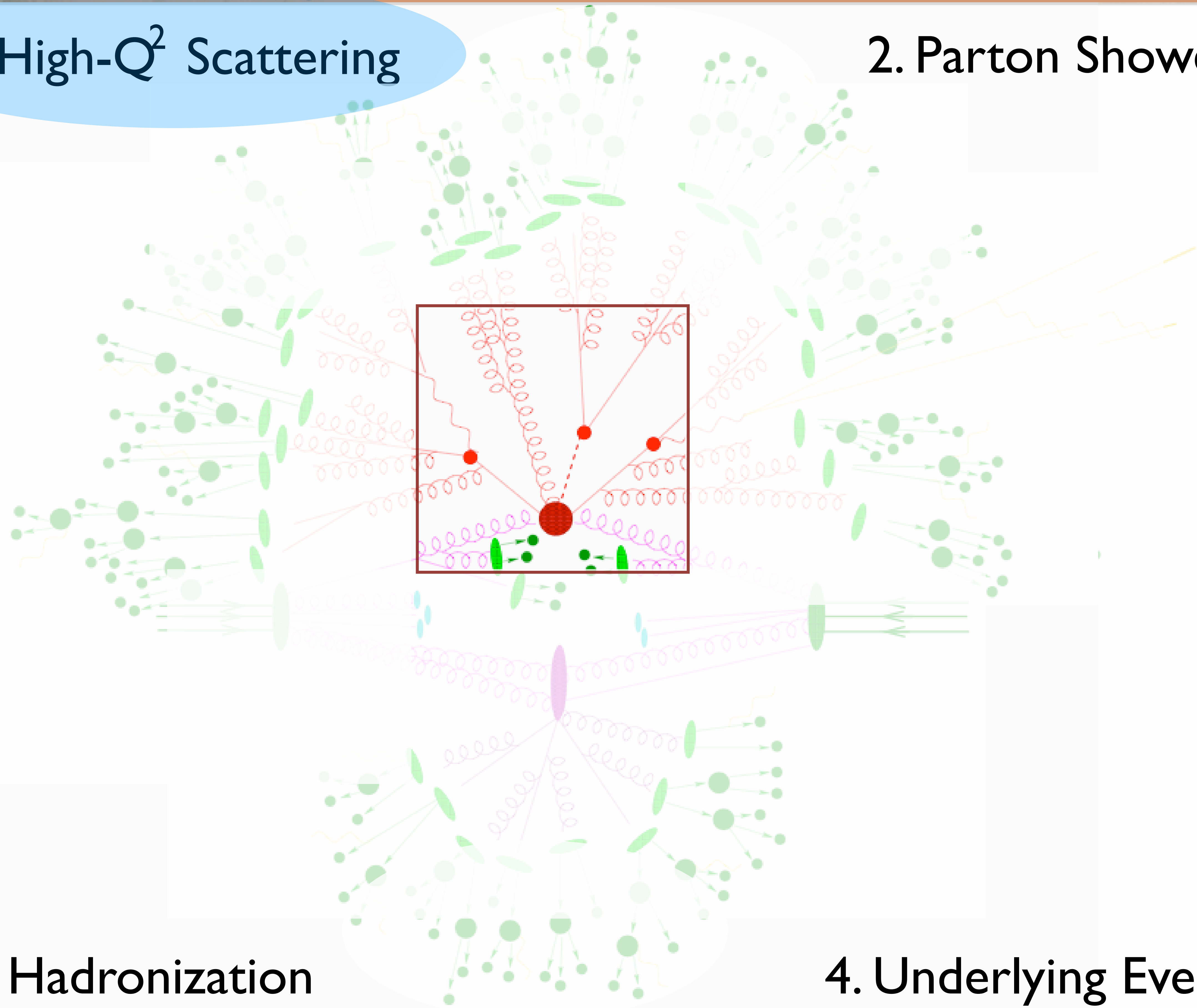
Simulation of collider events



Sherpa artist

1. High-Q^2 Scattering

2. Parton Shower



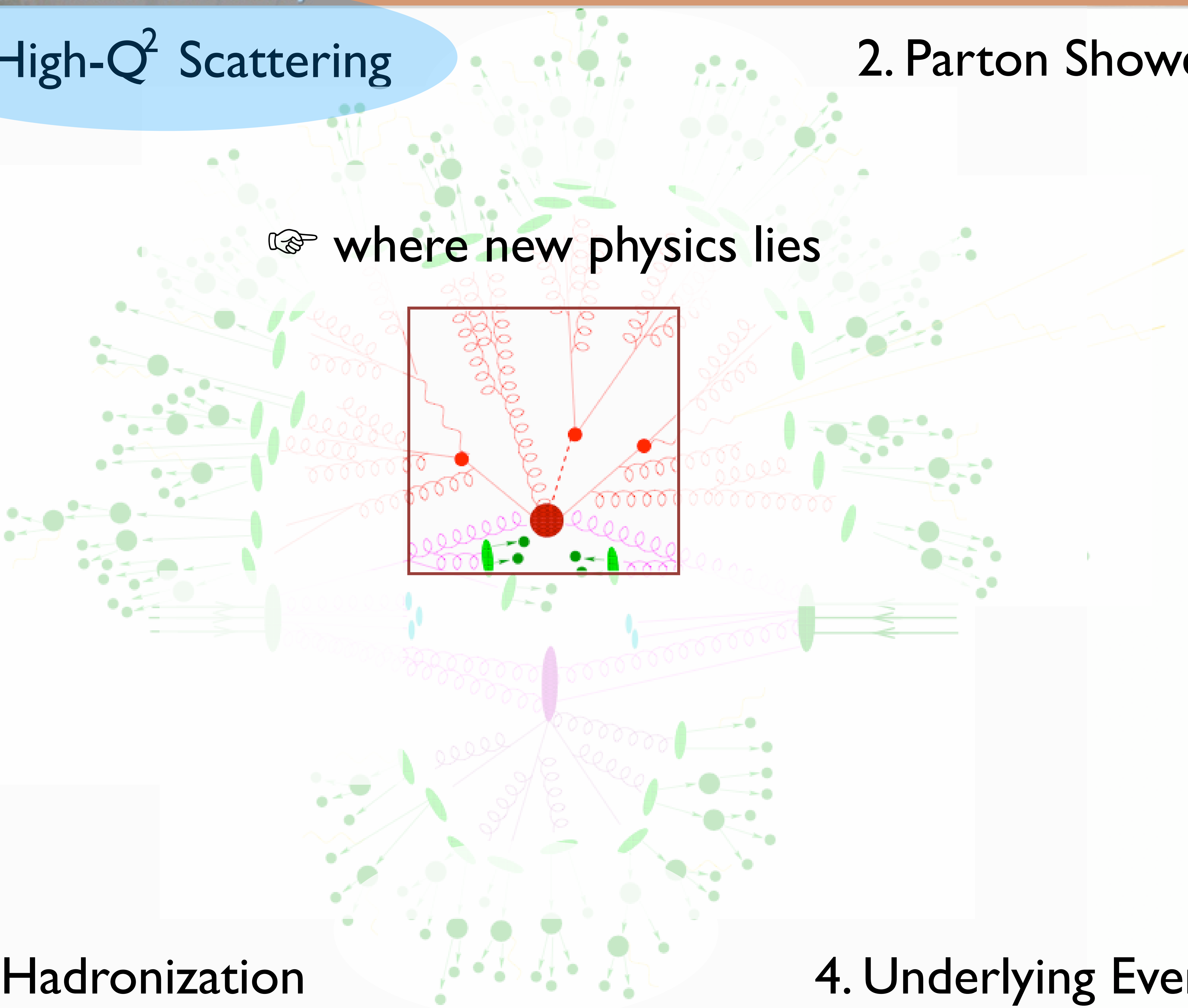
3. Hadronization

4. Underlying Event

1. High-Q^2 Scattering

2. Parton Shower

☞ where new physics lies



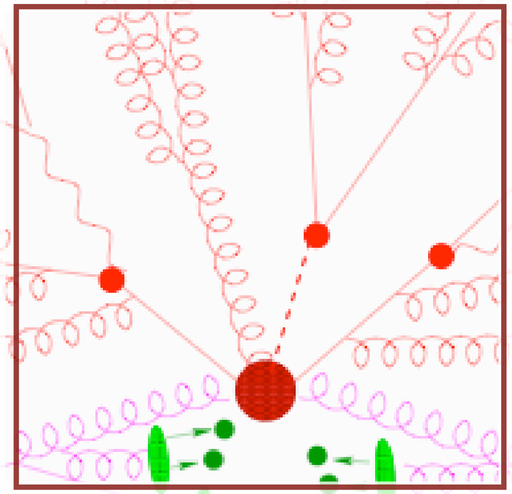
3. Hadronization

4. Underlying Event

1. High-Q^2 Scattering

2. Parton Shower

☞ where new physics lies



☞ process dependent

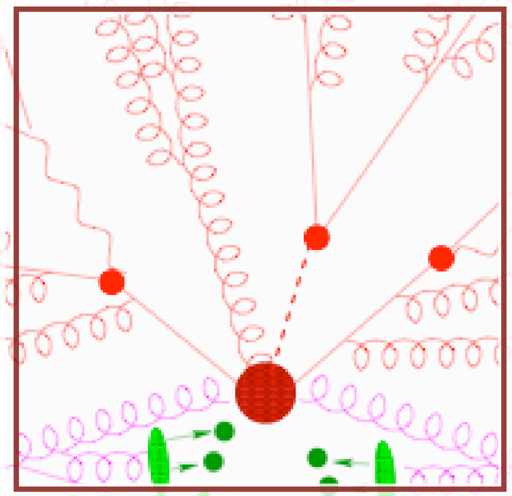
3. Hadronization

4. Underlying Event

1. High-Q^2 Scattering

2. Parton Shower

☞ where new physics lies



☞ process dependent

☞ first principles description

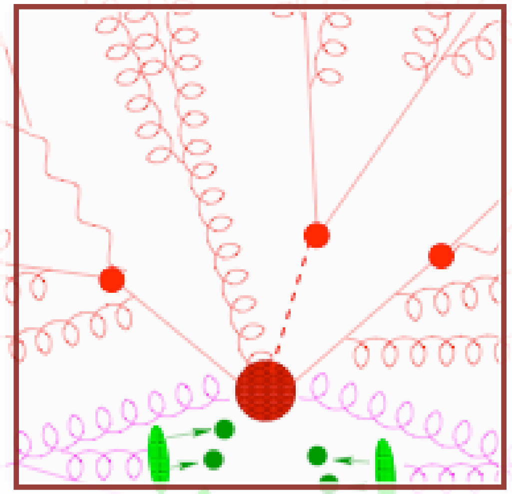
3. Hadronization

4. Underlying Event

1. High-Q^2 Scattering

2. Parton Shower

☞ where new physics lies



☞ process dependent

☞ first principles description

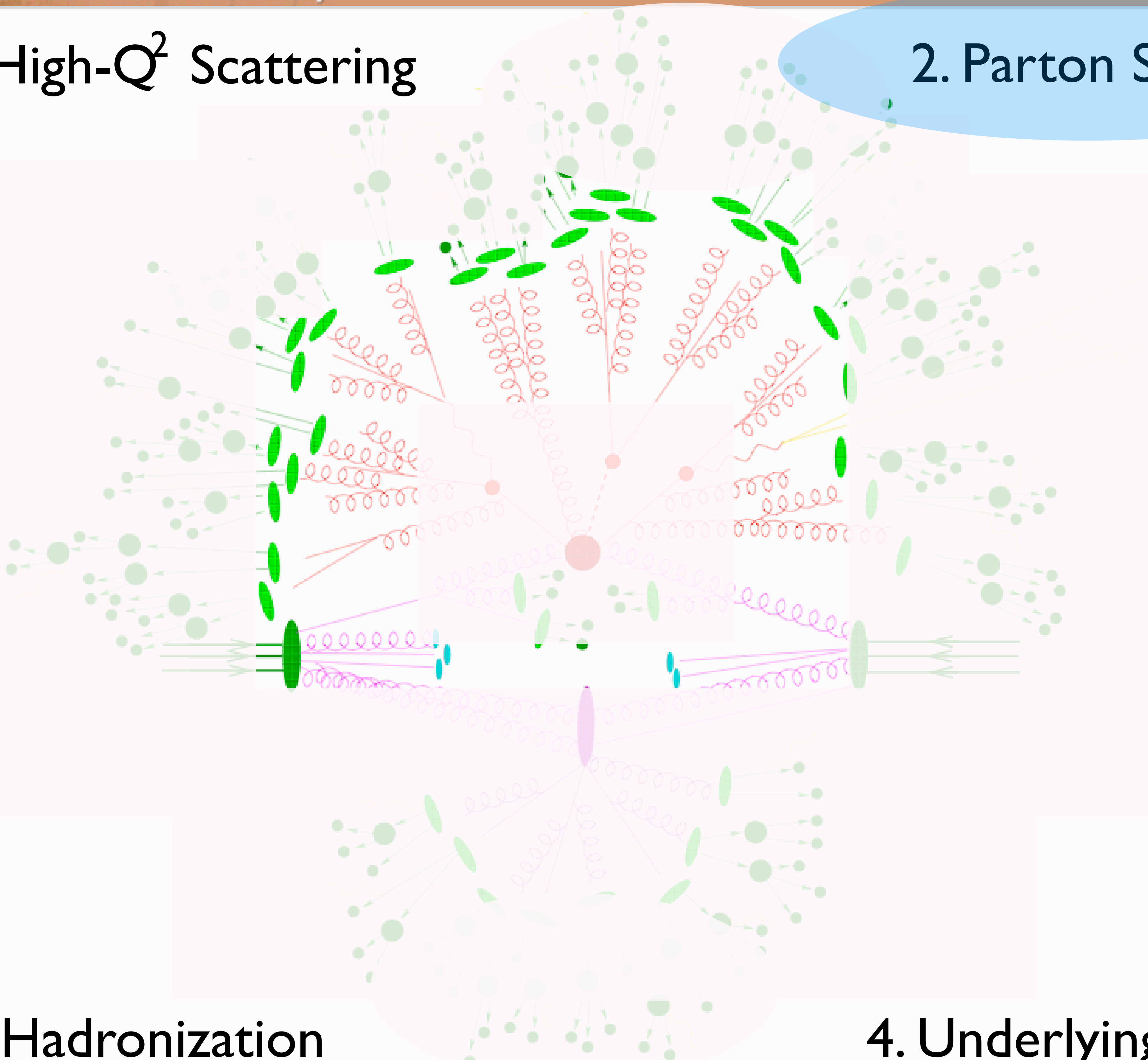
☞ it can be systematically improved

3. Hadronization

4. Underlying Event

1. High-Q^2 Scattering

2. Parton Shower

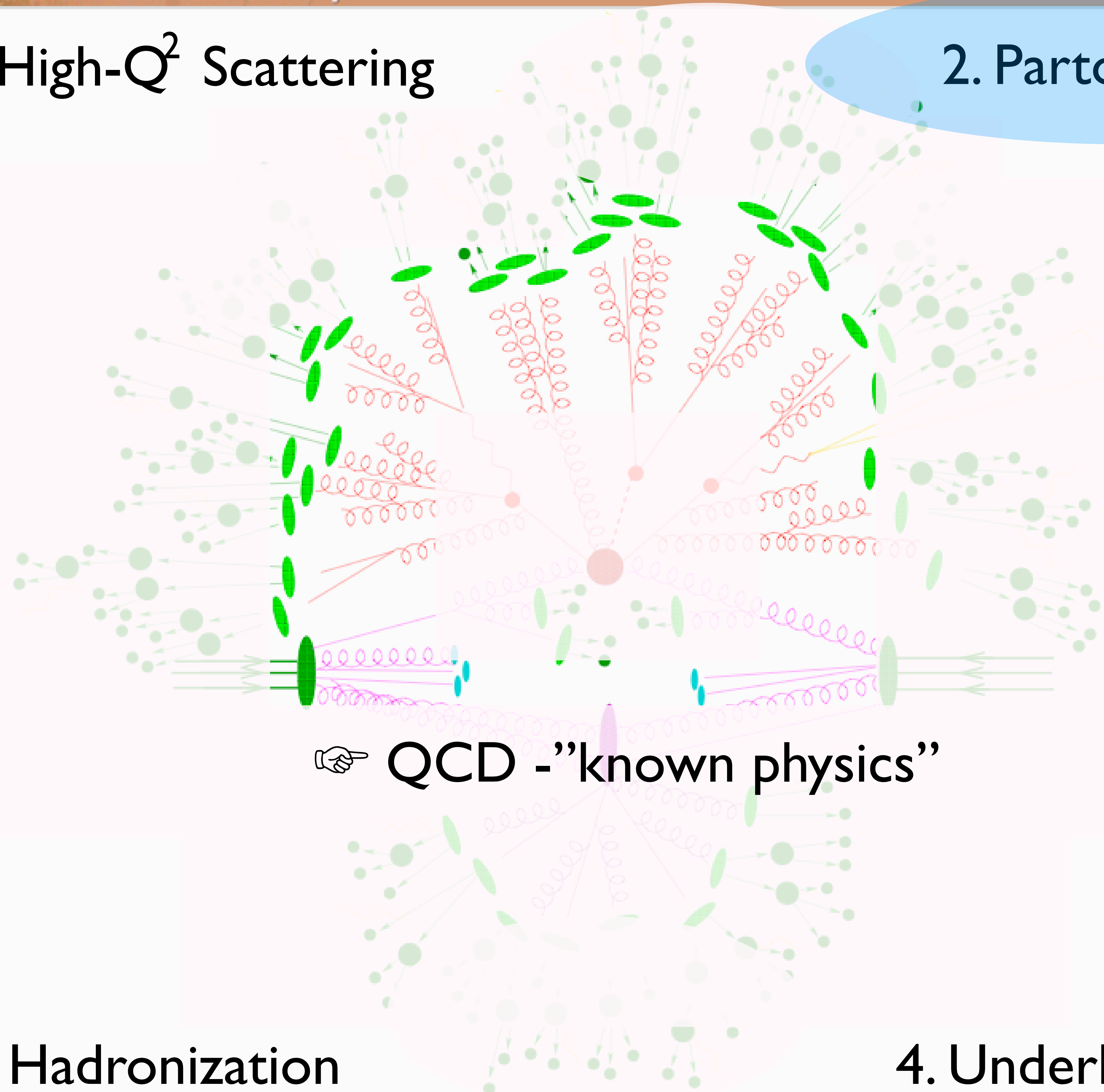


3. Hadronization

4. Underlying Event

1. High-Q^2 Scattering

2. Parton Shower



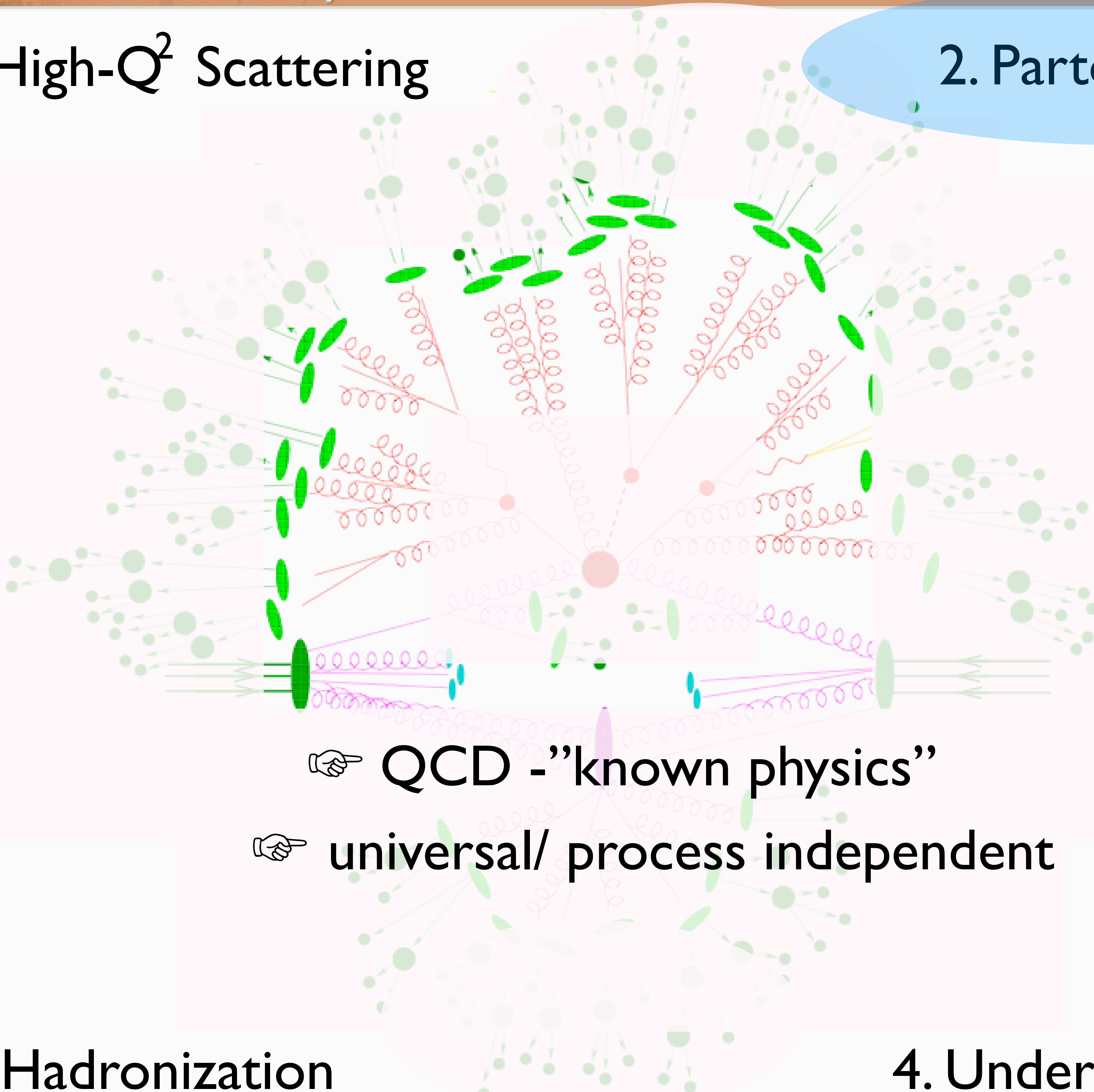
☞ QCD -"known physics"

3. Hadronization

4. Underlying Event

1. High-Q^2 Scattering

2. Parton Shower



☞ QCD -"known physics"

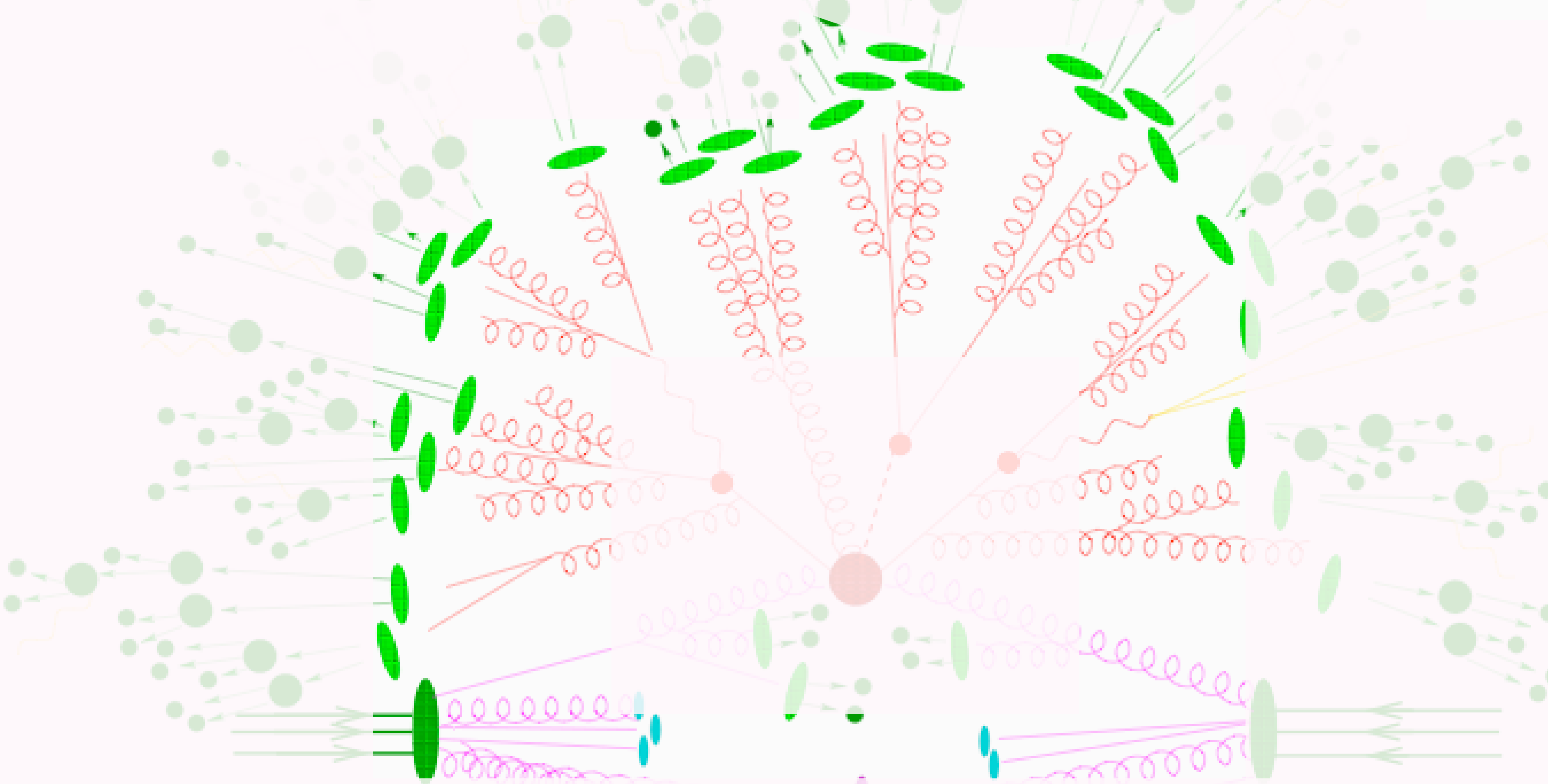
☞ universal/ process independent

3. Hadronization

4. Underlying Event

1. High-Q^2 Scattering

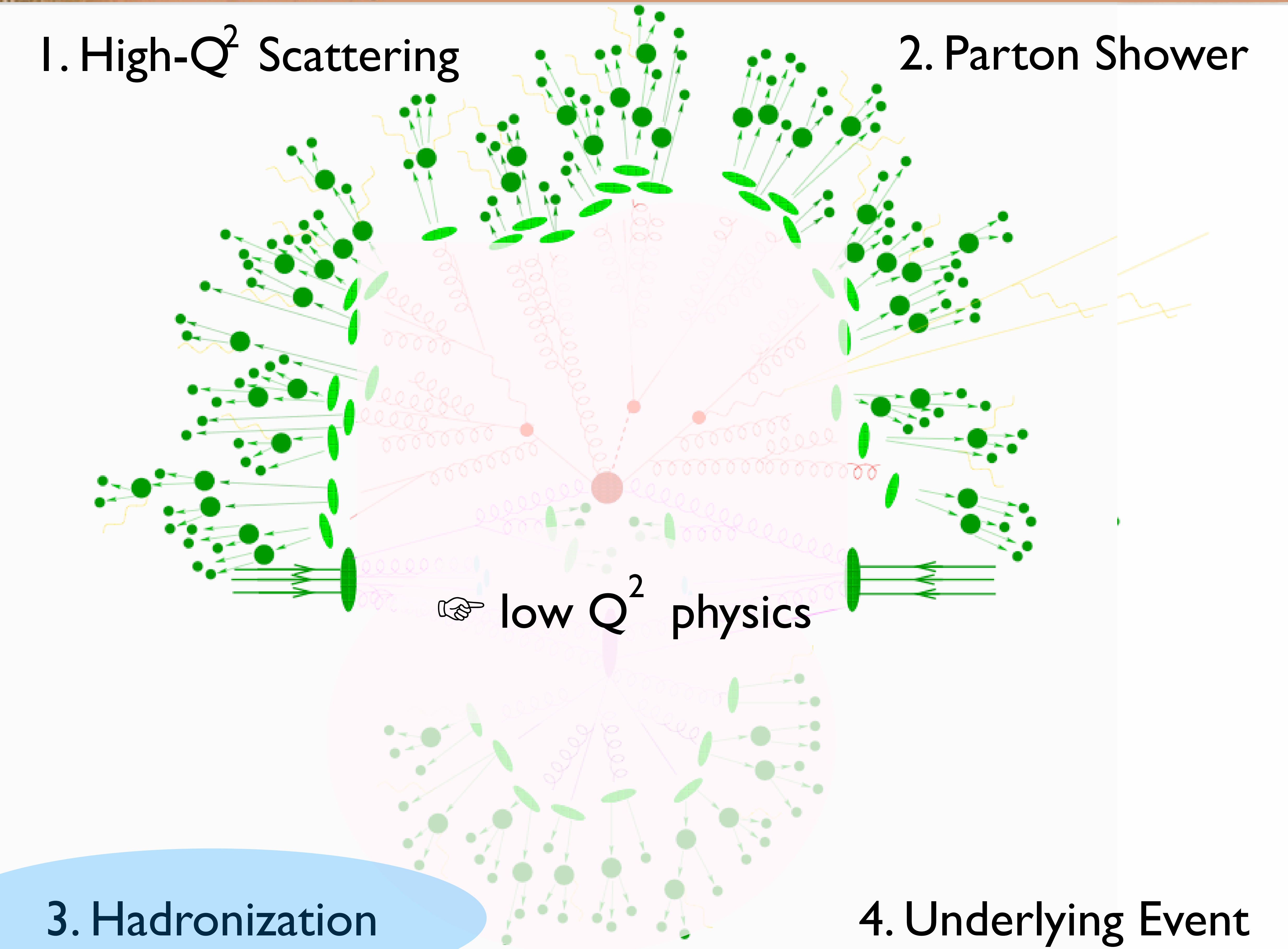
2. Parton Shower

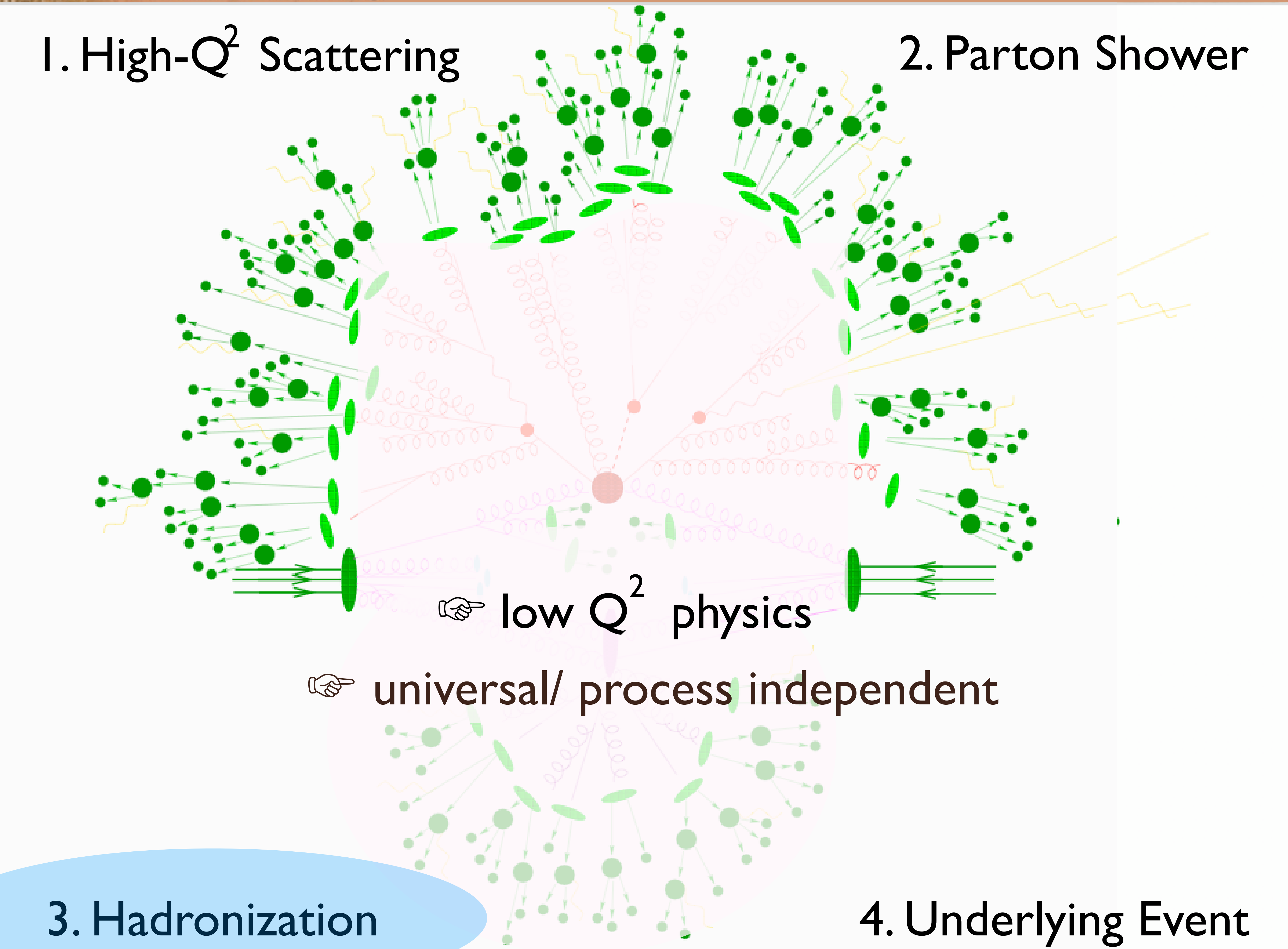


- ☞ QCD -”known physics”
- ☞ universal/ process independent
- ☞ first principles description

3. Hadronization

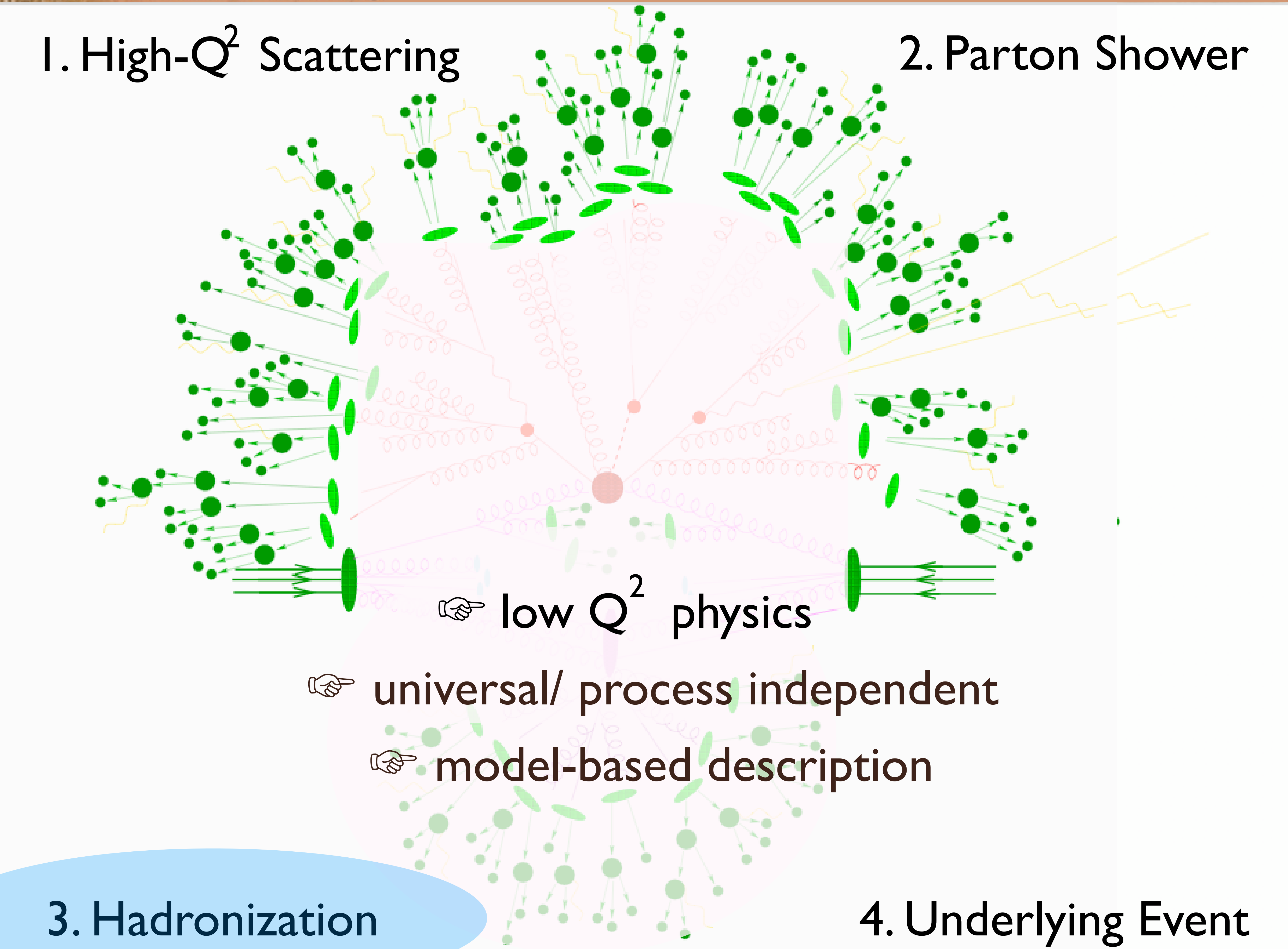
4. Underlying Event


1. High-Q^2 Scattering
2. Parton Shower
☞ low Q^2 physics
3. Hadronization
4. Underlying Event



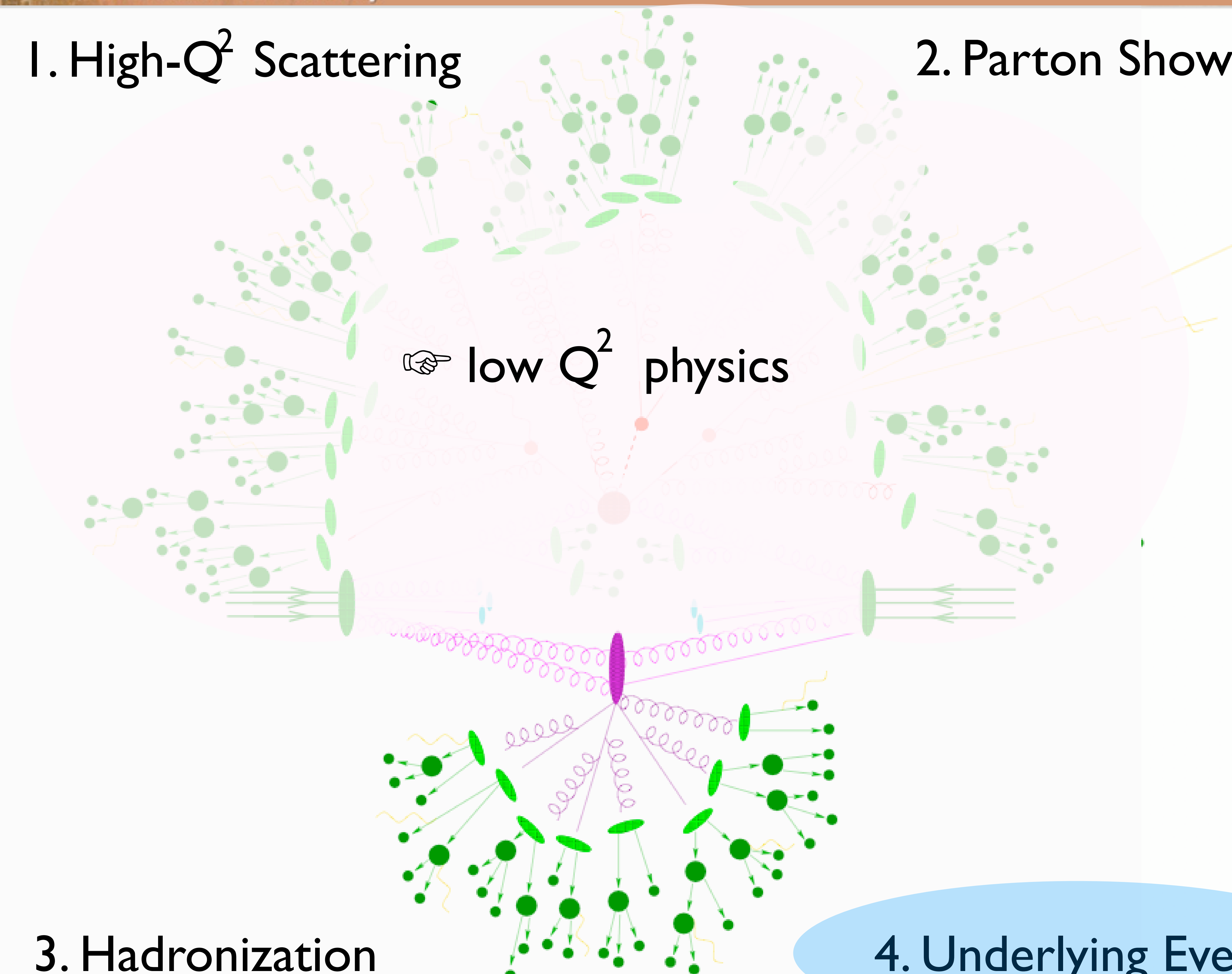
☞ low Q^2 physics

☞ universal/ process independent


1. High-Q² Scattering
2. Parton Shower
☞ low Q² physics
☞ universal/ process independent
☞ model-based description
3. Hadronization
4. Underlying Event

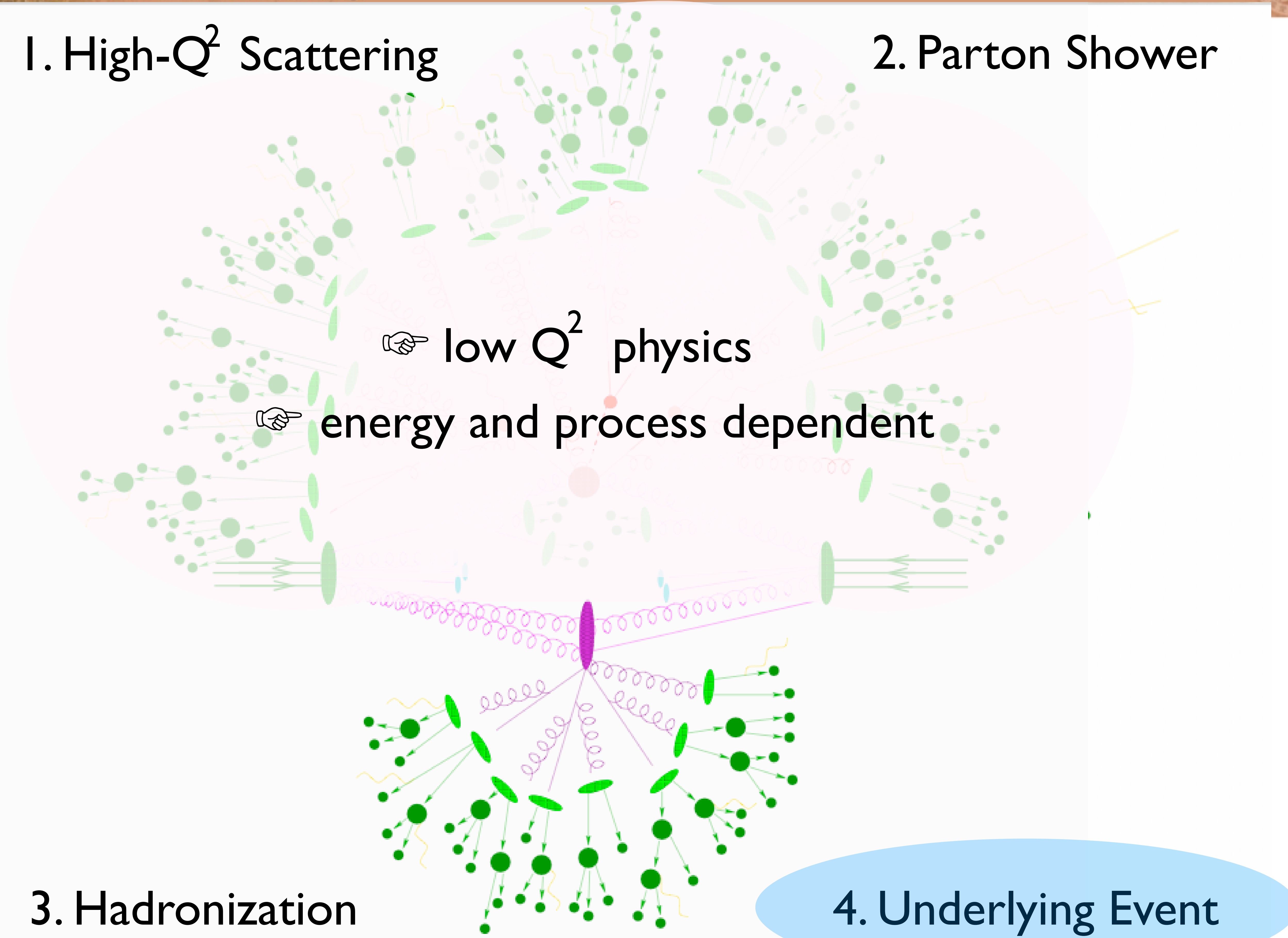
1. High-Q^2 Scattering

2. Parton Shower



3. Hadronization

4. Underlying Event

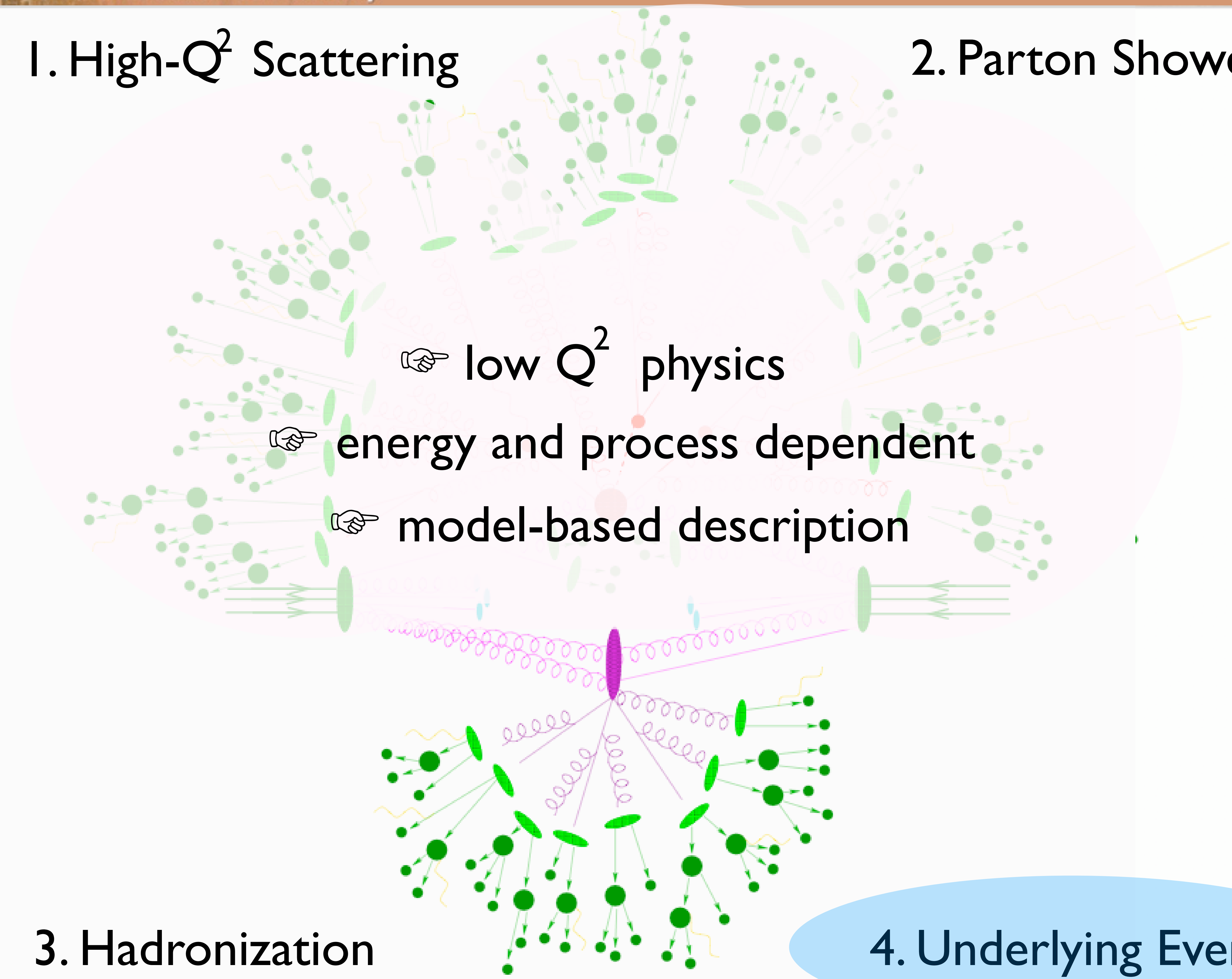


☞ low Q^2 physics

☞ energy and process dependent

1. High-Q^2 Scattering

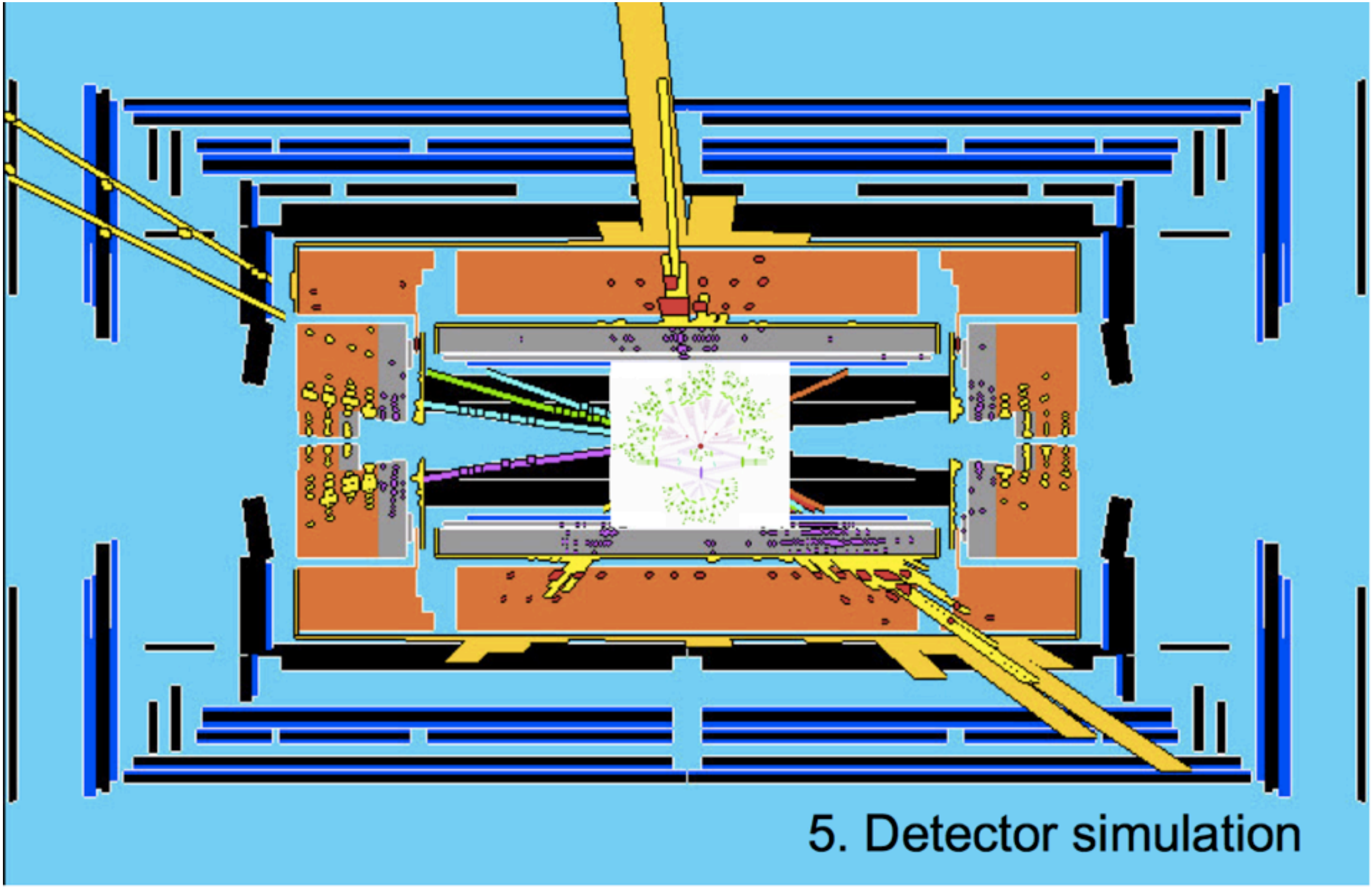
2. Parton Shower



- ☞ low Q^2 physics
- ☞ energy and process dependent
- ☞ model-based description

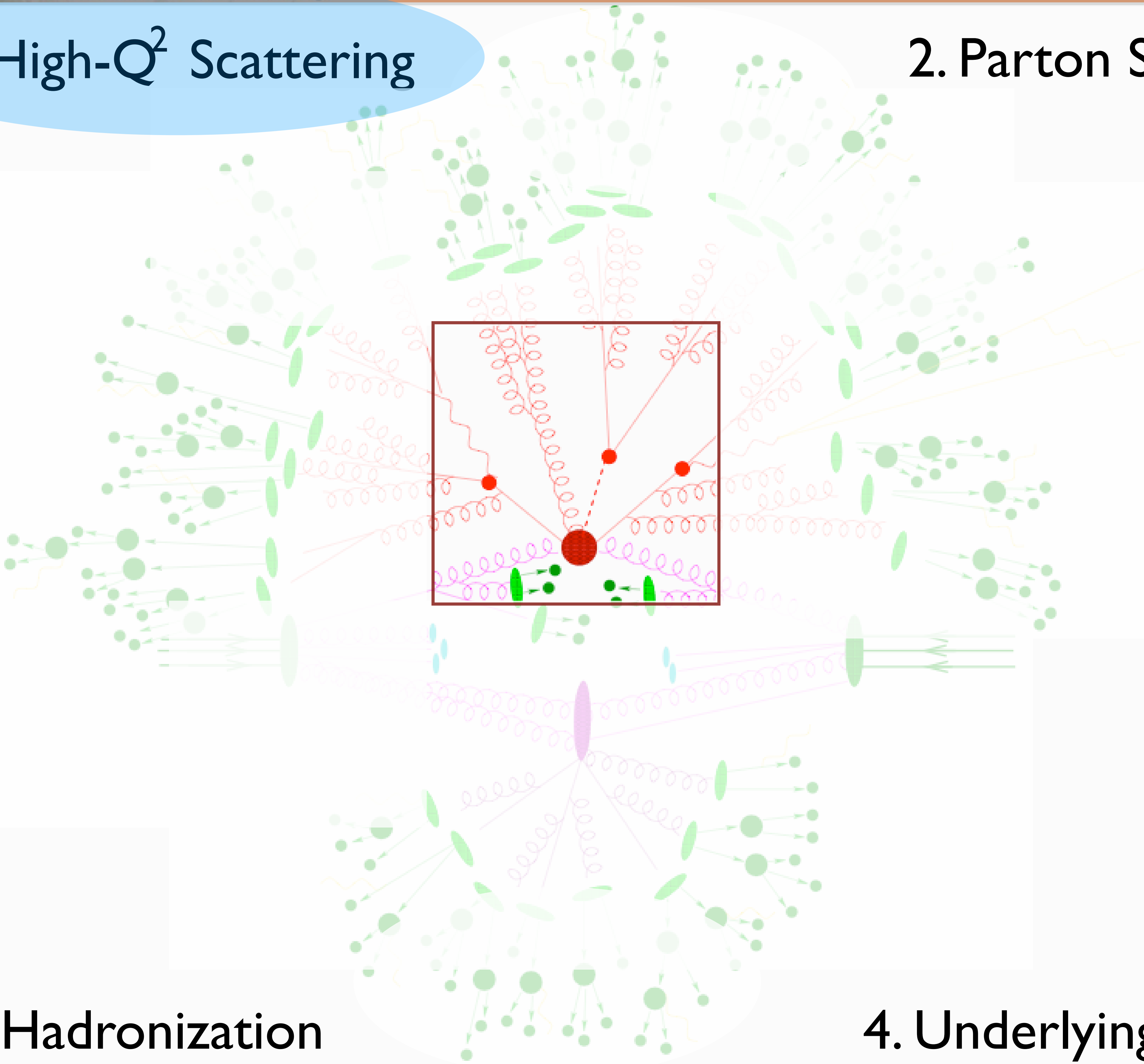
3. Hadronization

4. Underlying Event


5. Detector simulation

1. High-Q^2 Scattering

2. Parton Shower



3. Hadronization

4. Underlying Event

No.	Subprocess
Hard QCD processes:	
11	$f_i f_j \to f_i f_j$
12	$f_i \bar{f}_i \to f_k \bar{f}_k$
13	$f_i \bar{f}_i \to gg$
28	$f_i g \to f_i g$
53	$gg \to f_k \bar{f}_k$
68	$gg \to gg$
Soft QCD processes:	
91	elastic scattering
92	single diffraction (XB)
93	single diffraction (AX)
94	double diffraction
95	low-$p_\perp$ production
Prompt photons:	
14	$f_i \bar{f}_i \to g\gamma$
18	$f_i \bar{f}_i \to \gamma\gamma$
29	$f_i g \to f_i \gamma$
114	$gg \to \gamma\gamma$
115	$gg \to g\gamma$
Open heavy flavour: (also fourth generation)	
81	$f_i \bar{f}_i \to Q_k \bar{Q}_k$
82	$gg \to Q_k \bar{Q}_k$
83	$q_i f_j \to Q_k f_l$
84	$g\gamma \to Q_k \bar{Q}_k$
85	$\gamma\gamma \to F_k \bar{F}_k$

No.	Subprocess
Closed heavy flavour:	
86	$gg \to J/\psi g$
87	$gg \to \chi_{0c} g$
88	$gg \to \chi_{1c} g$
89	$gg \to \chi_{2c} g$
104	$gg \to \chi_{0c}$
105	$gg \to \chi_{2c}$
106	$gg \to J/\psi \gamma$
107	$g\gamma \to J/\psi g$
108	$\gamma\gamma \to J/\psi \gamma$
421	$gg \to c\bar{c}[^3S_1^{(1)}]g$
422	$gg \to c\bar{c}[^3S_1^{(8)}]g$
423	$gg \to c\bar{c}[^1S_0^{(8)}]g$
424	$gg \to c\bar{c}[^3P_J^{(8)}]g$
425	$gq \to q\,c\bar{c}[^3S_1^{(8)}]$
426	$gq \to q\,c\bar{c}[^1S_0^{(8)}]$
427	$gq \to q\,c\bar{c}[^3P_J^{(8)}]$
428	$q\bar{q} \to g\,c\bar{c}[^3S_1^{(8)}]$
429	$q\bar{q} \to g\,c\bar{c}[^1S_0^{(8)}]$
430	$q\bar{q} \to g\,c\bar{c}[^3P_J^{(8)}]$
431	$gg \to c\bar{c}[^3P_0^{(1)}]g$
432	$gg \to c\bar{c}[^3P_1^{(1)}]g$
433	$gg \to c\bar{c}[^3P_2^{(1)}]g$
434	$gq \to q\,c\bar{c}[^3P_0^{(1)}]$

No.	Subprocess
435	$gq \to q\,c\bar{c}[^3P_1^{(1)}]$
436	$gq \to q\,c\bar{c}[^3P_2^{(1)}]$
437	$q\bar{q} \to g\,c\bar{c}[^3P_0^{(1)}]$
438	$q\bar{q} \to g\,c\bar{c}[^3P_1^{(1)}]$
439	$q\bar{q} \to g\,c\bar{c}[^3P_2^{(1)}]$
461	$gg \to b\bar{b}[^3S_1^{(1)}]g$
462	$gg \to b\bar{b}[^3S_1^{(8)}]g$
463	$gg \to b\bar{b}[^1S_0^{(8)}]g$
464	$gg \to b\bar{b}[^3P_J^{(8)}]g$
465	$gq \to q\,b\bar{b}[^3S_1^{(8)}]$
466	$gq \to q\,b\bar{b}[^1S_0^{(8)}]$
467	$gq \to q\,b\bar{b}[^3P_J^{(8)}]$
468	$q\bar{q} \to g\,b\bar{b}[^3S_1^{(8)}]$
469	$q\bar{q} \to g\,b\bar{b}[^1S_0^{(8)}]$
470	$q\bar{q} \to g\,b\bar{b}[^3P_J^{(8)}]$
471	$gg \to b\bar{b}[^3P_0^{(1)}]g$
472	$gg \to b\bar{b}[^3P_1^{(1)}]g$
473	$gg \to b\bar{b}[^3P_2^{(1)}]g$
474	$gq \to q\,b\bar{b}[^3P_0^{(1)}]$
475	$gq \to q\,b\bar{b}[^3P_1^{(1)}]$
476	$gq \to q\,b\bar{b}[^3P_2^{(1)}]$
477	$q\bar{q} \to g\,b\bar{b}[^3P_0^{(1)}]$
478	$q\bar{q} \to g\,b\bar{b}[^3P_1^{(1)}]$
479	$q\bar{q} \to g\,b\bar{b}[^3P_2^{(1)}]$

No.	Subprocess
Deeply Inel. Scatt.:	
10	$f_i f_j \to f_k f_l$
99	$\gamma^* q \to q$
Photon-induced:	
33	$f_i \gamma \to f_i g$
34	$f_i \gamma \to f_i \gamma$
54	$g\gamma \to f_k \bar{f}_k$
58	$\gamma\gamma \to f_k \bar{f}_k$
131	$f_i \gamma^*_T \to f_i g$
132	$f_i \gamma^*_L \to f_i g$
133	$f_i \gamma^*_T \to f_i \gamma$
134	$f_i \gamma^*_L \to f_i \gamma$
135	$g\gamma^*_T \to f_i \bar{f}_i$
136	$g\gamma^*_L \to f_i \bar{f}_i$
137	$\gamma^*_T \gamma^*_T \to f_i \bar{f}_i$
138	$\gamma^*_T \gamma^*_L \to f_i \bar{f}_i$
139	$\gamma^*_L \gamma^*_T \to f_i \bar{f}_i$
140	$\gamma^*_L \gamma^*_L \to f_i \bar{f}_i$
80	$q_i \gamma \to q_k \pi^\pm$
W/Z production:	
1	$f_i \bar{f}_i \to \gamma^*/Z^0$
2	$f_i \bar{f}_j \to W^\pm$
22	$f_i \bar{f}_i \to Z^0 Z^0$
23	$f_i \bar{f}_j \to Z^0 W^\pm$
25	$f_i \bar{f}_i \to W^+ W^-$
15	$f_i \bar{f}_i \to g Z^0$
16	$f_i \bar{f}_j \to g W^\pm$
30	$f_i g \to f_i Z^0$
31	$f_i g \to f_k W^\pm$

No.	Subprocess
19	$f_i \bar{f}_i \to \gamma Z^0$
20	$f_i \bar{f}_j \to \gamma W^\pm$
35	$f_i \gamma \to f_i Z^0$
36	$f_i \gamma \to f_k W^\pm$
69	$\gamma\gamma \to W^+ W^-$
70	$\gamma W^\pm \to Z^0 W^\pm$
Light SM Higgs:	
3	$f_i \bar{f}_i \to h^0$
24	$f_i \bar{f}_i \to Z^0 h^0$
26	$f_i \bar{f}_j \to W^\pm h^0$
32	$f_i g \to f_i h^0$
102	$gg \to h^0$
103	$\gamma\gamma \to h^0$
110	$f_i \bar{f}_i \to \gamma h^0$
111	$f_i \bar{f}_i \to g h^0$
112	$f_i g \to f_i h^0$
113	$gg \to g h^0$
121	$gg \to Q_k \bar{Q}_k h^0$
122	$q_i \bar{q}_i \to Q_k \bar{Q}_k h^0$
123	$f_i f_j \to f_i f_j h^0$
124	$f_i f_j \to f_k f_l h^0$
Heavy SM Higgs:	
5	$Z^0 Z^0 \to h^0$
8	$W^+ W^- \to h^0$
71	$Z^0_L Z^0_L \to Z^0_L Z^0_L$
72	$Z^0_L Z^0_L \to W^+_L W^-_L$
73	$Z^0_L W^\pm_L \to Z^0_L W^\pm_L$
76	$W^+_L W^-_L \to Z^0_L Z^0_L$
77	$W^\pm_L W^\pm_L \to W^\pm_L W^\pm_L$

No.	Subprocess
BSM Neutral Higgs:	
151	$f_i \bar{f}_i \to H^0$
152	$gg \to H^0$
153	$\gamma\gamma \to H^0$
171	$f_i \bar{f}_i \to Z^0 H^0$
172	$f_i \bar{f}_j \to W^\pm H^0$
173	$f_i f_j \to f_i f_j H^0$
174	$f_i f_j \to f_k f_l H^0$
181	$gg \to Q_k \bar{Q}_k H^0$
182	$q_i \bar{q}_i \to Q_k \bar{Q}_k H^0$
183	$f_i \bar{f}_i \to g H^0$
184	$f_i g \to f_i H^0$
185	$gg \to g H^0$
156	$f_i \bar{f}_i \to A^0$
157	$gg \to A^0$
158	$\gamma\gamma \to A^0$
176	$f_i \bar{f}_i \to Z^0 A^0$
177	$f_i \bar{f}_j \to W^\pm A^0$
178	$f_i f_j \to f_i f_j A^0$
179	$f_i f_j \to f_k f_l A^0$
186	$gg \to Q_k \bar{Q}_k A^0$
187	$q_i \bar{q}_i \to Q_k \bar{Q}_k A^0$
188	$f_i \bar{f}_i \to g A^0$
189	$f_i g \to f_i A^0$
190	$gg \to g A^0$

No.	Subprocess
Charged Higgs:	
143	$f_i \bar{f}_j \to H^+$
161	$f_i g \to f_k H^+$
401	$gg \to \bar{t}bH^+$
402	$q\bar{q} \to \bar{t}bH^+$
Higgs pairs:	
297	$f_i \bar{f}_j \to H^\pm h^0$
298	$f_i \bar{f}_j \to H^\pm H^0$
299	$f_i \bar{f}_i \to A^0 h^0$
300	$f_i \bar{f}_i \to A^0 H^0$
301	$f_i \bar{f}_i \to H^+ H^-$
New gauge bosons:	
141	$f_i \bar{f}_i \to \gamma/Z^0/Z'^0$
142	$f_i \bar{f}_j \to W'^+$
144	$f_i \bar{f}_j \to R$
Leptoquarks:	
145	$q_i \ell_j \to L_Q$
162	$qg \to \ell L_Q$
163	$gg \to L_Q \bar{L}_Q$
164	$q_i \bar{q}_i \to L_Q \bar{L}_Q$

No.	Subprocess
Technicolor:	
149	$gg \to \eta_{tc}$
191	$f_i \bar{f}_i \to \rho^0_{tc}$
192	$f_i \bar{f}_j \to \rho^+_{tc}$
193	$f_i \bar{f}_i \to \omega^0_{tc}$
194	$f_i \bar{f}_i \to f_k \bar{f}_k$
195	$f_i \bar{f}_j \to f_k \bar{f}_l$
361	$f_i \bar{f}_i \to W^+_L W^-_L$
362	$f_i \bar{f}_i \to W^\pm_L \pi^\mp_{tc}$
363	$f_i \bar{f}_i \to \pi^+_{tc} \pi^-_{tc}$
364	$f_i \bar{f}_i \to \gamma \pi^0_{tc}$
365	$f_i \bar{f}_i \to \gamma \pi'^0_{tc}$
366	$f_i \bar{f}_i \to Z^0 \pi^0_{tc}$
367	$f_i \bar{f}_i \to Z^0 \pi'^0_{tc}$
368	$f_i \bar{f}_i \to W^\pm \pi^\mp_{tc}$
370	$f_i \bar{f}_j \to W^\pm_L Z^0_L$
371	$f_i \bar{f}_j \to W^\pm_L \pi^0_{tc}$
372	$f_i \bar{f}_j \to \pi^\pm_{tc} Z^0_L$
373	$f_i \bar{f}_j \to \pi^\pm_{tc} \pi^0_{tc}$
374	$f_i \bar{f}_j \to \gamma \pi^\pm_{tc}$
375	$f_i \bar{f}_j \to Z^0 \pi^\pm_{tc}$
376	$f_i \bar{f}_j \to W^\pm \pi^0_{tc}$
377	$f_i \bar{f}_j \to W^\pm \pi'^0_{tc}$
381	$q_i q_j \to q_i q_j$
382	$q_i \bar{q}_i \to q_k \bar{q}_k$
383	$q_i \bar{q}_i \to gg$
384	$f_i g \to f_i g$
385	$gg \to q_k \bar{q}_k$
386	$gg \to gg$
387	$f_i \bar{f}_i \to Q_k \bar{Q}_k$
388	$gg \to Q_k \bar{Q}_k$

No.	Subprocess
Compositeness:	
146	$e\gamma \to e^*$
147	$dg \to d^*$
148	$ug \to u^*$
167	$q_i q_j \to d^* q_k$
168	$q_i q_j \to u^* q_k$
169	$q_i \bar{q}_i \to e^\pm e^{*\mp}$
165	$f_i \bar{f}_i (\to \gamma^*/Z^0) \to f_k \bar{f}_k$
166	$f_i \bar{f}_j (\to W^\pm) \to f_k \bar{f}_l$
Left-right symmetry:	
341	$\ell_i \ell_j \to H^{\pm\pm}_L$
342	$\ell_i \ell_j \to H^{\pm\pm}_R$
343	$\ell^\pm_i \gamma \to H^{\pm\pm}_L e^\mp$
344	$\ell^\pm_i \gamma \to H^{\pm\pm}_R e^\mp$
345	$\ell^\pm_i \gamma \to H^{\pm\pm}_L \mu^\mp$
346	$\ell^\pm_i \gamma \to H^{\pm\pm}_R \mu^\mp$
347	$\ell^\pm_i \gamma \to H^{\pm\pm}_L \tau^\mp$
348	$\ell^\pm_i \gamma \to H^{\pm\pm}_R \tau^\mp$
349	$f_i \bar{f}_i \to H^{++}_L H^{--}_L$
350	$f_i \bar{f}_i \to H^{++}_R H^{--}_R$
351	$f_i f_j \to f_k f_l H^{\pm\pm}_L$
352	$f_i f_j \to f_k f_l H^{\pm\pm}_R$
353	$f_i \bar{f}_i \to Z^0_R$
354	$f_i \bar{f}_j \to W^\pm_R$
Extra Dimensions:	
391	$f\bar{f} \to G^*$
392	$gg \to G^*$
393	$q\bar{q} \to gG^*$
394	$qg \to qG^*$
395	$gg \to gG^*$

No.	Subprocess
SUSY:	
201	$f_i \bar{f}_i \to \tilde{e}_L \tilde{e}_L^*$
202	$f_i \bar{f}_i \to \tilde{e}_R \tilde{e}_R^*$
203	$f_i \bar{f}_i \to \tilde{e}_L \tilde{e}_R^* +$
204	$f_i \bar{f}_i \to \tilde{\mu}_L \tilde{\mu}_L^*$
205	$f_i \bar{f}_i \to \tilde{\mu}_R \tilde{\mu}_R^*$
206	$f_i \bar{f}_i \to \tilde{\mu}_L \tilde{\mu}_R^* +$
207	$f_i \bar{f}_i \to \tilde{\tau}_1 \tilde{\tau}_1^*$
208	$f_i \bar{f}_i \to \tilde{\tau}_2 \tilde{\tau}_2^*$
209	$f_i \bar{f}_i \to \tilde{\tau}_1 \tilde{\tau}_2^* +$
210	$f_i \bar{f}_j \to \tilde{\ell}_L \tilde{\nu}_\ell^* +$
211	$f_i \bar{f}_j \to \tilde{\tau}_1 \tilde{\nu}_\tau^* +$
212	$f_i \bar{f}_j \to \tilde{\tau}_2 \tilde{\nu}_\tau^* +$
213	$f_i \bar{f}_i \to \tilde{\nu}_\ell \tilde{\nu}_\ell^*$
214	$f_i \bar{f}_i \to \tilde{\nu}_\tau \tilde{\nu}_\tau^*$
216	$f_i \bar{f}_i \to \tilde{\chi}_1 \tilde{\chi}_1$
217	$f_i \bar{f}_i \to \tilde{\chi}_2 \tilde{\chi}_2$
218	$f_i \bar{f}_i \to \tilde{\chi}_3 \tilde{\chi}_3$
219	$f_i \bar{f}_i \to \tilde{\chi}_4 \tilde{\chi}_4$
220	$f_i \bar{f}_i \to \tilde{\chi}_1 \tilde{\chi}_2$
221	$f_i \bar{f}_i \to \tilde{\chi}_1 \tilde{\chi}_3$
222	$f_i \bar{f}_i \to \tilde{\chi}_1 \tilde{\chi}_4$
223	$f_i \bar{f}_i \to \tilde{\chi}_2 \tilde{\chi}_3$
224	$f_i \bar{f}_i \to \tilde{\chi}_2 \tilde{\chi}_4$
225	$f_i \bar{f}_i \to \tilde{\chi}_3 \tilde{\chi}_4$
226	$f_i \bar{f}_i \to \tilde{\chi}^\pm_1 \tilde{\chi}^\mp_1$
227	$f_i \bar{f}_i \to \tilde{\chi}^\pm_2 \tilde{\chi}^\mp_2$
228	$f_i \bar{f}_i \to \tilde{\chi}^\pm_1 \tilde{\chi}^\mp_2$
229	$f_i \bar{f}_j \to \tilde{\chi}_1 \tilde{\chi}^\pm_1$

No.	Subprocess
230	$f_i \bar{f}_j \to \tilde{\chi}_2 \tilde{\chi}^\pm_1$
231	$f_i \bar{f}_j \to \tilde{\chi}_3 \tilde{\chi}^\pm_1$
232	$f_i \bar{f}_j \to \tilde{\chi}_4 \tilde{\chi}^\pm_1$
233	$f_i \bar{f}_j \to \tilde{\chi}_1 \tilde{\chi}^\pm_2$
234	$f_i \bar{f}_j \to \tilde{\chi}_2 \tilde{\chi}^\pm_2$
235	$f_i \bar{f}_j \to \tilde{\chi}_3 \tilde{\chi}^\pm_2$
236	$f_i \bar{f}_j \to \tilde{\chi}_4 \tilde{\chi}^\pm_2$
237	$f_i \bar{f}_i \to \tilde{g} \tilde{\chi}_1$
238	$f_i \bar{f}_i \to \tilde{g} \tilde{\chi}_2$
239	$f_i \bar{f}_i \to \tilde{g} \tilde{\chi}_3$
240	$f_i \bar{f}_i \to \tilde{g} \tilde{\chi}_4$
241	$f_i \bar{f}_j \to \tilde{g} \tilde{\chi}^\pm_1$
242	$f_i \bar{f}_j \to \tilde{g} \tilde{\chi}^\pm_2$
243	$f_i \bar{f}_i \to \tilde{g} \tilde{g}$
244	$gg \to \tilde{g} \tilde{g}$
246	$f_i g \to \tilde{q}_{iL} \tilde{\chi}_1$
247	$f_i g \to \tilde{q}_{iR} \tilde{\chi}_1$
248	$f_i g \to \tilde{q}_{iL} \tilde{\chi}_2$
249	$f_i g \to \tilde{q}_{iR} \tilde{\chi}_2$
250	$f_i g \to \tilde{q}_{iL} \tilde{\chi}_3$
251	$f_i g \to \tilde{q}_{iR} \tilde{\chi}_3$
252	$f_i g \to \tilde{q}_{iL} \tilde{\chi}_4$
253	$f_i g \to \tilde{q}_{iR} \tilde{\chi}_4$
254	$f_i g \to \tilde{q}_{jL} \tilde{\chi}^\pm_1$
256	$f_i g \to \tilde{q}_{jL} \tilde{\chi}^\pm_2$
258	$f_i g \to \tilde{q}_{iL} \tilde{g}$
259	$f_i g \to \tilde{q}_{iR} \tilde{g}$
261	$f_i \bar{f}_i \to \tilde{t}_1 \tilde{t}_1^*$
262	$f_i \bar{f}_i \to \tilde{t}_2 \tilde{t}_2^*$

No.	Subprocess
263	$f_i \bar{f}_i \to \tilde{t}_1 \tilde{t}_2^* +$
264	$gg \to \tilde{t}_1 \tilde{t}_1^*$
265	$gg \to \tilde{t}_2 \tilde{t}_2^*$
271	$f_i f_j \to \tilde{q}_{iL} \tilde{q}_{jL}$
272	$f_i f_j \to \tilde{q}_{iR} \tilde{q}_{jR}$
273	$f_i f_j \to \tilde{q}_{iL} \tilde{q}_{jR} +$
274	$f_i \bar{f}_j \to \tilde{q}_{iL} \tilde{q}_{jL}^*$
275	$f_i \bar{f}_j \to \tilde{q}_{iR} \tilde{q}_{jR}^*$
276	$f_i \bar{f}_j \to \tilde{q}_{iL} \tilde{q}_{jR}^* +$
277	$f_i \bar{f}_i \to \tilde{q}_{jL} \tilde{q}_{jL}^*$
278	$f_i \bar{f}_i \to \tilde{q}_{jR} \tilde{q}_{jR}^*$
279	$gg \to \tilde{q}_{iL} \tilde{q}_{iL}^*$
280	$gg \to \tilde{q}_{iR} \tilde{q}_{iR}^*$
281	$bq_i \to \tilde{b}_1 \tilde{q}_{iL}$
282	$bq_i \to \tilde{b}_2 \tilde{q}_{iR}$
283	$bq_i \to \tilde{b}_1 \tilde{q}_{iR} + \tilde{b}_2 \tilde{q}_{iL}$
284	$b\bar{q}_i \to \tilde{b}_1 \tilde{q}_{iL}^*$
285	$b\bar{q}_i \to \tilde{b}_2 \tilde{q}_{iR}^*$
286	$b\bar{q}_i \to \tilde{b}_1 \tilde{q}_{iR}^* + \tilde{b}_2 \tilde{q}_{iL}^*$
287	$f_i \bar{f}_i \to \tilde{b}_1 \tilde{b}_1^*$
288	$f_i \bar{f}_i \to \tilde{b}_2 \tilde{b}_2^*$
289	$gg \to \tilde{b}_1 \tilde{b}_1^*$
290	$gg \to \tilde{b}_2 \tilde{b}_2^*$
291	$bb \to \tilde{b}_1 \tilde{b}_1$
292	$bb \to \tilde{b}_2 \tilde{b}_2$
293	$bb \to \tilde{b}_1 \tilde{b}_2$
294	$bg \to \tilde{b}_1 \tilde{g}$
295	$bg \to \tilde{b}_2 \tilde{g}$
296	$b\bar{b} \to \tilde{b}_1 \tilde{b}_2^* +$

List of processes implemented in Pythia (by hand!)

Automated Matrix Element Generators

Automated Matrix Element Generators

- High-Q2 scattering processes: In principle infinite number of processes for innumerable number of models

Automated Matrix Element Generators

- High-Q2 scattering processes: In principle infinite number of processes for innumerable number of models
- Implementation by hand time-consuming, labor intensive and error prone

Automated Matrix Element Generators

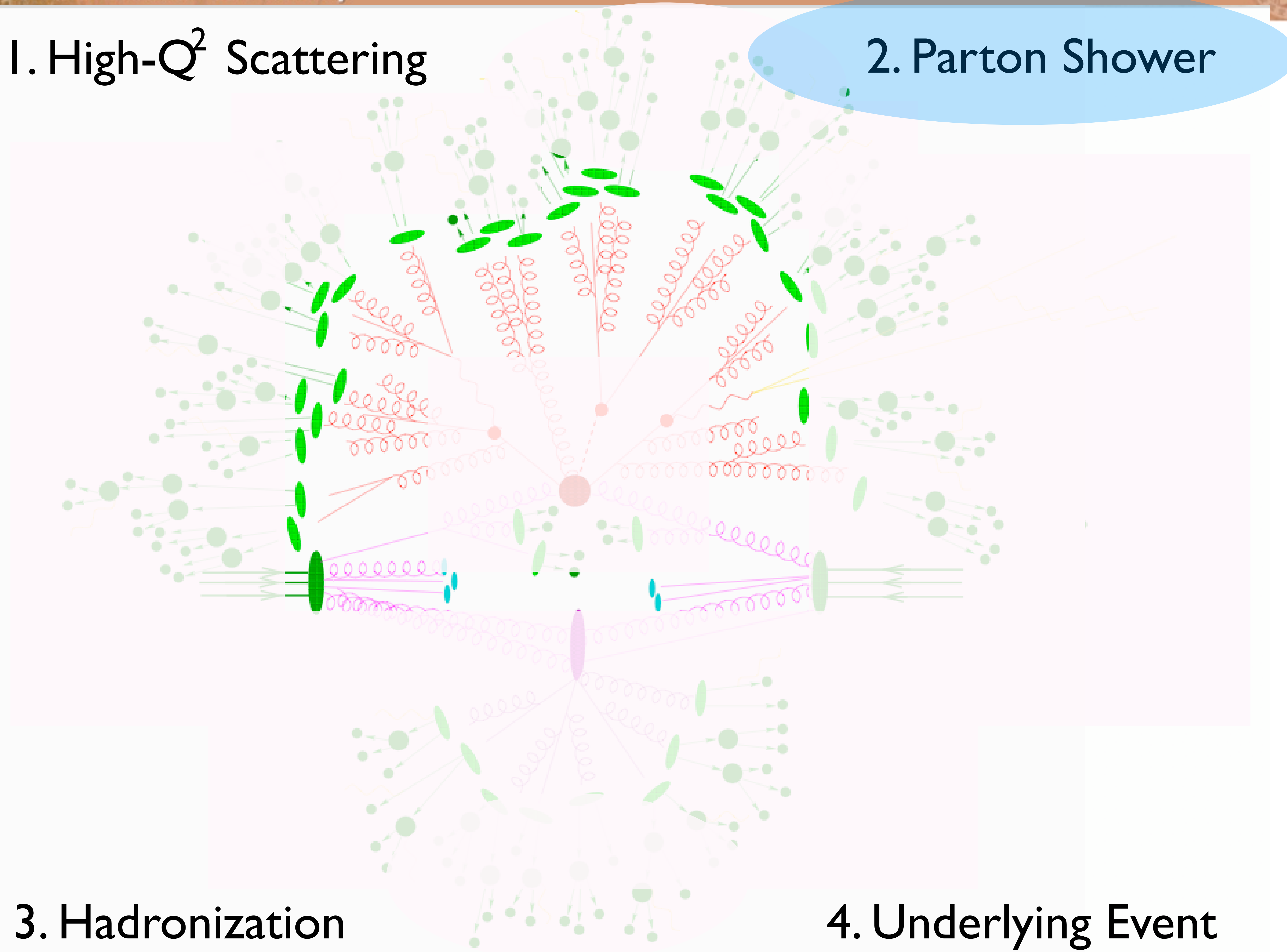
- High-Q2 scattering processes: In principle infinite number of processes for innumerable number of models
- Implementation by hand time-consuming, labor intensive and error prone
- Instead: Automated matrix element generators
  - ➡ Use Feynman rules to build diagrams

Automated Matrix Element Generators

- High-Q2 scattering processes: In principle infinite number of processes for innumerable number of models
- Implementation by hand time-consuming, labor intensive and error prone
- Instead: Automated matrix element generators
  - ➡ Use Feynman rules to build diagrams
- Given files defining the model content: particles, parameters and interactions, allows to generate any process for a given model!

Automated Matrix Element Generators

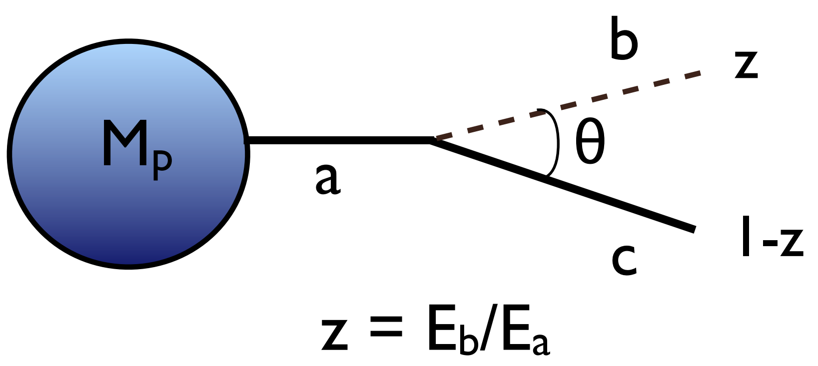
- Automatic matrix element generators:
  - ➡ CalcHep / CompHep
  - ➡ MadGraph
  - ➡ AMEGIC++ (Sherpa)
  - ➡ Whizard
- Standard Model only, with fast matrix elements for high parton multiplicity final states:
  - ➡ AlpGen
  - ➡ HELAC
  - ➡ COMIX (Sherpa)


1. High-Q^2 Scattering
2. Parton Shower
3. Hadronization
4. Underlying Event

Parton Shower basics

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

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



soft and collinear divergencies

Collinear factorization:

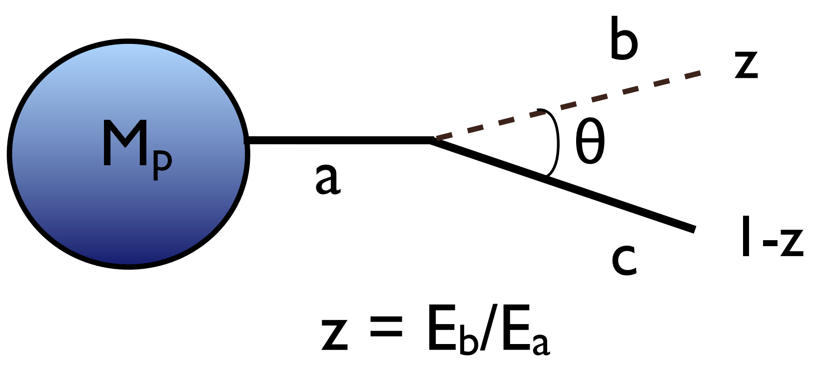
$$|M_{p+1}|^2 d\Phi_{p+1} \simeq |M_p|^2 d\Phi_p \frac{dt}{t}\frac{\alpha_S}{2\pi}P(z)dzd\phi$$

when θ is small.

Parton Shower basics

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

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



soft and collinear divergencies

More about this later!

Collinear factorization:

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

when θ is small.

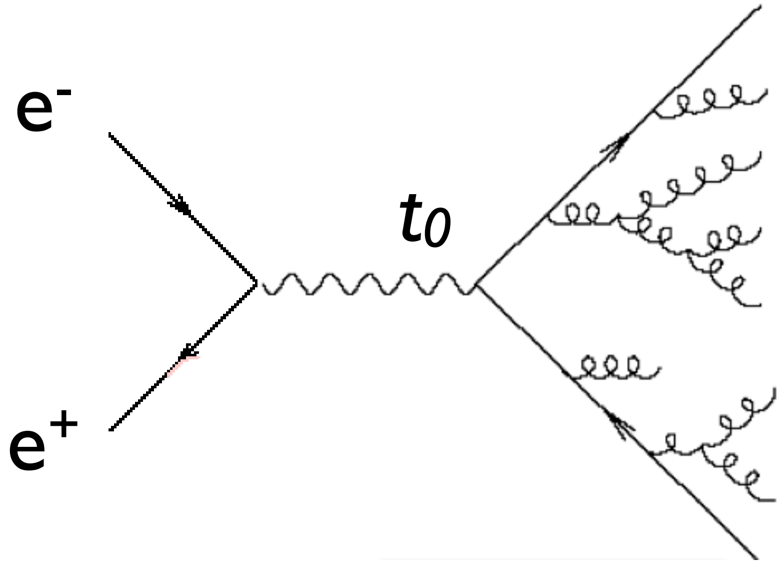
Parton showers

Parton showers

- Factorization allows us to simulate QCD multi-particle final states by performing many 2-particle splittings

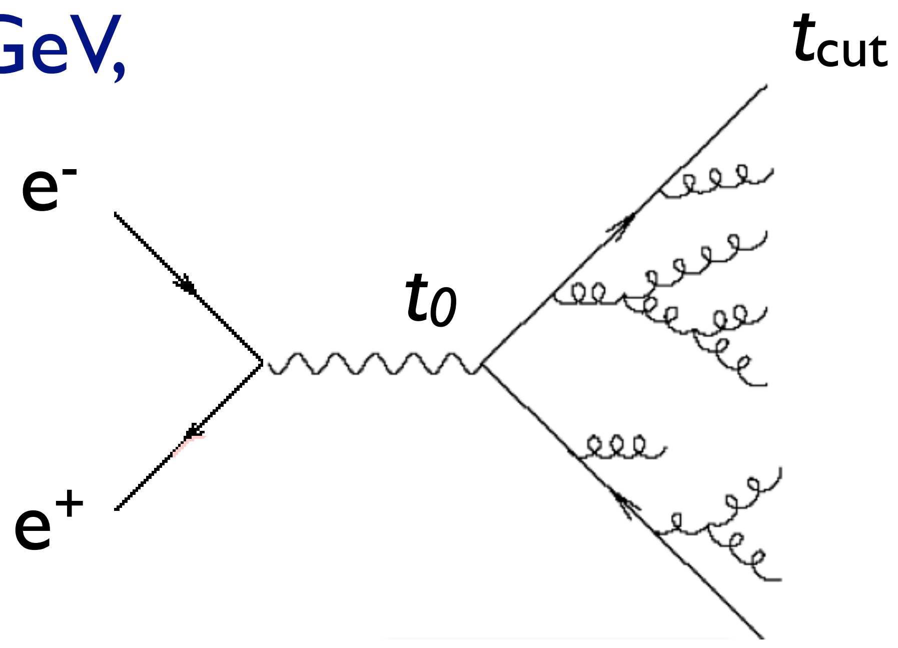
Parton showers

- Factorization allows us to simulate QCD multi-particle final states by performing many 2-particle splittings
- The result is a "cascade" or "shower" of partons with ever smaller virtualities.



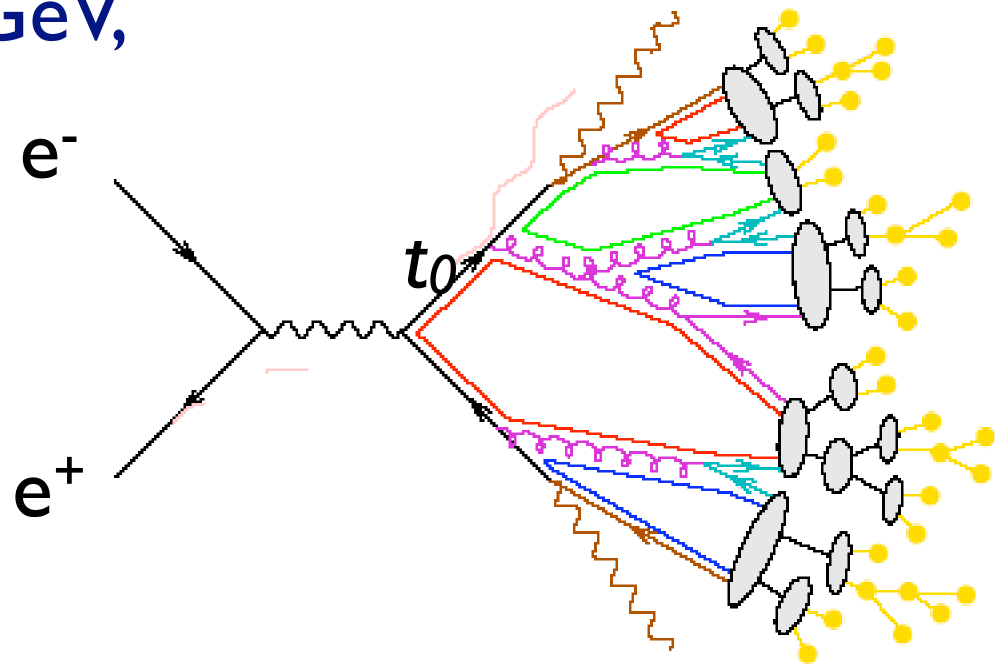
Parton showers

- Factorization allows us to simulate QCD multi-particle final states by performing many 2-particle splittings
- The result is a "cascade" or "shower" of partons with ever smaller virtualities.
- The procedure stops when the scale of the splitting is below some t_{cut} , usually close to 1 GeV, the scale where non-perturbative effects start dominating over the perturbative parton shower.



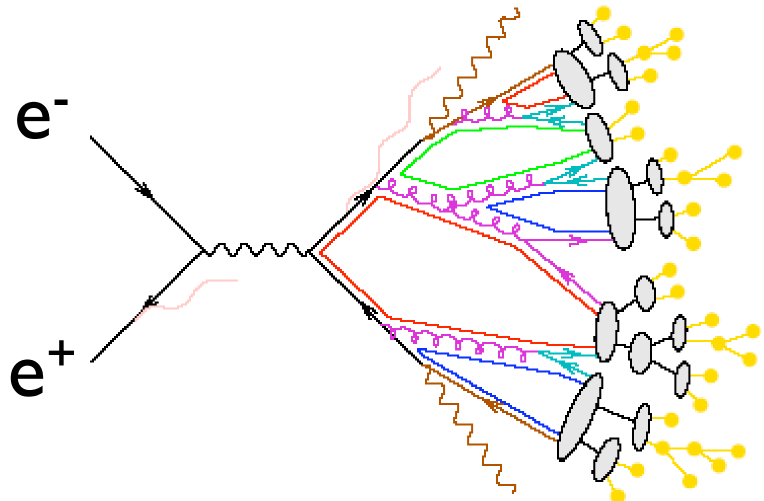
Parton showers

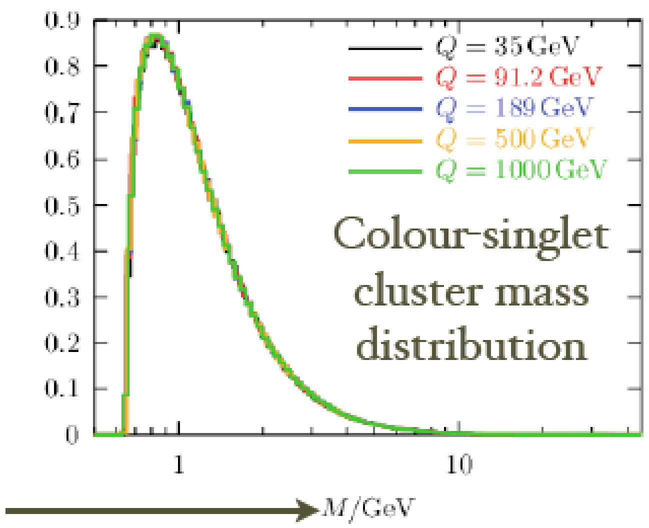
- Factorization allows us to simulate QCD multi-particle final states by performing many 2-particle splittings
- The result is a "cascade" or "shower" of partons with ever smaller virtualities.
- The procedure stops when the scale of the splitting is below some t_{cut} , usually close to 1 GeV, the scale where non-perturbative effects start dominating over the perturbative parton shower.
- At this point, partons must turn into color-neutral hadrons.



From Parton Showers to Hadronization

- The parton shower evolves the hard scattering down to the scale of O(1GeV).
- At this scale, QCD is no longer perturbative. Some hadronization model is used to describe the transition from the perturbative PS region to the non-perturbative hadronization region.
- Main hadronization models:
  - ➡ String hadronization (Pythia) [Andersson,Gustafson,Ingelman,Sjöstrand (1983)]
  - ➡ Cluster hadronization (Herwig) [Webber (1984)]
- Hadronization only acts locally, not sensitive to high-q^2 scattering.





Parton Shower MC event generators

- General-purpose tools
- Complete exclusive description of the events: hard scattering, showering, hadronization, underlying event
- Reliable and well tuned to experimental data.

most well-known: PYTHIA, HERWIG, SHERPA

- You will hear much more about Parton Showers in the next lecture

Detector simulation

- Detector simulation
  - ➡ Fast general-purpose detector simulators: Delphes, PGS (“Pretty good simulations”), AcerDet
  - ➡ Specify parameters to simulate different experiments
- Experiment-specific fast simulation
  - ➡ Detector response parameterized
  - ➡ Run time: ms-s/event
- Experiment-specific full simulation
  - ➡ Full tracking of particles through detector using GEANT
  - ➡ Run time: several minutes/event

Summary of Background

Summary of Background

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

Summary of Background

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

Summary of Background

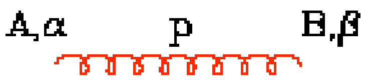
- Despite the apparent enormous complexity of simulation of complete collider events, nature has kindly allowed us to factorize the simulation into separate steps
- The Monte Carlo method allows us to step-by-step simulate hard scattering, parton shower, particle decays, hadronization, and underlying event
- Hard-working MC program developers have provided a multitude of tools that can be used to simulate complete collider events with a few keystrokes and the click of a button

Simulation with MadGraph 5

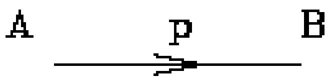
Outline:

- Computing the matrix element
  - ➡ HELAS / ALOHA
- Features of MG5
- Live demonstration

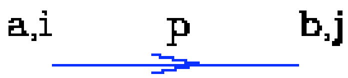
The Matrix Element



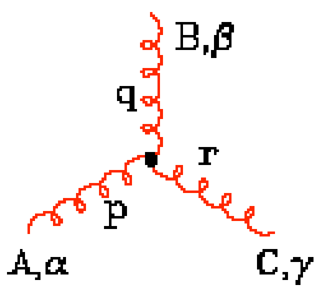
$$\delta^{AB}\left[-g^{\alpha\beta}+(1-\lambda)\frac{p^\alpha p^\beta}{p^2+i\epsilon}\right]\frac{i}{p^2+i\epsilon}$$



$$\delta^{AB}\frac{i}{(p^2+i\epsilon)}$$

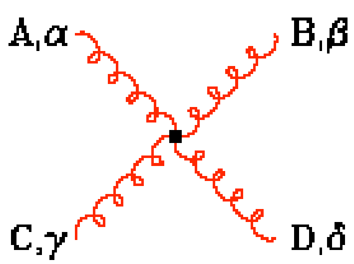


$$\delta^{ab}\frac{i}{(\not{p}-m+i\epsilon)_{ji}}$$

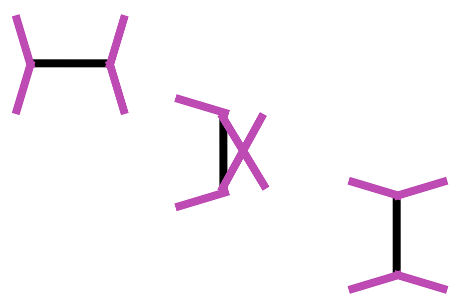


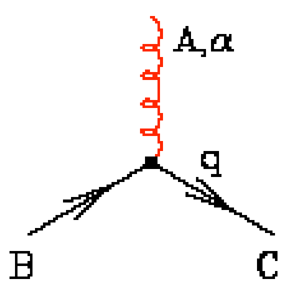
$$-g\ f^{ABC}[(p-q)^\gamma g^{\alpha\beta}+(q-r)^\alpha g^{\beta\gamma}+(r-p)^\beta g^{\gamma\alpha}]$$

(all momenta incoming)

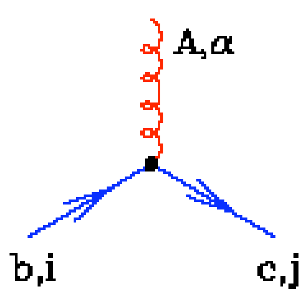


$$-ig^2\ f^{XAC}f^{XBD}\ [g^{\alpha\beta}g^{\gamma\delta}-g^{\alpha\delta}g^{\beta\gamma}]$$
$$-ig^2\ f^{XAD}f^{XBC}\ [g^{\alpha\beta}g^{\gamma\delta}-g^{\alpha\gamma}g^{\beta\delta}]$$
$$-ig^2\ f^{XAB}f^{XCD}\ [g^{\alpha\gamma}g^{\beta\delta}-g^{\alpha\delta}g^{\beta\gamma}]$$



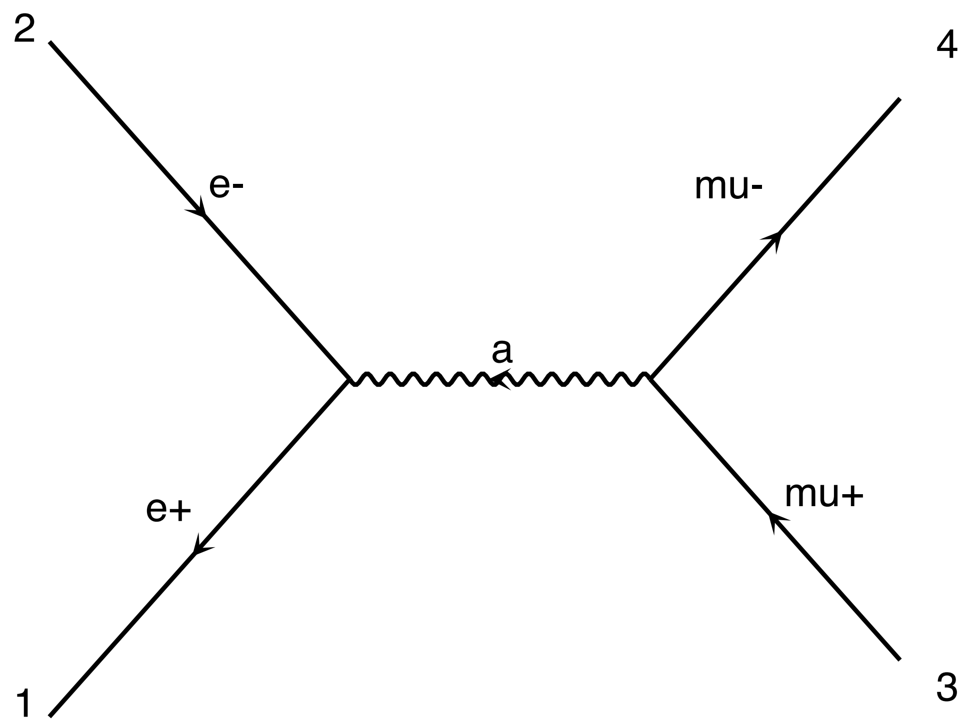


$$g\ f^{ABC}q^\alpha$$

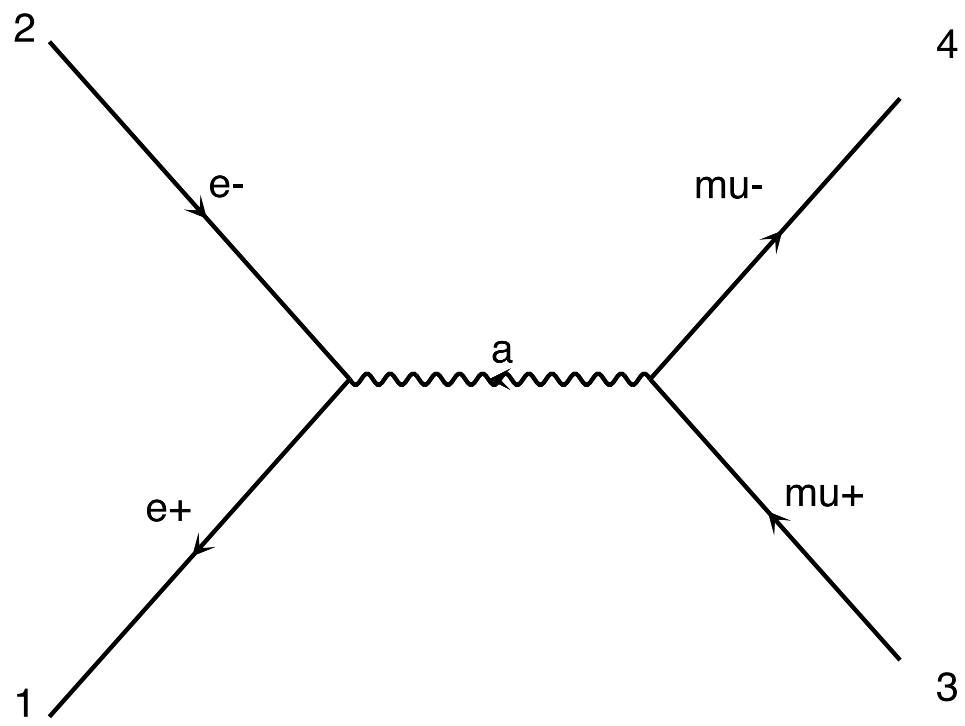


$$-ig\ (t^A)_{cb}\ (\gamma^\alpha)_{ji}$$

Evaluating a square Matrix Element

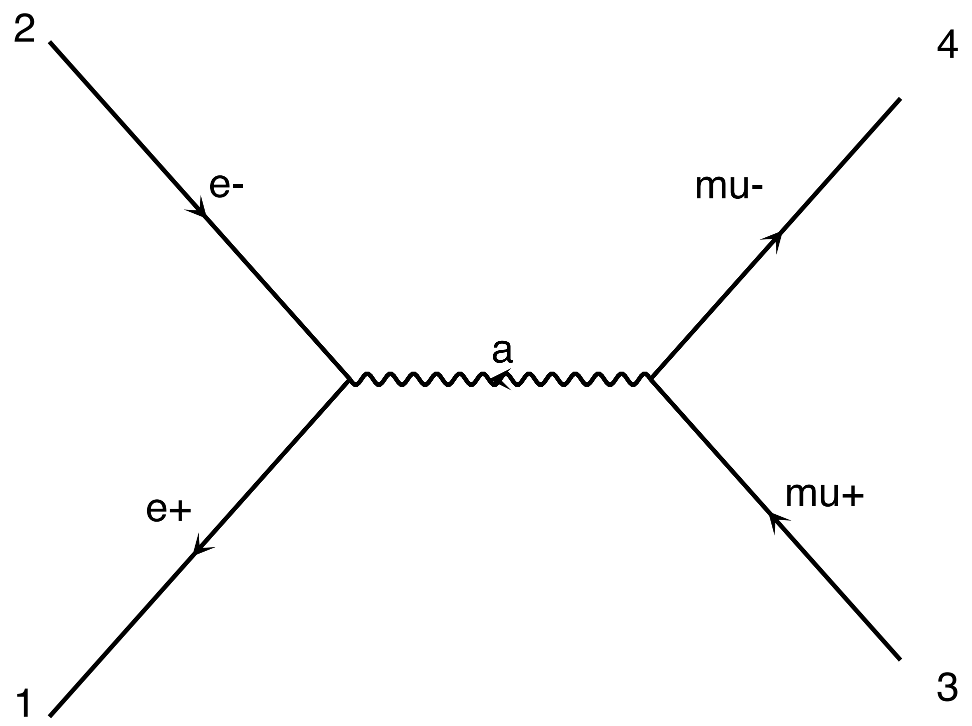


Evaluating a square Matrix Element



$$\mathcal{M} = e^2(\bar{u}\gamma^\mu v)\,\frac{g_{\mu\nu}}{q^2}\,(\bar{u}\gamma^\nu v)$$

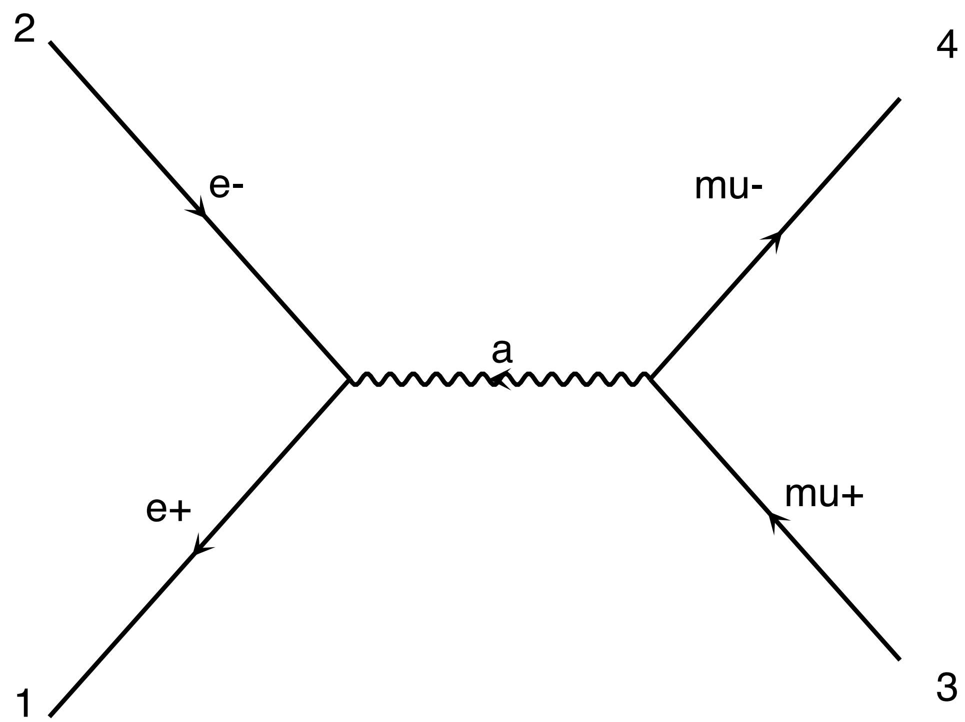
Evaluating a square Matrix Element



$$\mathcal{M} = e^2(\bar{u}\gamma^{\mu}v)\,\frac{g_{\mu\nu}}{q^2}\,(\bar{u}\gamma^{\nu}v)$$

$$\frac{1}{4}\sum_{pol}|\mathcal{M}|^2 = \frac{1}{4}\sum_{pol}\mathcal{M}^*\mathcal{M}$$

Evaluating a square Matrix Element

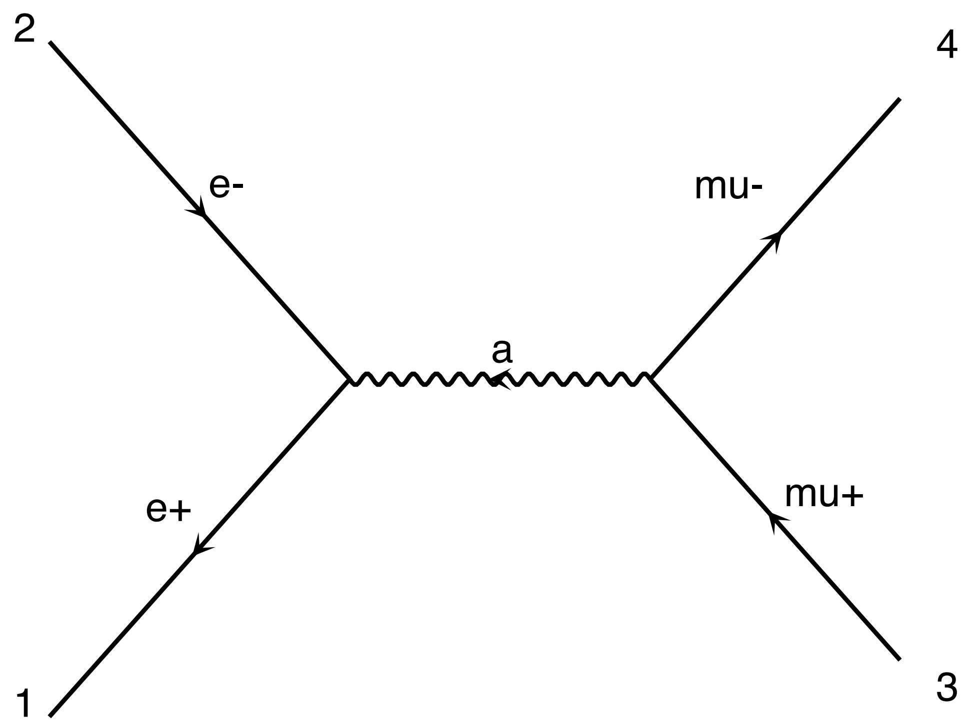


$$\mathcal{M} = e^2 (\bar{u}\gamma^\mu v) \frac{g_{\mu\nu}}{q^2} (\bar{u}\gamma^\nu v)$$

$$\frac{1}{4}\sum_{pol} |\mathcal{M}|^2 = \frac{1}{4}\sum_{pol} \mathcal{M}^* \mathcal{M}$$

$$\sum_{pol} \bar{u}u = \not{p} + m$$

Evaluating a square Matrix Element



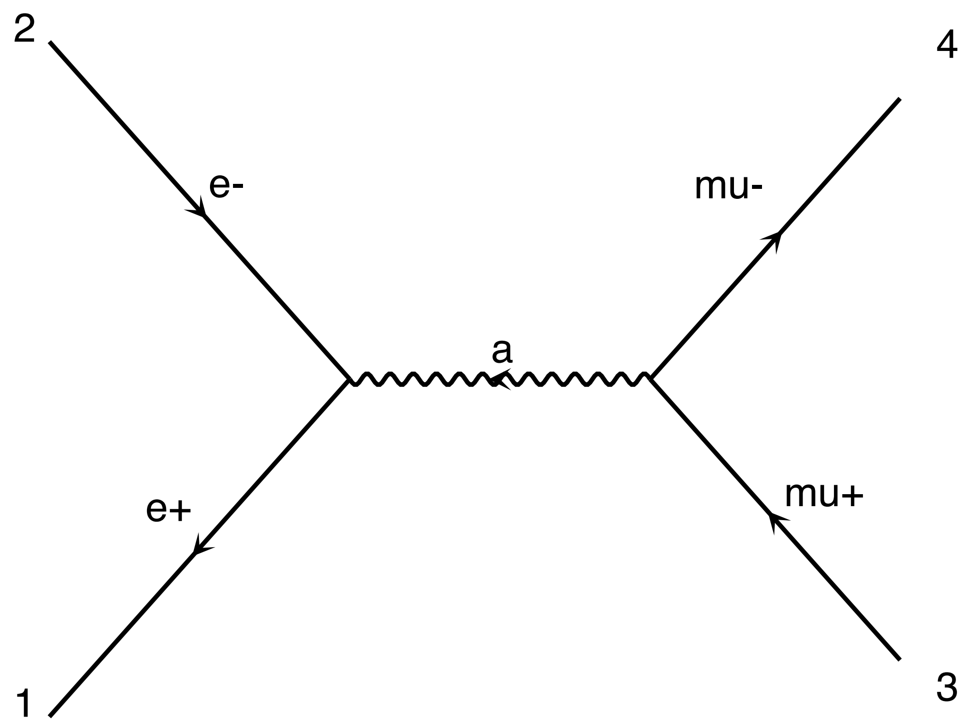
$$\mathcal{M} = e^2(\bar{u}\gamma^\mu v)\,\frac{g_{\mu\nu}}{q^2}\,(\bar{u}\gamma^\nu v)$$

$$\frac{1}{4}\sum_{pol}|\mathcal{M}|^2 = \frac{1}{4}\sum_{pol}\mathcal{M}^*\mathcal{M}$$

$$\sum_{pol}\bar{u}u = \not{p} + m$$

$$\rightarrow \frac{e^4}{4q^4}Tr[\not{p}_1\gamma^\mu\not{p}_2\gamma^\nu]Tr[\not{p}_3\gamma_\mu\not{p}_4\gamma_\nu]$$

Evaluating a square Matrix Element



$$\mathcal{M} = e^2(\bar{u}\gamma^\mu v)\,\frac{g_{\mu\nu}}{q^2}\,(\bar{u}\gamma^\nu v)$$

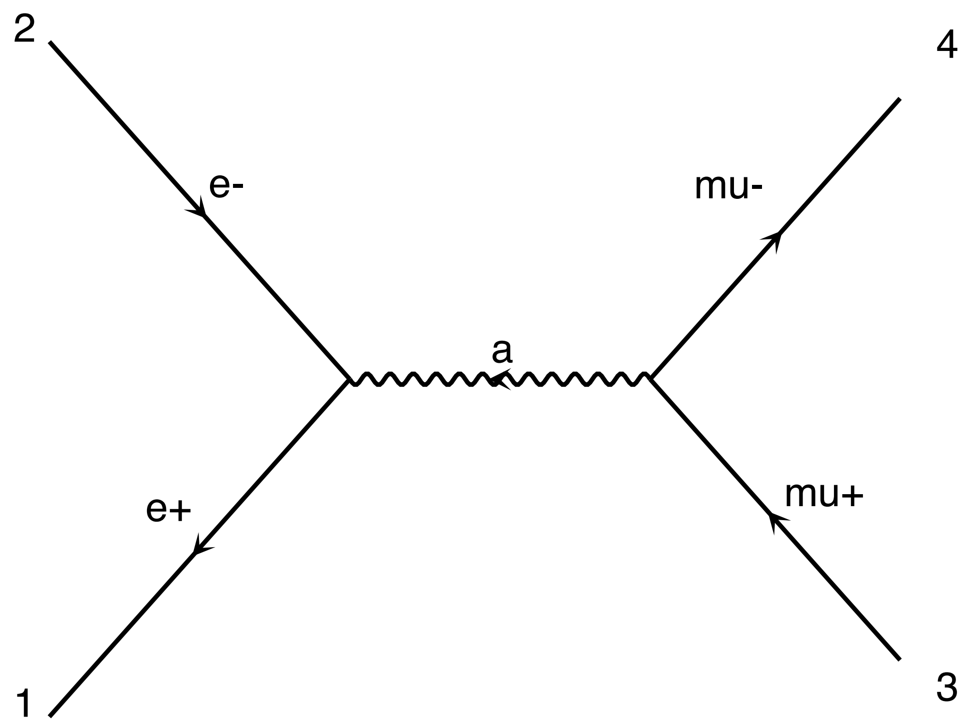
$$\frac{1}{4}\sum_{pol}|\mathcal{M}|^2 = \frac{1}{4}\sum_{pol}\mathcal{M}^*\mathcal{M}$$

$$\sum_{pol}\bar{u}u = \not{p} + m$$

$$\rightarrow \quad \frac{e^4}{4q^4}Tr[\not{p}_1\gamma^\mu\not{p}_2\gamma^\nu]Tr[\not{p}_3\gamma_\mu\not{p}_4\gamma_\nu]$$

Very Efficient !!!

Evaluating a square Matrix Element



$$\mathcal{M} = e^2 (\bar{u}\gamma^\mu v) \frac{g_{\mu\nu}}{q^2} (\bar{u}\gamma^\nu v)$$

$$\frac{1}{4}\sum_{pol} |\mathcal{M}|^2 = \frac{1}{4}\sum_{pol} \mathcal{M}^* \mathcal{M}$$

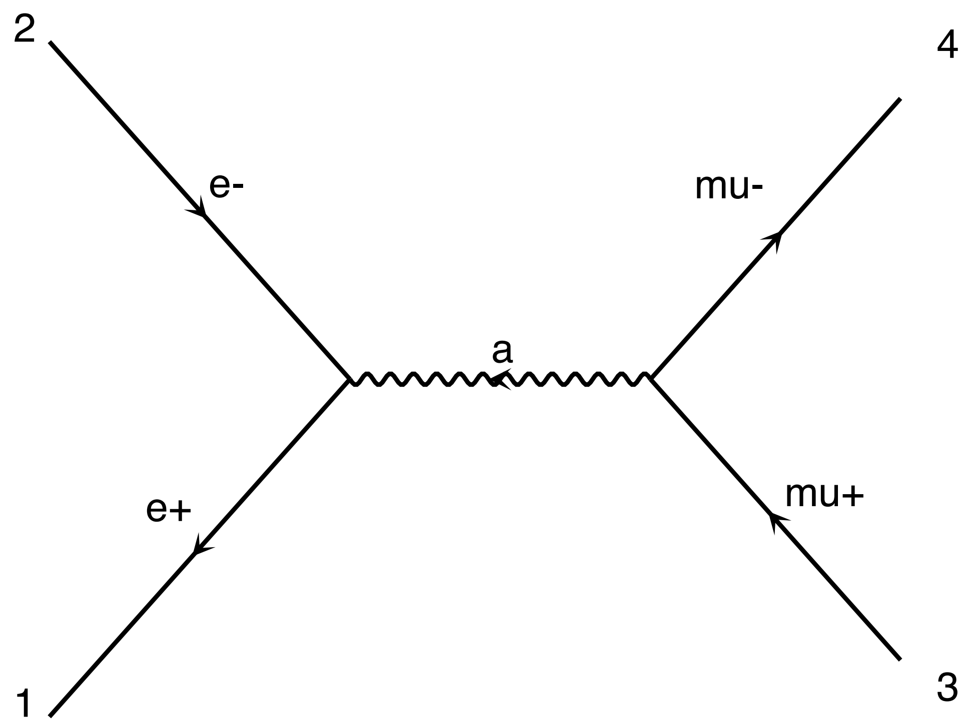
$$\sum_{pol} \bar{u}u = \not{p} + m$$

$$\rightarrow \frac{e^4}{4q^4} Tr[\not{p}_1 \gamma^\mu \not{p}_2 \gamma^\nu] Tr[\not{p}_3 \gamma_\mu \not{p}_4 \gamma_\nu]$$

Very Efficient !!!

But the number of terms rises as N^2

Evaluating a square Matrix Element



$$\mathcal{M} = e^2 (\bar{u}\gamma^\mu v)\,\frac{g_{\mu\nu}}{q^2}\,(\bar{u}\gamma^\nu v)$$

$$\frac{1}{4}\sum_{pol} |\mathcal{M}|^2 = \frac{1}{4}\sum_{pol} \mathcal{M}^* \mathcal{M}$$

$$\sum_{pol} \bar{u}u = \not{p} + m$$

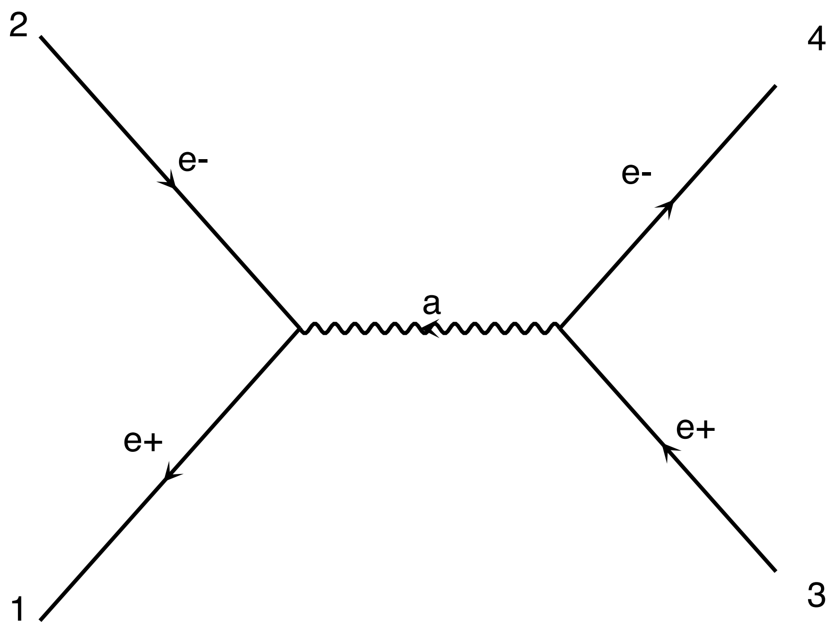
$$\rightarrow \quad \frac{e^4}{4q^4} Tr[\not{p}_1 \gamma^\mu \not{p}_2 \gamma^\nu] Tr[\not{p}_3 \gamma_\mu \not{p}_4 \gamma_\nu]$$

Very Efficient !!!

But the number of terms rises as N^2

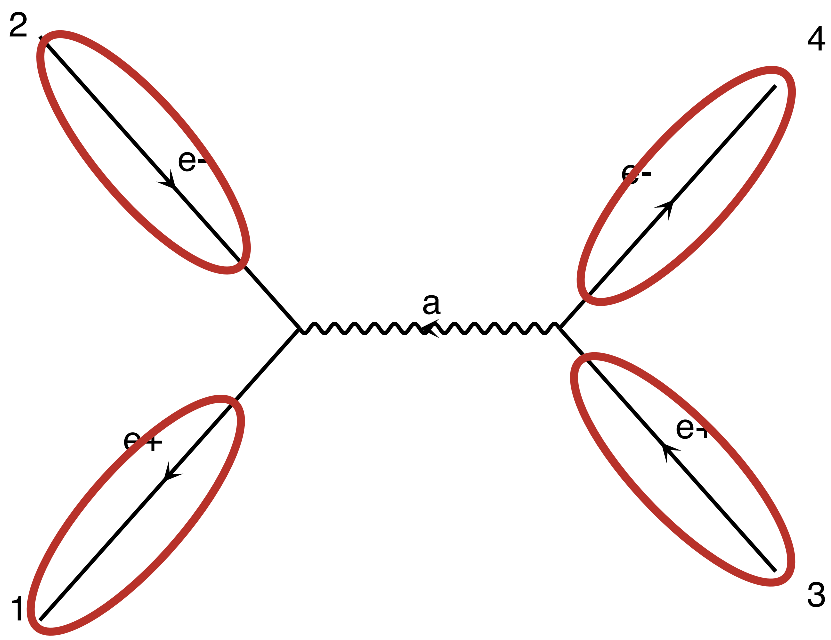
Only for 2→2 and 2→3

- Idea: Evaluate $\mathcal{M}$ for fixed helicity of external particles
  - ➡ Multiply $\mathcal{M}$ with $\mathcal{M}^*$ -> $|\mathcal{M}|$^2
  - ➡ Loop on Helicity and sum the results



$$\mathcal{M} = \bar{u}\gamma^\mu v P_{\mu\nu} \bar{u}\gamma^\nu v$$

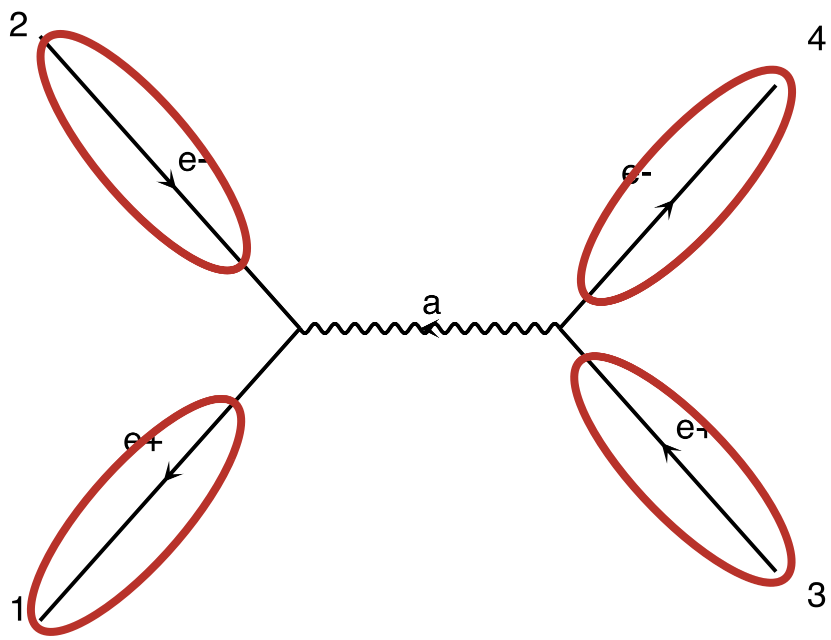
- Idea: Evaluate $\mathcal{M}$ for fixed helicity of external particles
  - ➡ Multiply $\mathcal{M}$ with $\mathcal{M}^*$ -> $|\mathcal{M}|$^2
  - ➡ Loop on Helicity and sum the results



$$\mathcal{M} = \bar{u}\gamma^{\mu}v P_{\mu\nu}\bar{u}\gamma^{\nu}v$$

Numbers for given helicity and momenta

- Idea: Evaluate $\mathcal{M}$ for fixed helicity of external particles
  - ➡ Multiply $\mathcal{M}$ with $\mathcal{M}^*$ -> |$\mathcal{M}$|^2
  - ➡ Loop on Helicity and sum the results

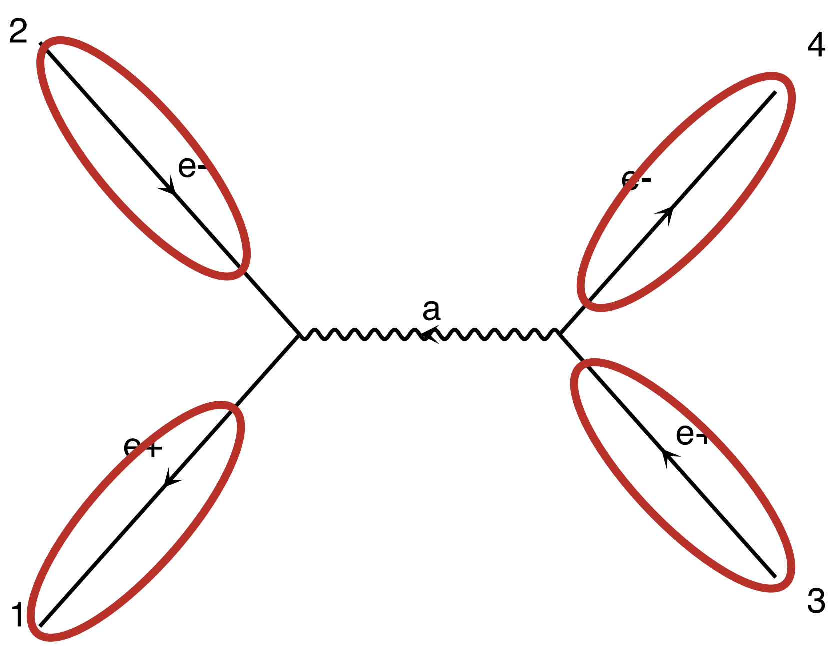


$$\mathcal{M} = \bar{u}\gamma^\mu v P_{\mu\nu} \bar{u}\gamma^\nu v$$

Numbers for given helicity and momenta

```
CALL OXXXXX(P(0,1),ZERO,NHEL(1),-1*IC(1),W(1,1))
CALL IXXXXX(P(0,2),ZERO,NHEL(2),+1*IC(2),W(1,2))
CALL IXXXXX(P(0,3),ZERO,NHEL(3),-1*IC(3),W(1,3))
CALL OXXXXX(P(0,4),ZERO,NHEL(4),+1*IC(4),W(1,4))
```

- Idea: Evaluate $\mathcal{M}$ for fixed helicity of external particles
  - ➡ Multiply $\mathcal{M}$ with $\mathcal{M}^*$ -> $|\mathcal{M}|$^2
  - ➡ Loop on Helicity and sum the results



$$\mathcal{M} = \bar{u}\gamma^\mu v P_{\mu\nu} \bar{u}\gamma^\nu v$$

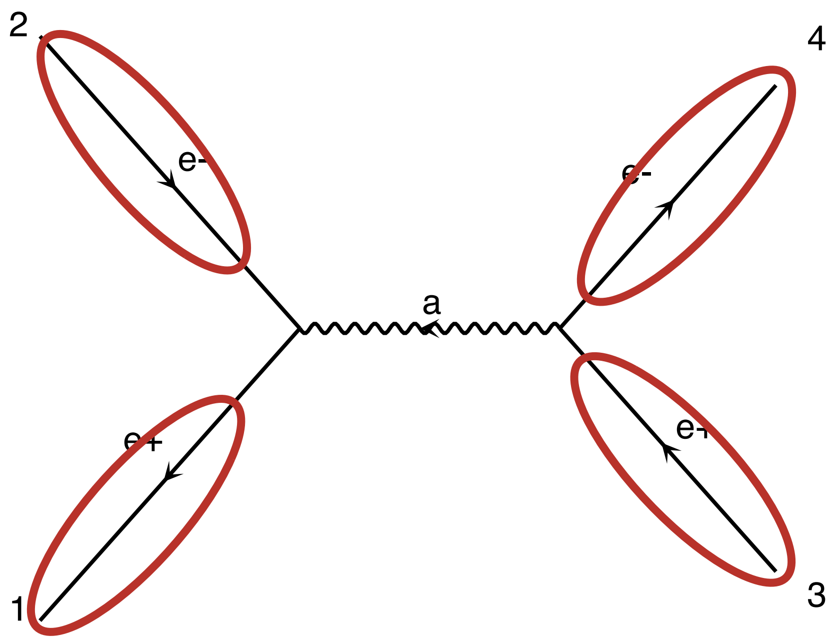
Numbers for given helicity and momenta

```
CALL OXXXXX(P(0,1),ZERO,NHEL(1),-1*IC(1),W(1,1))
```

Input: momenta, mass, helicity

Ouput: Wavefunction (given by an analytical formula)

- Idea: Evaluate $\mathcal{M}$ for fixed helicity of external particles
  - ➡ Multiply $\mathcal{M}$ with $\mathcal{M}^*$ -> |$\mathcal{M}$|^2
  - ➡ Loop on Helicity and sum the results

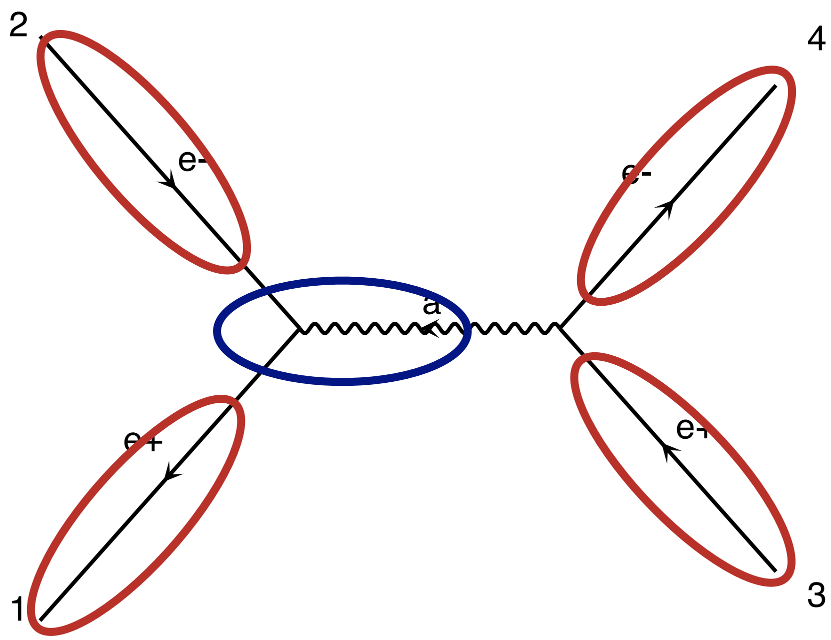


$$\mathcal{M} = \bar{u}\gamma^{\mu}v P_{\mu\nu}\bar{u}\gamma^{\nu}v$$

Numbers for given helicity and momenta

```
CALL OXXXXX(P(0,1),ZERO,NHEL(1),-1*IC(1),W(1,1))
CALL IXXXXX(P(0,2),ZERO,NHEL(2),+1*IC(2),W(1,2))
CALL IXXXXX(P(0,3),ZERO,NHEL(3),-1*IC(3),W(1,3))
CALL OXXXXX(P(0,4),ZERO,NHEL(4),+1*IC(4),W(1,4))
```

- Idea: Evaluate $\mathcal{M}$ for fixed helicity of external particles
  - ➡ Multiply $\mathcal{M}$ with $\mathcal{M}^*$ -> $|\mathcal{M}|$^2
  - ➡ Loop on Helicity and sum the results



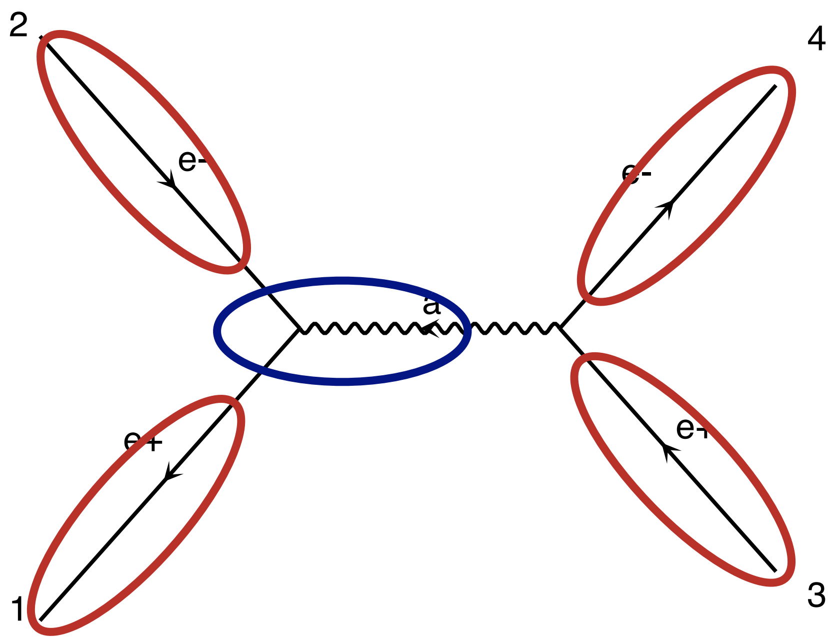
$$\mathcal{M} = \bar{u}\gamma^\mu v P_{\mu\nu} \bar{u}\gamma^\nu v$$

Numbers for given helicity and momenta

Calculate propagator wavefunctions

```
CALL OXXXXX(P(0,1),ZERO,NHEL(1),-1*IC(1),W(1,1))
CALL IXXXXX(P(0,2),ZERO,NHEL(2),+1*IC(2),W(1,2))
CALL IXXXXX(P(0,3),ZERO,NHEL(3),-1*IC(3),W(1,3))
CALL OXXXXX(P(0,4),ZERO,NHEL(4),+1*IC(4),W(1,4))
CALL JIOXXX(W(1,2),W(1,1),GAL,ZERO,ZERO,W(1,5))
```

- Idea: Evaluate $\mathcal{M}$ for fixed helicity of external particles
  - ➡ Multiply $\mathcal{M}$ with $\mathcal{M}^*$ -> $|\mathcal{M}|$^2
  - ➡ Loop on Helicity and sum the results



$$\mathcal{M} = \bar{u}\gamma^\mu v P_{\mu\nu} \bar{u}\gamma^\nu v$$

Numbers for given helicity and momenta

Calculate propagator wavefunctions

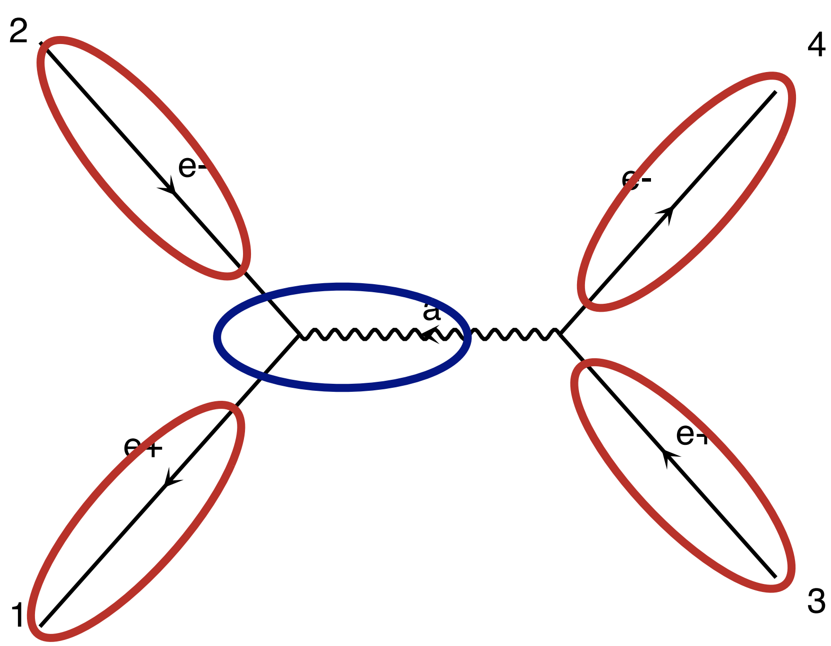
```
CALL OXXXXX(P(0,1),ZERO,NHEL(1),-1*IC(1),W(1,1))
CALL IXXXXX(P(0,2),ZERO,NHEL(2),+1*IC(2),W(1,2))
```

Input: Wavefunctions, mass, width, coupling

```
CALL JIOXXX(W(1,2),W(1,1),GAL,ZERO,ZERO,W(1,5))
```

Ouput: Wavefunction (given by an analytical formula)

- Idea: Evaluate $\mathcal{M}$ for fixed helicity of external particles
  - ➡ Multiply $\mathcal{M}$ with $\mathcal{M}^*$ -> $|\mathcal{M}|$^2
  - ➡ Loop on Helicity and sum the results



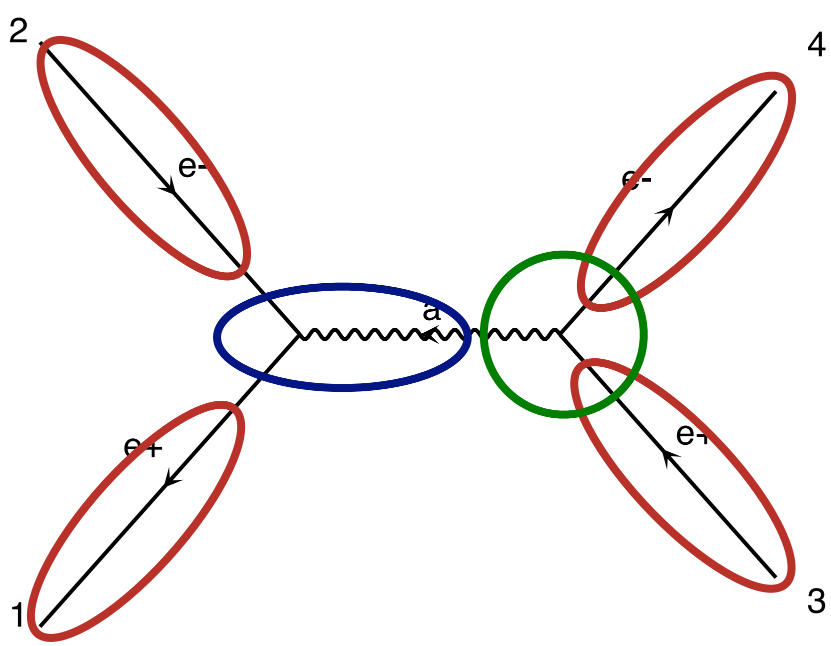
$$\mathcal{M} = \bar{u}\gamma^\mu v P_{\mu\nu} \bar{u}\gamma^\nu v$$

Numbers for given helicity and momenta

Calculate propagator wavefunctions

```
CALL OXXXXX(P(0,1),ZERO,NHEL(1),-1*IC(1),W(1,1))
CALL IXXXXX(P(0,2),ZERO,NHEL(2),+1*IC(2),W(1,2))
CALL IXXXXX(P(0,3),ZERO,NHEL(3),-1*IC(3),W(1,3))
CALL OXXXXX(P(0,4),ZERO,NHEL(4),+1*IC(4),W(1,4))
CALL JIOXXX(W(1,2),W(1,1),GAL,ZERO,ZERO,W(1,5))
```

- Idea: Evaluate $\mathcal{M}$ for fixed helicity of external particles
  - ➡ Multiply $\mathcal{M}$ with $\mathcal{M}^*$ -> $|\mathcal{M}|$^2
  - ➡ Loop on Helicity and sum the results



$$\mathcal{M} = \bar{u}\gamma^{\mu} v P_{\mu\nu} \bar{u}\gamma^{\nu} v$$

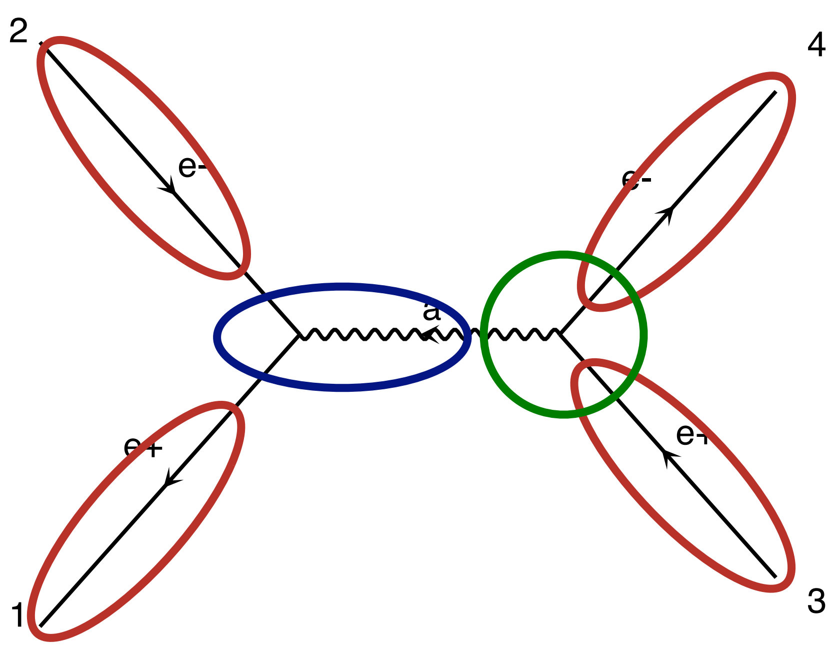
Numbers for given helicity and momenta

Calculate propagator wavefunctions

Finally evaluate amplitude (c-number)

```
CALL OXXXXX(P(0,1),ZERO,NHEL(1),-1*IC(1),W(1,1))
CALL IXXXXX(P(0,2),ZERO,NHEL(2),+1*IC(2),W(1,2))
CALL IXXXXX(P(0,3),ZERO,NHEL(3),-1*IC(3),W(1,3))
CALL OXXXXX(P(0,4),ZERO,NHEL(4),+1*IC(4),W(1,4))
CALL JIOXXX(W(1,2),W(1,1),GAL,ZERO,ZERO,W(1,5))
CALL IOVXXX(W(1,3),W(1,4),W(1,5),GAL,AMP(1))
```

- Idea: Evaluate $\mathcal{M}$ for fixed helicity of external particles
  - ➡ Multiply $\mathcal{M}$ with $\mathcal{M}^*$ -> $|\mathcal{M}|$^2
  - ➡ Loop on Helicity and sum the results



$$\mathcal{M} = \bar{u}\gamma^{\mu} v P_{\mu\nu} \bar{u}\gamma^{\nu} v$$

Numbers for given helicity and momenta

Calculate propagator wavefunctions

Finally evaluate amplitude (c-number)

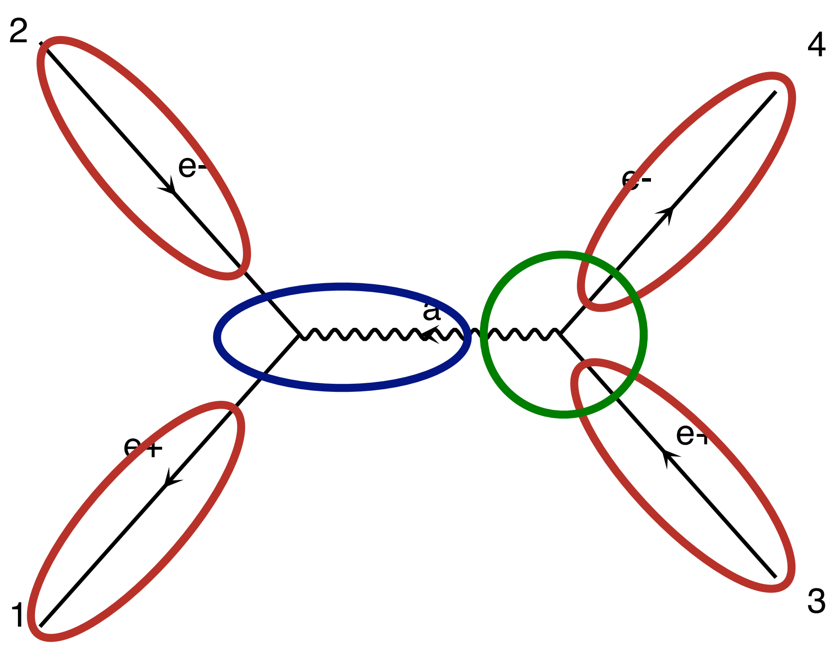
```
CALL OXXXXX(P(0,1),ZERO,NHEL(1),-1*IC(1),W(1,1))
CALL IXXXXX(P(0,2),ZERO,NHEL(2),+1*IC(2),W(1,2))
CALL IXXXXX(P(0,3),ZERO,NHEL(3),-1*IC(3),W(1,3))
```

Input: Wavefunctions, coupling

```
CALL IOVXXX(W(1,3),W(1,4),W(1,5),GAL,AMP(1))
```

Ouput: Amplitude

- Idea: Evaluate $\mathcal{M}$ for fixed helicity of external particles
  - ➡ Multiply $\mathcal{M}$ with $\mathcal{M}^*$ -> $|\mathcal{M}|$^2
  - ➡ Loop on Helicity and sum the results



$$\mathcal{M} = \bar{u}\gamma^{\mu}v P_{\mu\nu} \bar{u}\gamma^{\nu}v$$

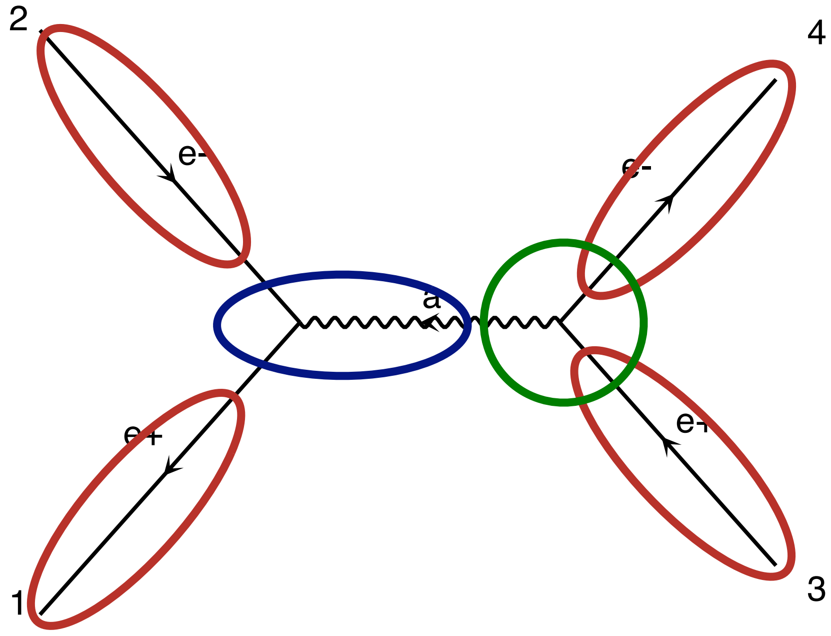
Numbers for given helicity and momenta

Calculate propagator wavefunctions

Finally evaluate amplitude (c-number)

```
CALL OXXXXX(P(0,1),ZERO,NHEL(1),-1*IC(1),W(1,1))
CALL IXXXXX(P(0,2),ZERO,NHEL(2),+1*IC(2),W(1,2))
CALL IXXXXX(P(0,3),ZERO,NHEL(3),-1*IC(3),W(1,3))
CALL OXXXXX(P(0,4),ZERO,NHEL(4),+1*IC(4),W(1,4))
CALL JIOXXX(W(1,2),W(1,1),GAL,ZERO,ZERO,W(1,5))
CALL IOVXXX(W(1,3),W(1,4),W(1,5),GAL,AMP(1))
```

- Idea: Evaluate $\mathcal{M}$ for fixed helicity of external particles
  - ➡ Multiply $\mathcal{M}$ with $\mathcal{M}^*$ -> $|\mathcal{M}|$^2
  - ➡ Loop on Helicity and sum the results



$$\mathcal{M} = \bar{u}\gamma^{\mu}v P_{\mu\nu}\bar{u}\gamma^{\nu}v$$

Numbers for given helicity and momenta

Calculate propagator wavefunctions

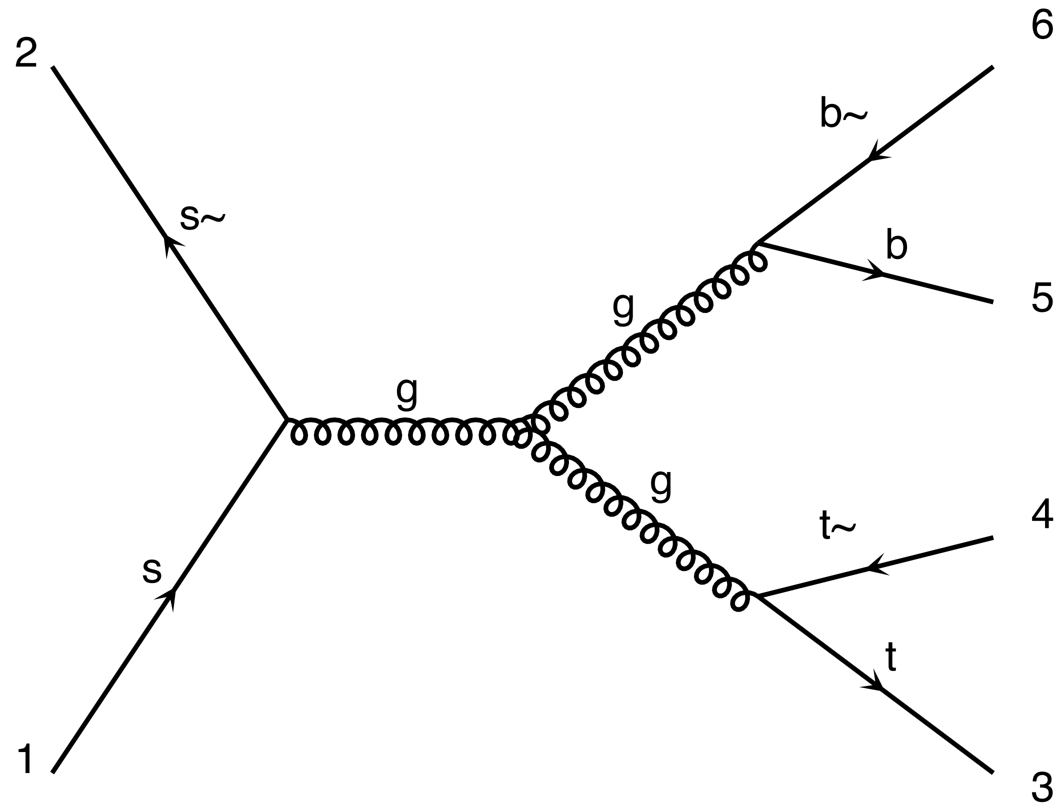
Finally evaluate amplitude (c-number)

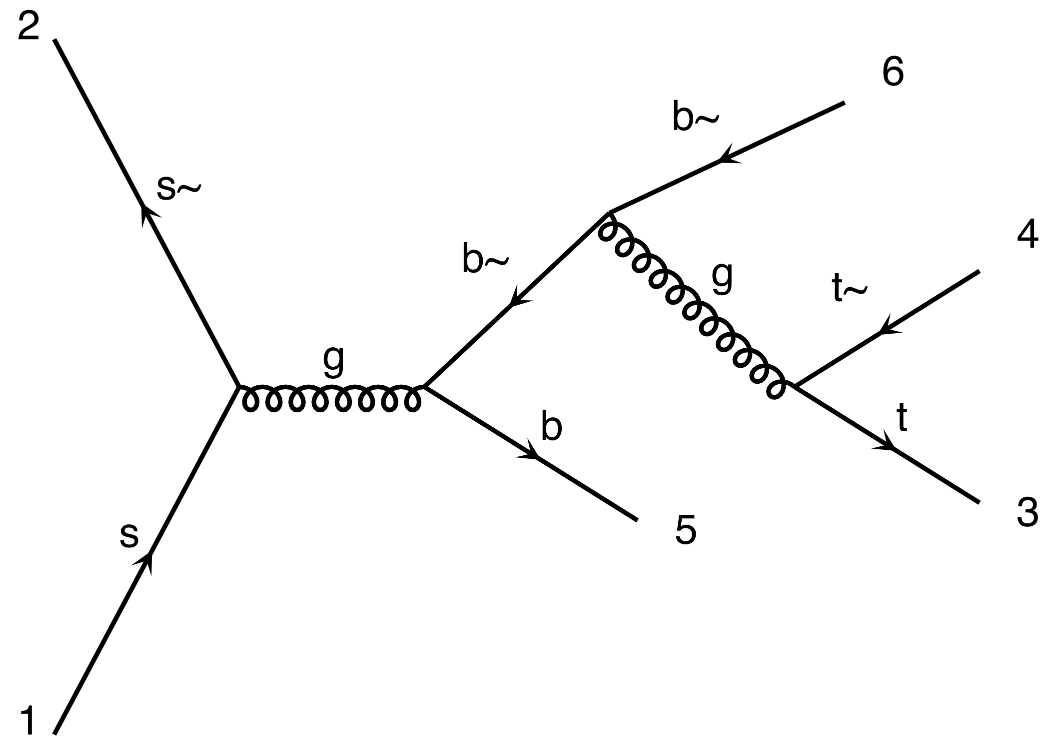
Helicity amplitude calls written by MadGraph

```
CALL OXXXXX(P(0,1),ZERO,NHEL(1),-1*IC(1),W(1,1))
CALL IXXXXX(P(0,2),ZERO,NHEL(2),+1*IC(2),W(1,2))
CALL IXXXXX(P(0,3),ZERO,NHEL(3),-1*IC(3),W(1,3))
CALL OXXXXX(P(0,4),ZERO,NHEL(4),+1*IC(4),W(1,4))
CALL JIOXXX(W(1,2),W(1,1),GAL,ZERO,ZERO,W(1,5))
CALL IOVXXX(W(1,3),W(1,4),W(1,5),GAL,AMP(1))
```

Real case

Known





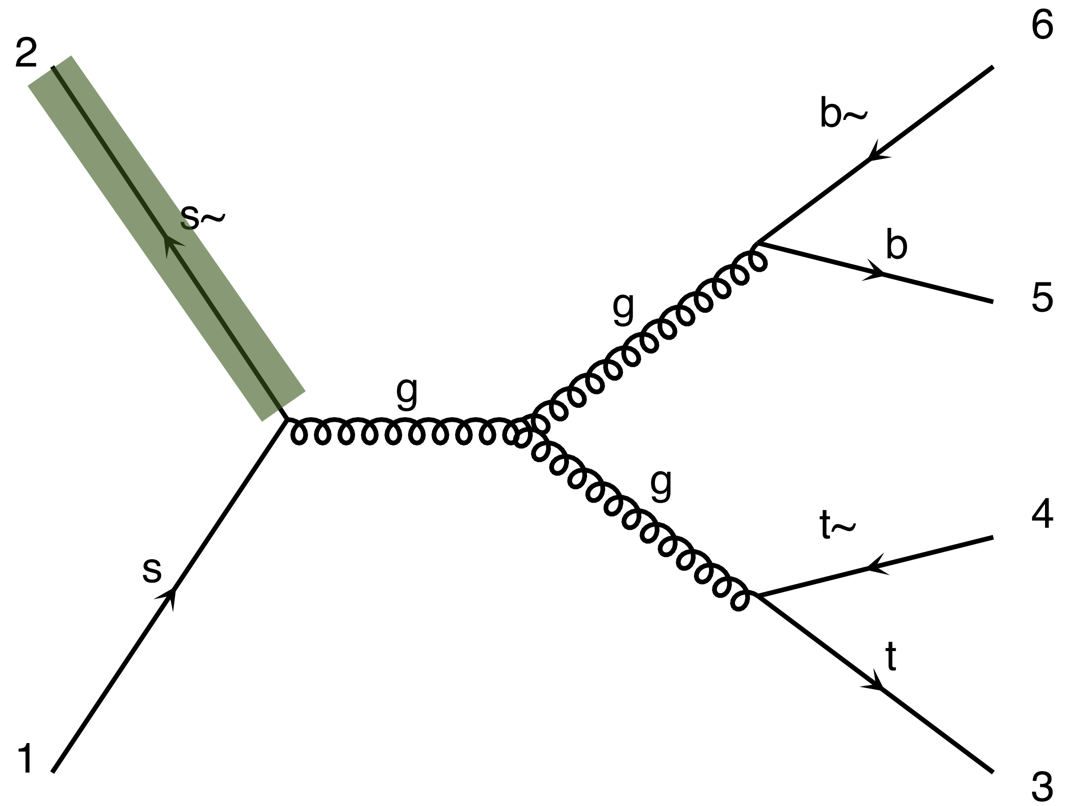
Number of routines: 0

Number of routines:0

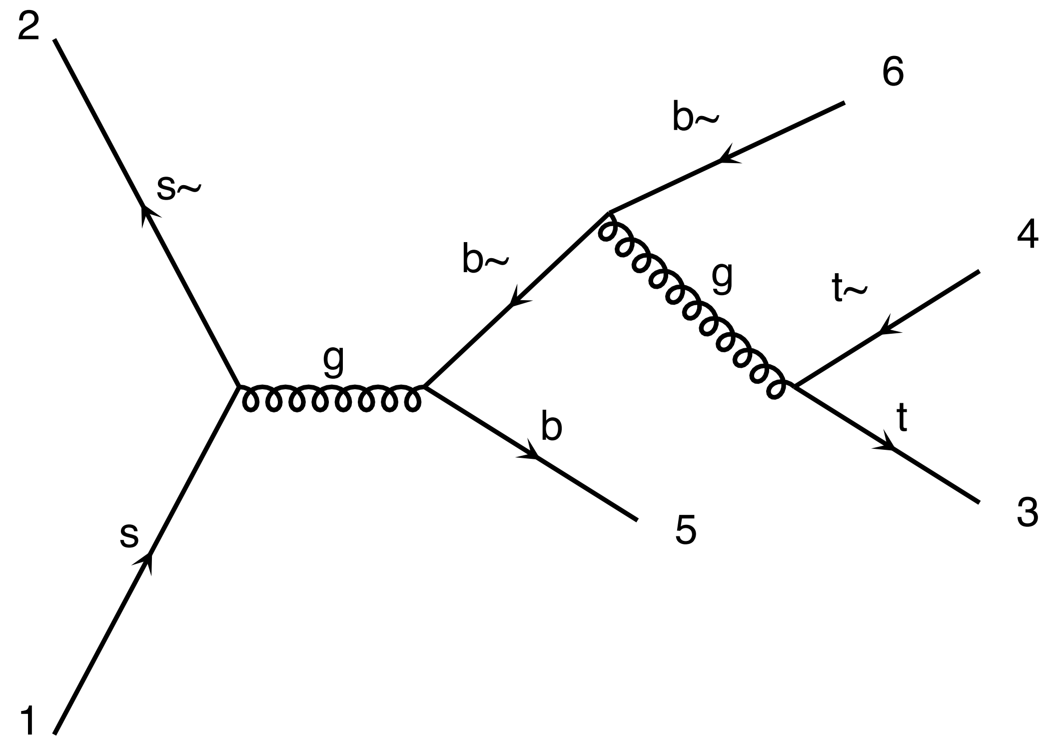
Number of routines for both: 0

Real case

Known



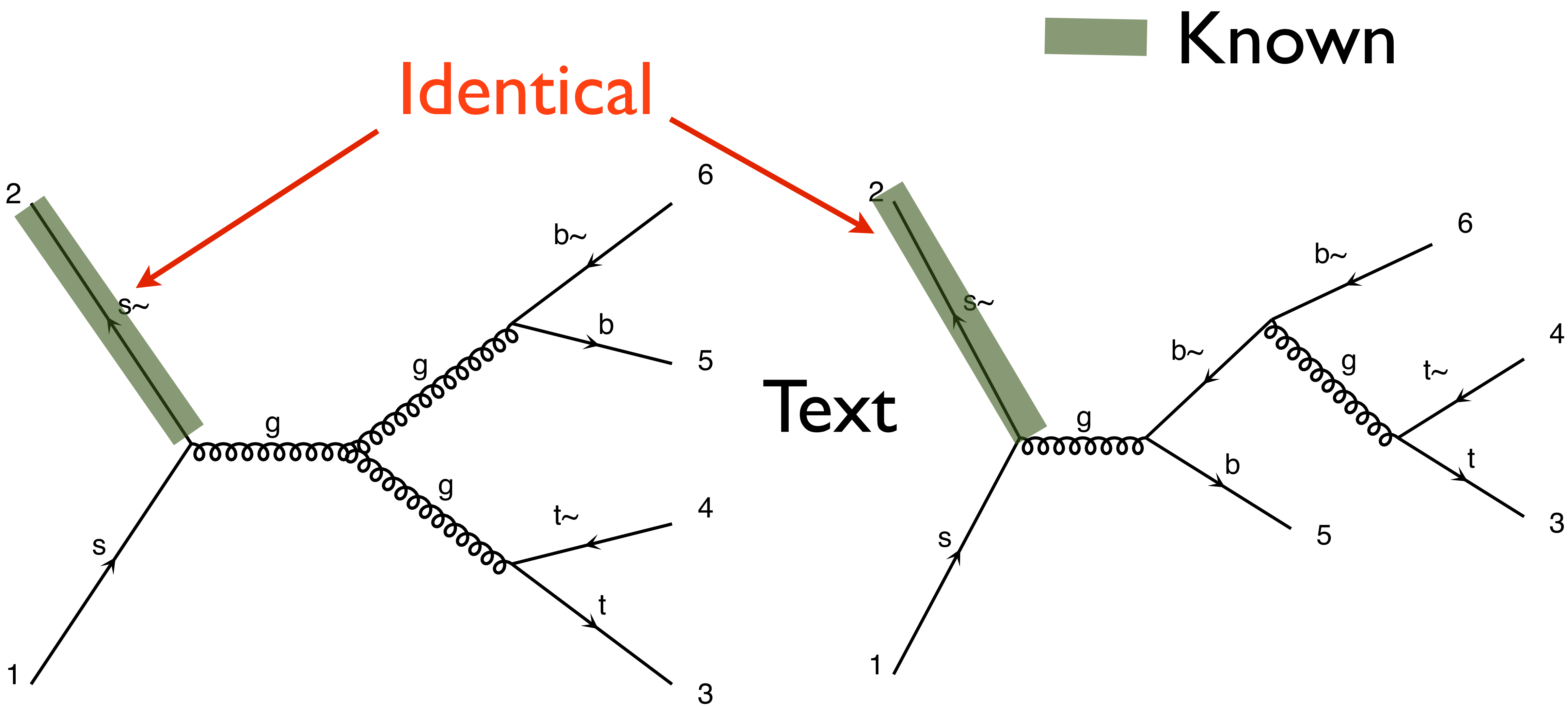
Number of routines:1



Number of routines:0

Number of routines for both: 1

Real case

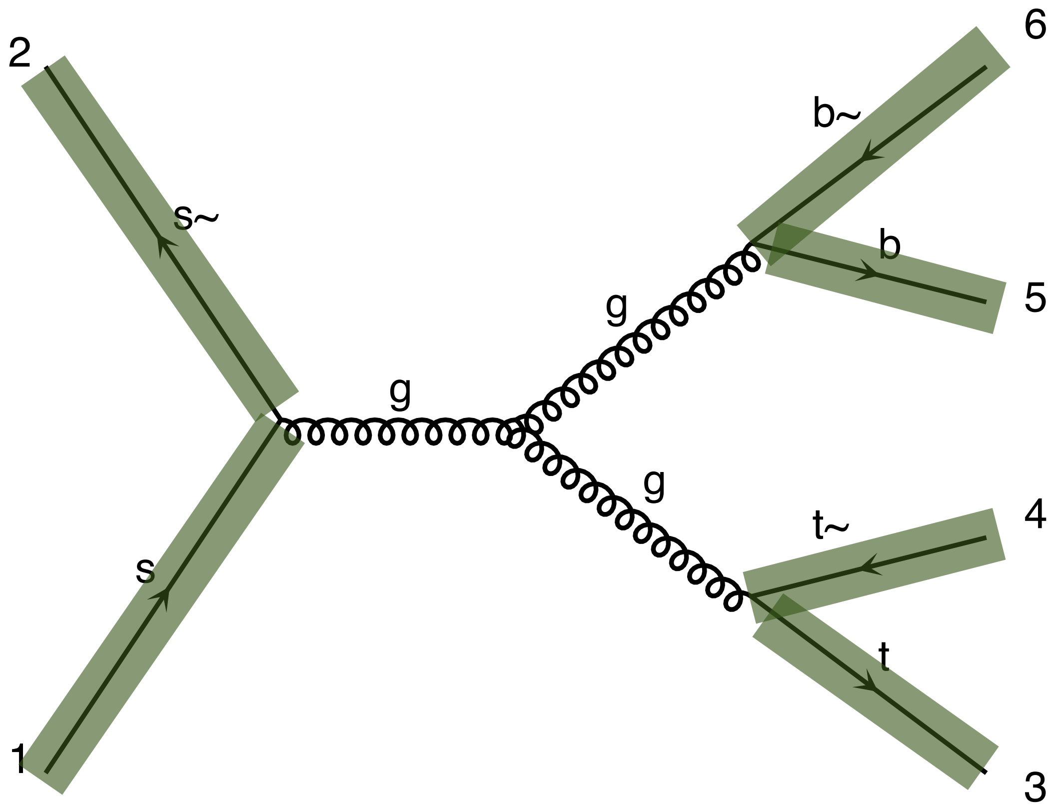


Number of routines: 1

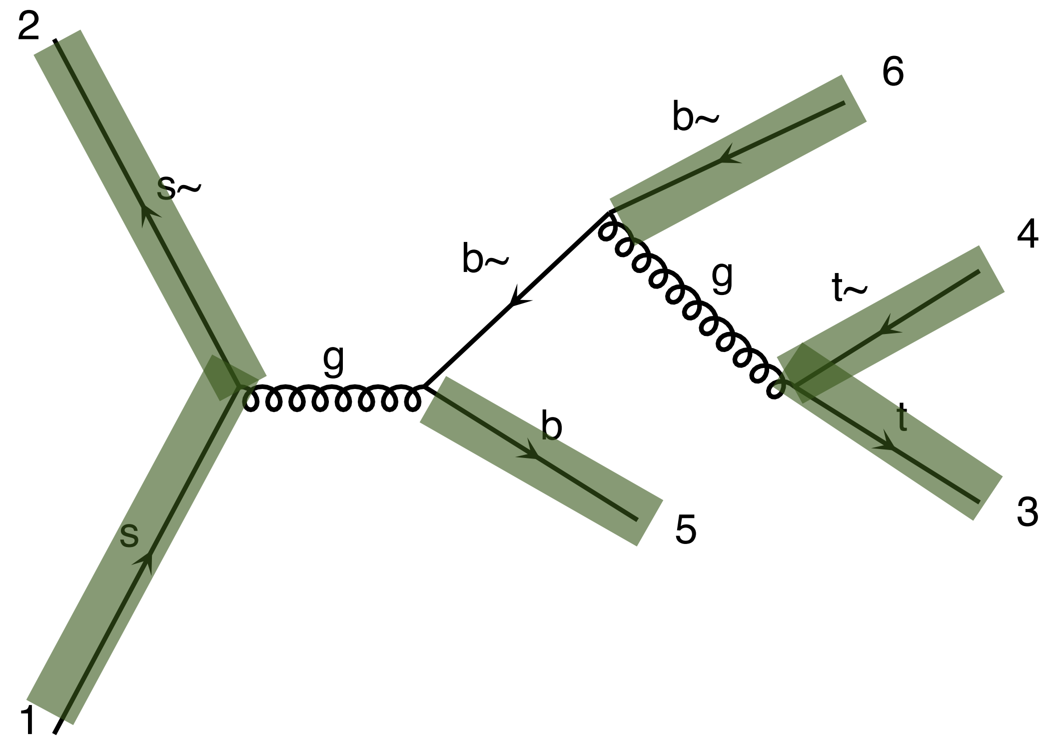
Number of routines: 1

Number of routines for both: 1

Real case



Number of routines: 6

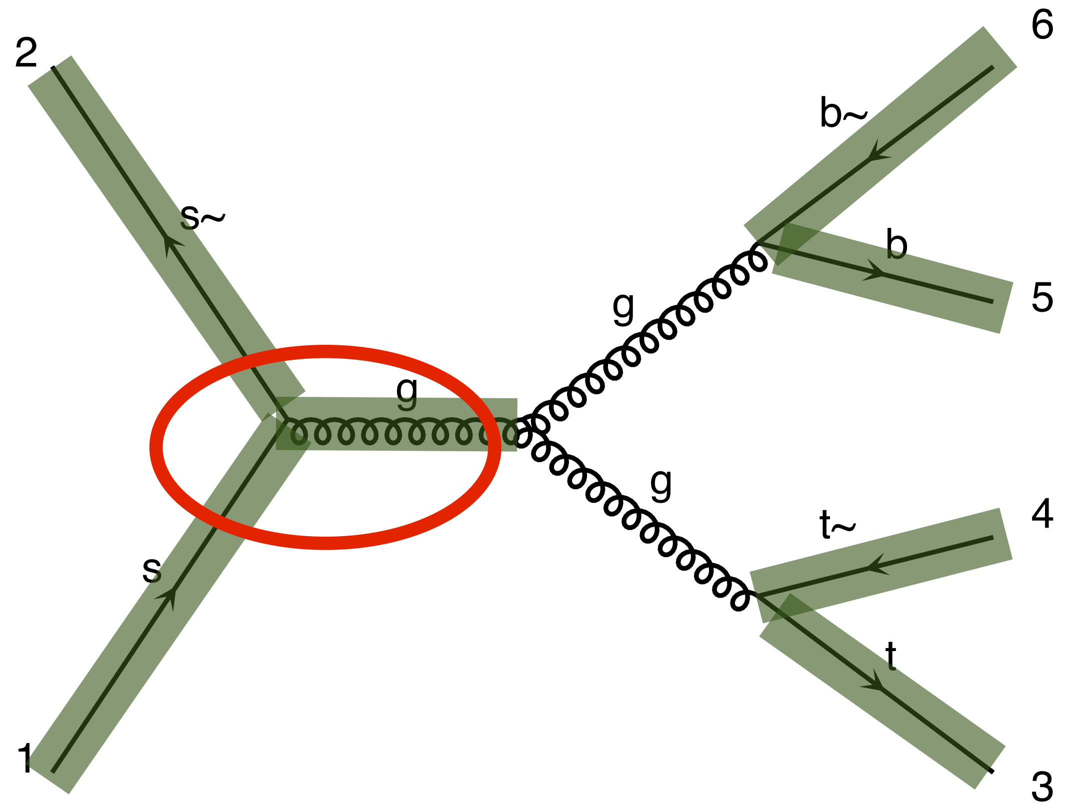


Number of routines: 6

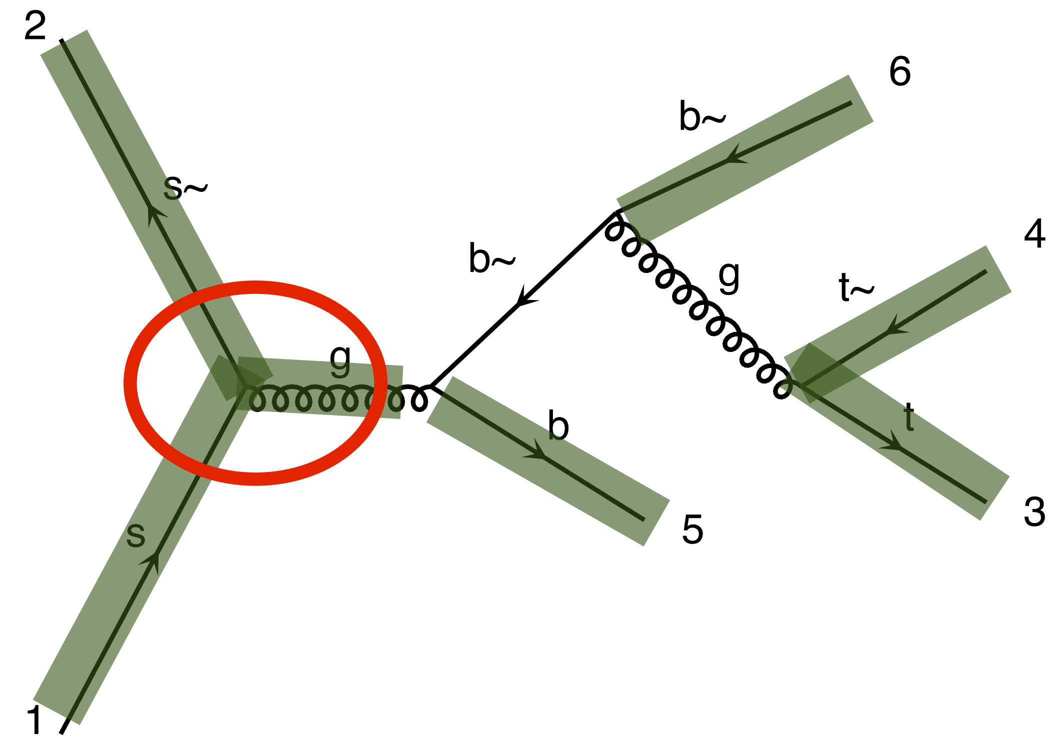
Number of routines for both: 6

Real case

Known



Number of routines: 7

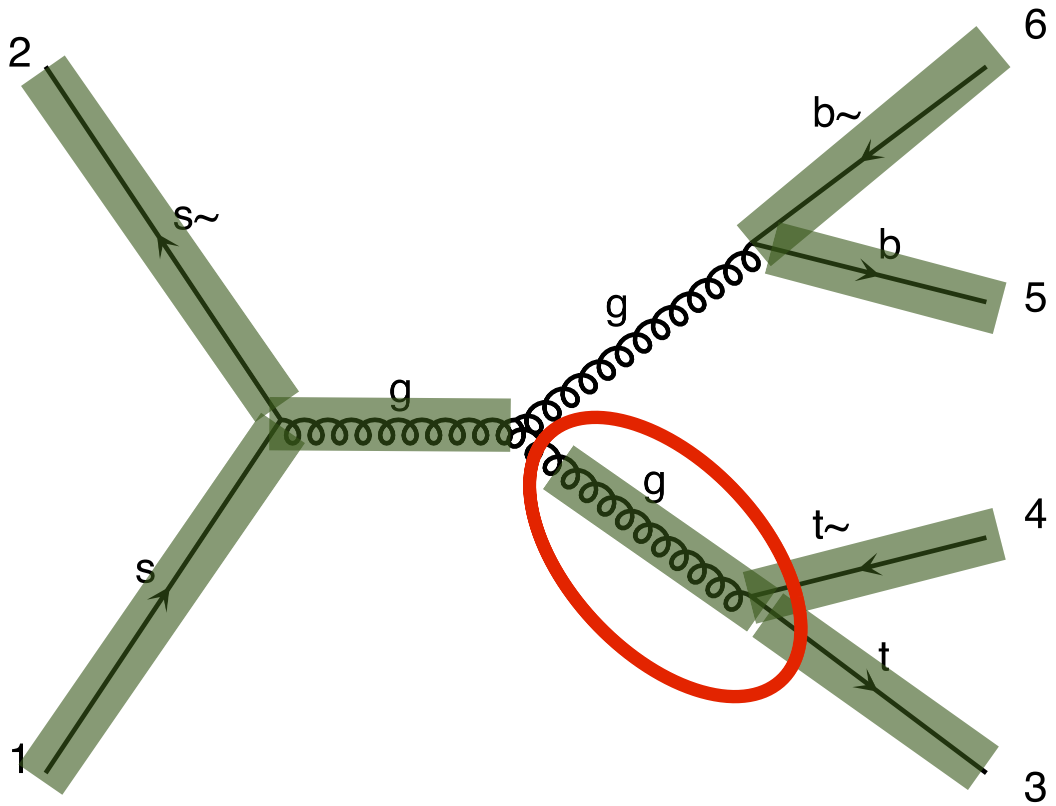


Number of routines: 7

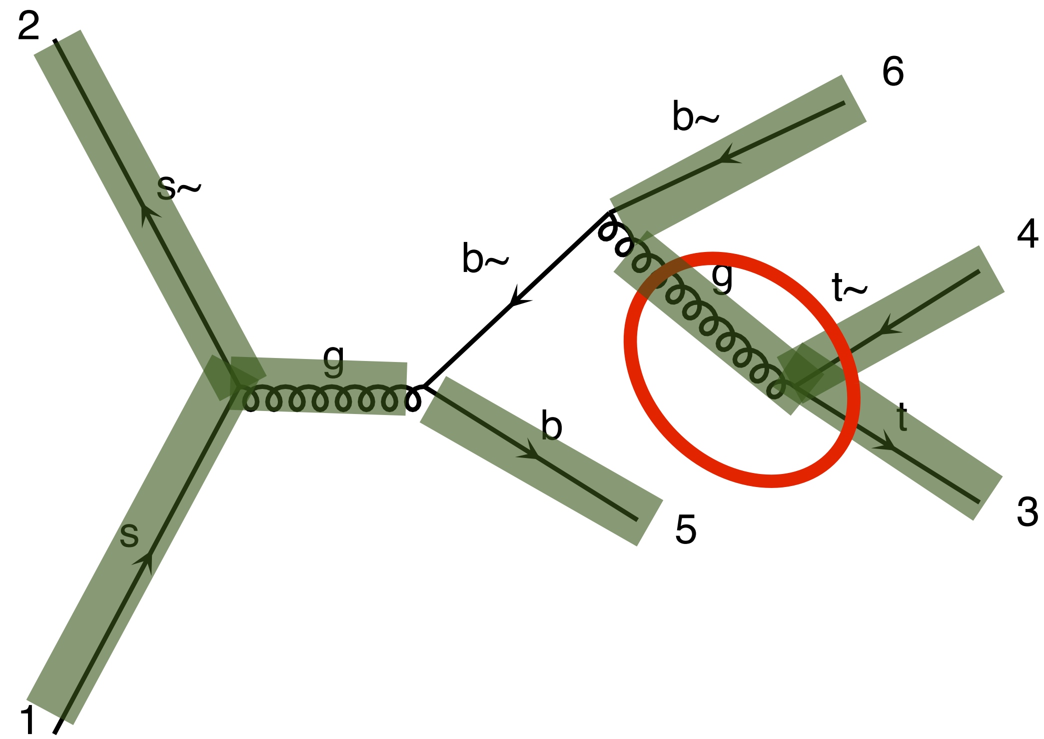
Number of routines for both: 7

Real case

Known



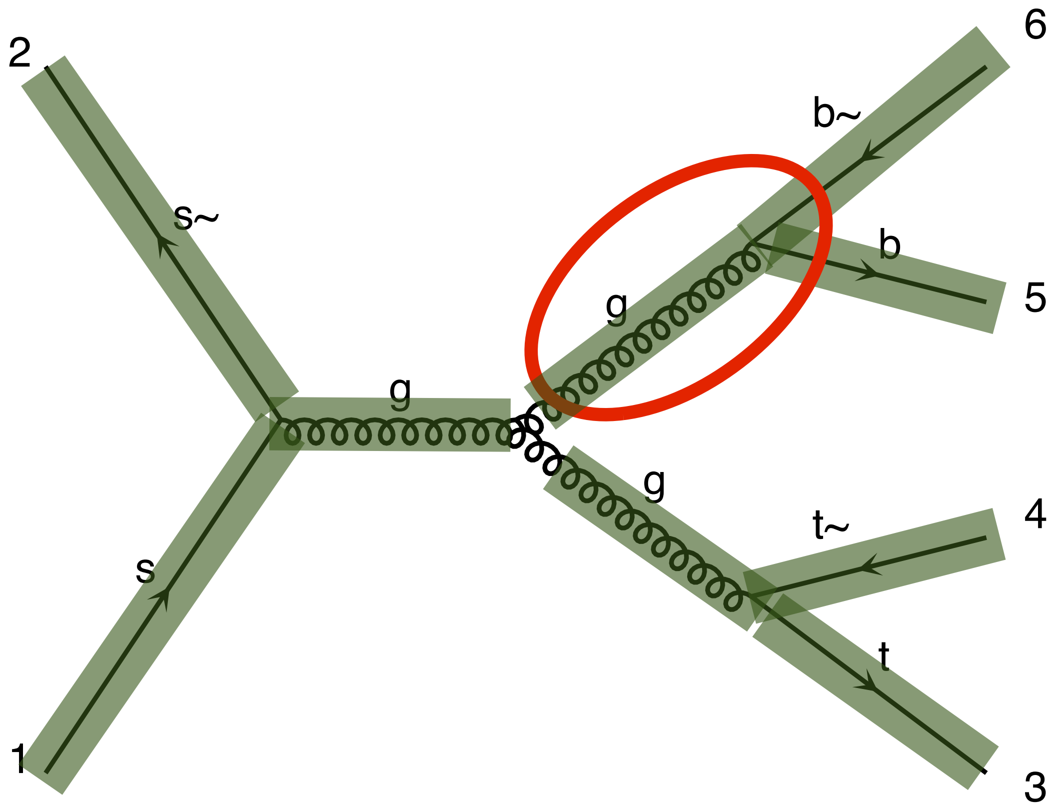
Number of routines: 8



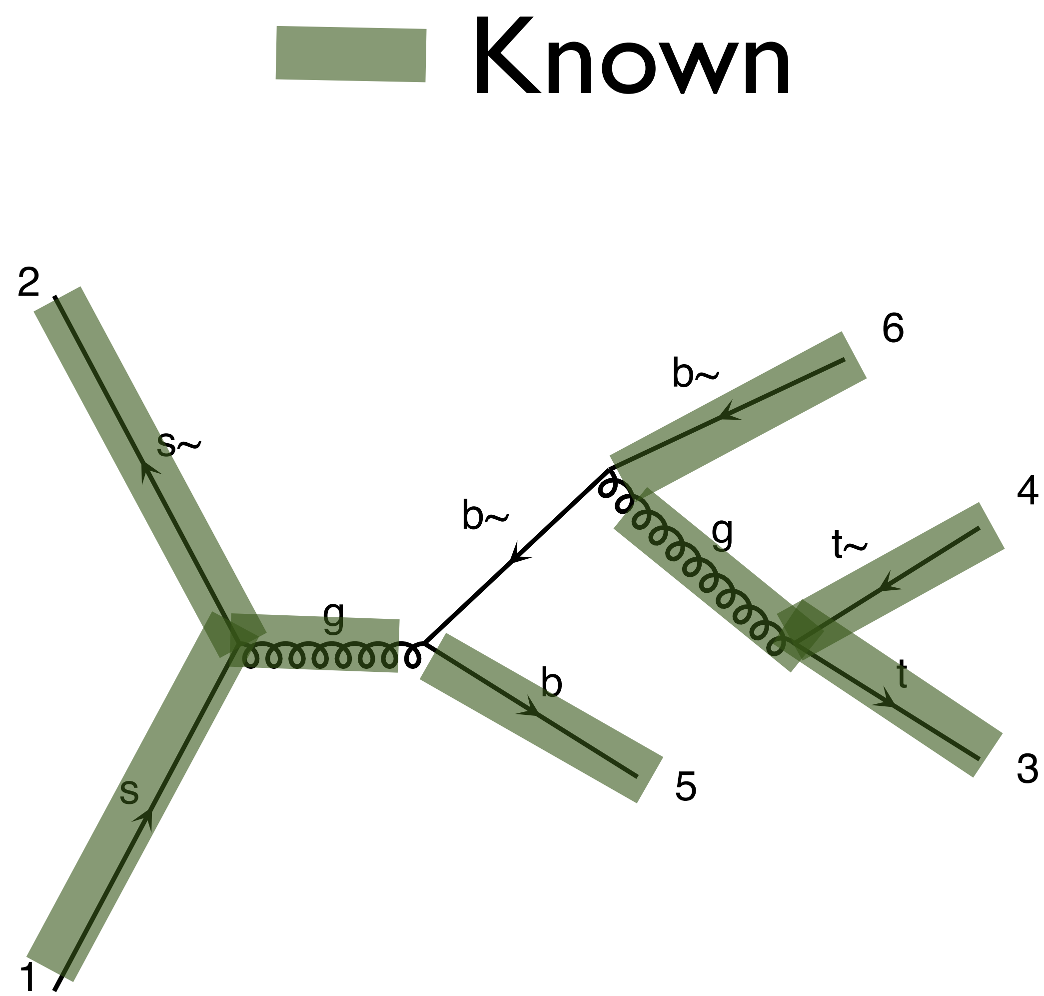
Number of routines: 8

Number of routines for both: 8

Real case



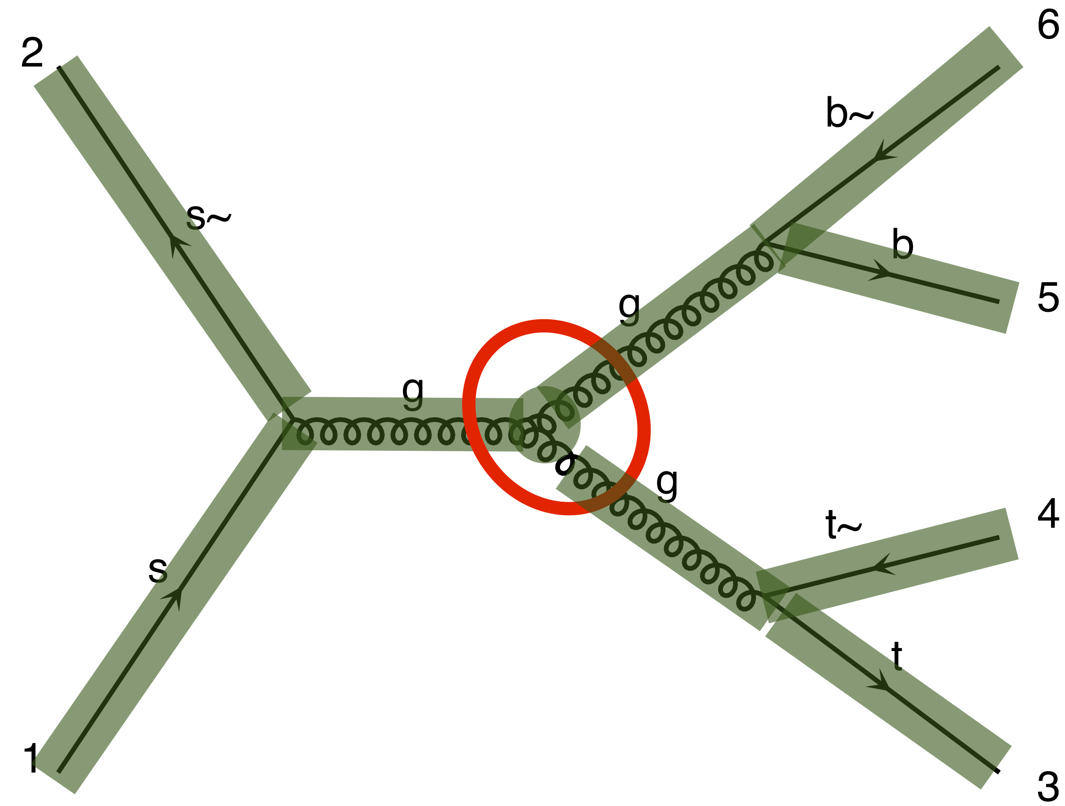
Number of routines: 9



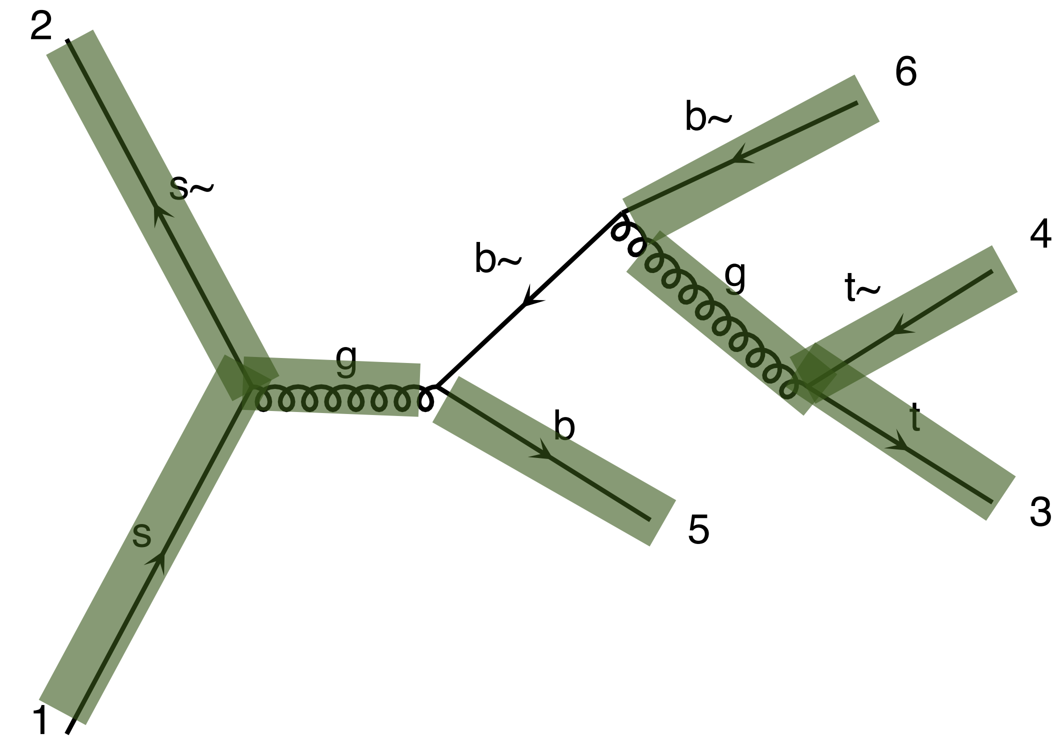
Number of routines: 8

Number of routines for both: 9

Real case



Number of routines: 10

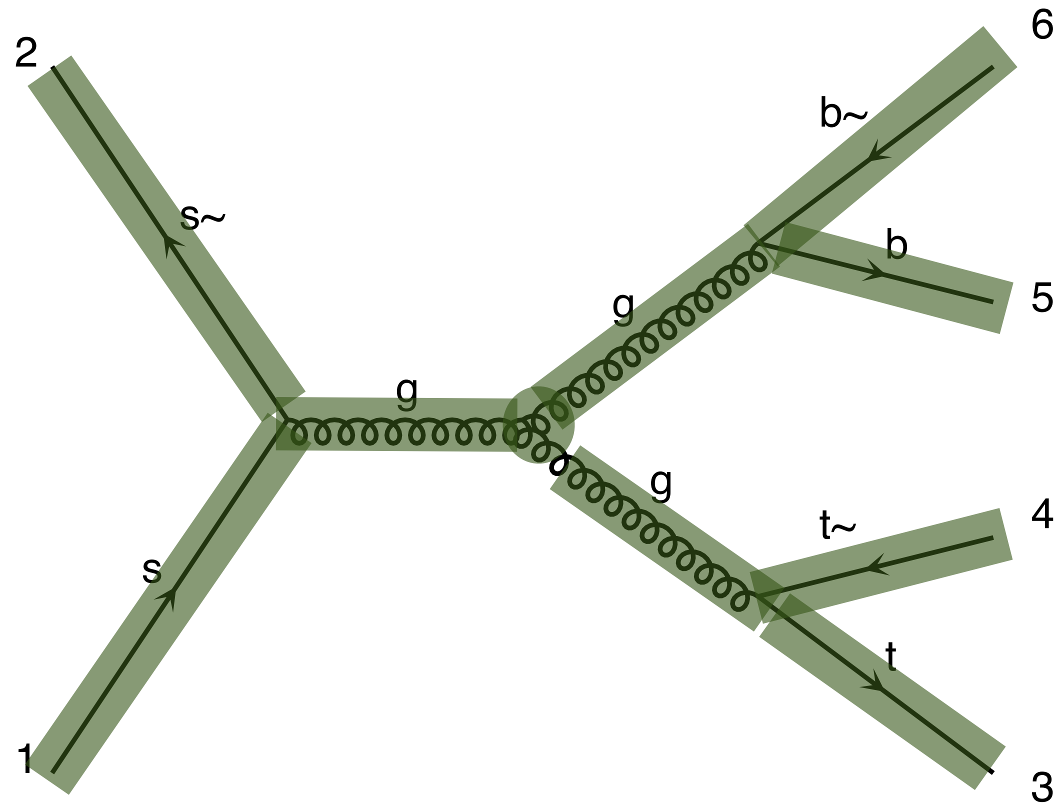


Number of routines: 8

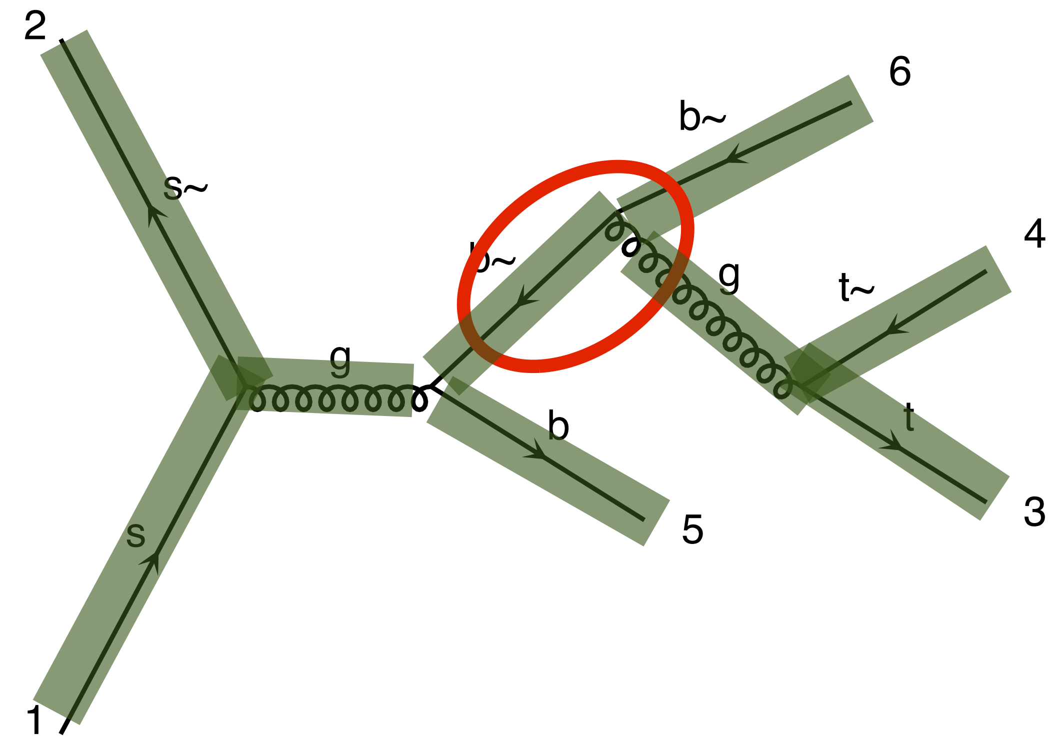
Number of routines for both: 10

Real case

Known



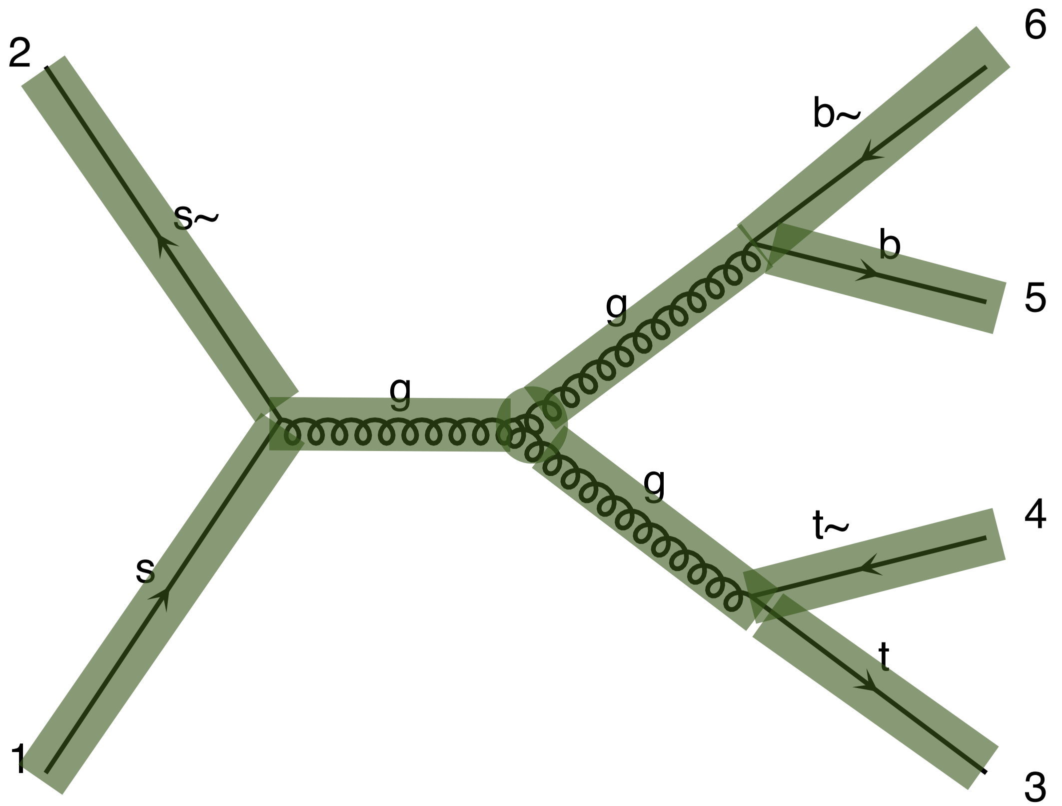
Number of routines:10



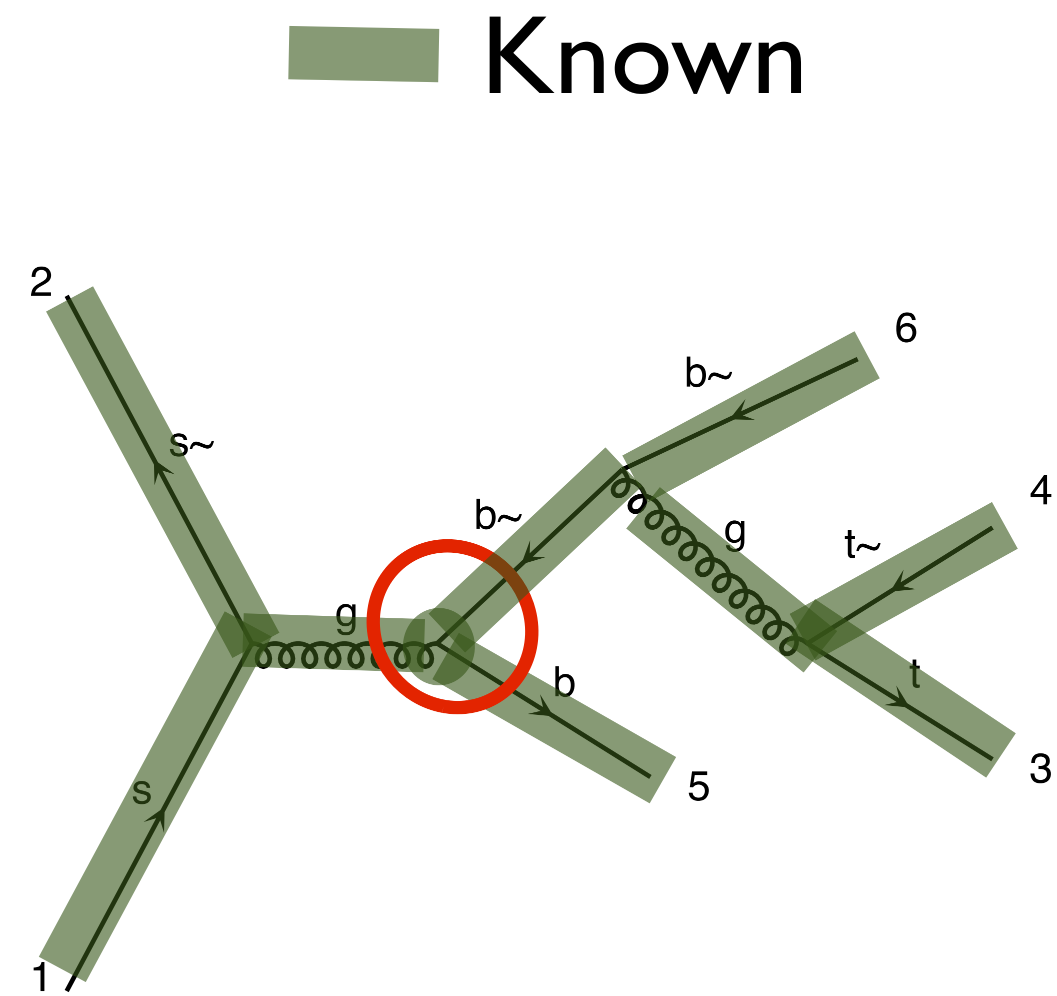
Number of routines: 9

Number of routines for both: 11

Real case



Number of routines:10



Number of routines: 10

Number of routines for both: 12

Basics: Helicity amplitudes

- Thanks to new diagram generation algorithm, wf recycling much more efficient in MG5 than MG4

Process	Amplitudes	Wavefunctions		Run time		
		MG 4	MG 5	MG 4	MG 5	no recycling
$u\bar{u} \to e^+e^-$	2	6	6	$< 6\mu s$	$< 6\mu s$	
$u\bar{u} \to e^+e^-e^+e^-$	48	62	32	0.22 ms	0.14 ms	
$u\bar{u} \to e^+e^-e^+e^-e^+e^-$	3474	3194	301	46.5 ms	19.0 ms	300,000
$u\bar{u} \to d\bar{d}$	1	5	5	$< 4\mu s$	$< 4\mu s$	
$u\bar{u} \to d\bar{d}g$	5	11	11	27 μs	27 μs	
$u\bar{u} \to d\bar{d}gg$	38	47	29	0.42 ms	0.31 ms	
$u\bar{u} \to d\bar{d}ggg$	393	355	122	10.8 ms	6.75 ms	
$u\bar{u} \to u\bar{u}gg$	76	84	40	1.24 ms	0.80 ms	
$u\bar{u} \to u\bar{u}ggg$	786	682	174	35.7 ms	17.2 ms	
$u\bar{u} \to d\bar{d}d\bar{d}$	14	28	19	84 μs	83 μs	
$u\bar{u} \to d\bar{d}d\bar{d}g$	132	178	65	1.88 ms	1.15 ms	
$u\bar{u} \to d\bar{d}d\bar{d}gg$	1590	1782	286	141 ms	34.4 ms	
$u\bar{u} \to d\bar{d}d\bar{d}d\bar{d}$	612	758	141	42.5 ms	6.6 ms	5500

Time for matrix element evaluation on a Sony Vaio TZ laptop

HELAS

HELAS

- Original HELicity Amplitude Subroutine library

[Murayama, Watanabe, Hagiwara]

HELAS

- Original HELicity Amplitude Subroutine library
  [Murayama, Watanabe, Hagiwara]

- One routine per Lorentz structure
  - ➡ MSSM [cho, al] hep-ph/0601063 (2006)
  - ➡ HEFT [Frederix] (2007)
  - ➡ Spin 2 [Hagiwara, al] 0805.2554 (2008)
  - ➡ Spin 3/2 [Mawatari, al] 1101.1289 (2011)

HELAS

- Original HELicity Amplitude Subroutine library
  [Murayama, Watanabe, Hagiwara]

- One routine per Lorentz structure
  - ➡ MSSM [cho, al] hep-ph/0601063 (2006)
  - ➡ HEFT [Frederix] (2007)
  - ➡ Spin 2 [Hagiwara, al] 0805.2554 (2008)
  - ➡ Spin 3/2 [Mawatari, al] 1101.1289 (2011)

Chiral Perturbation

BNV Model

Effective Field Theory

NMSSM

Full HEFT

Chromo-magnetic operator

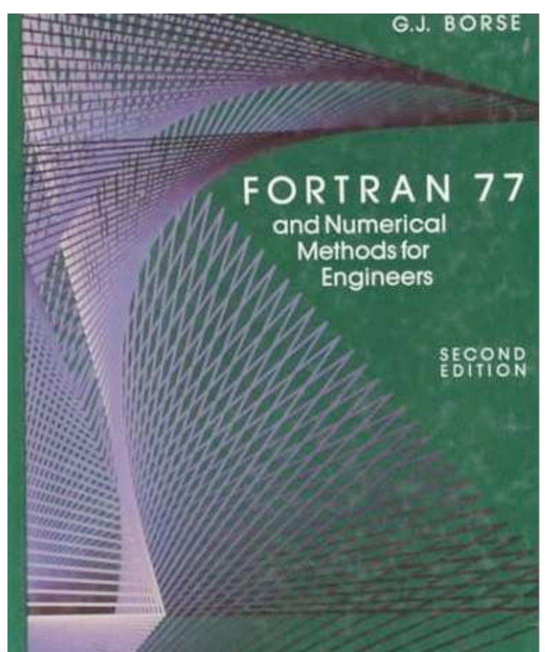
Black Holes

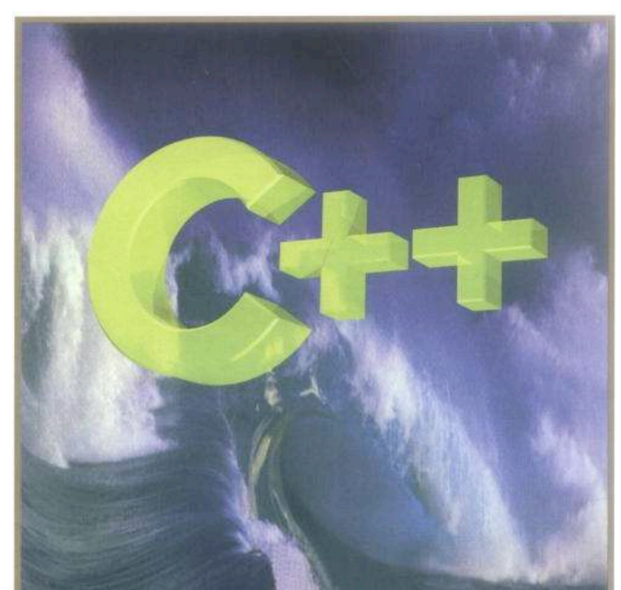
ALOHA

ALOHA
~~Google~~ translate

From: UFO ▾ To: Helicity [Translate]

Type text or a website address or translate a document.

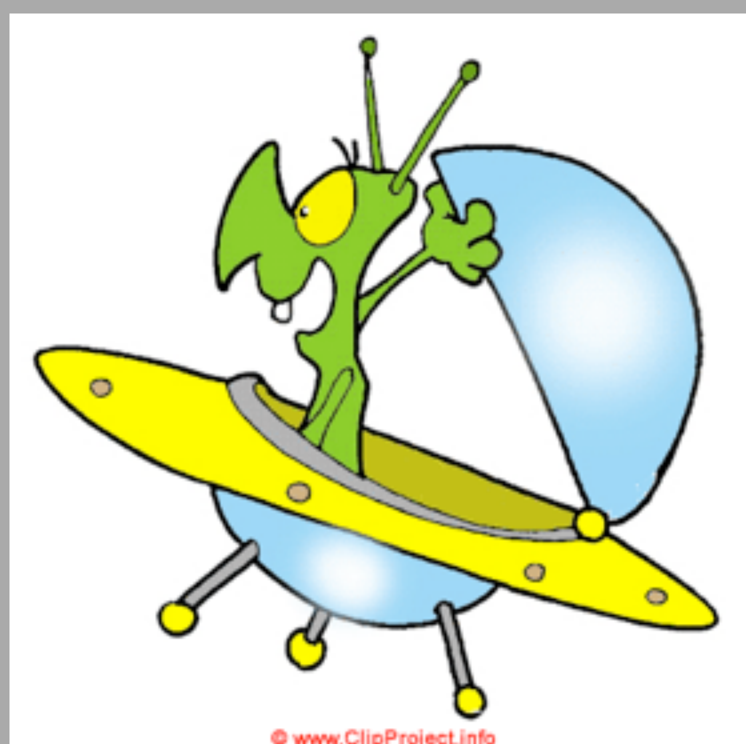




ALOHA

ALOHA
~~Google~~ translate

From: UFO To: Helicity Translate



- FeynRules output
- New Model Format
- Gosam/ Herwig++/ MG5
- Fully generic color/Lorentz/…

[Degrande, Duhr, Fuks, Grellscheid, OM, Reiter:108.2040]

Type t

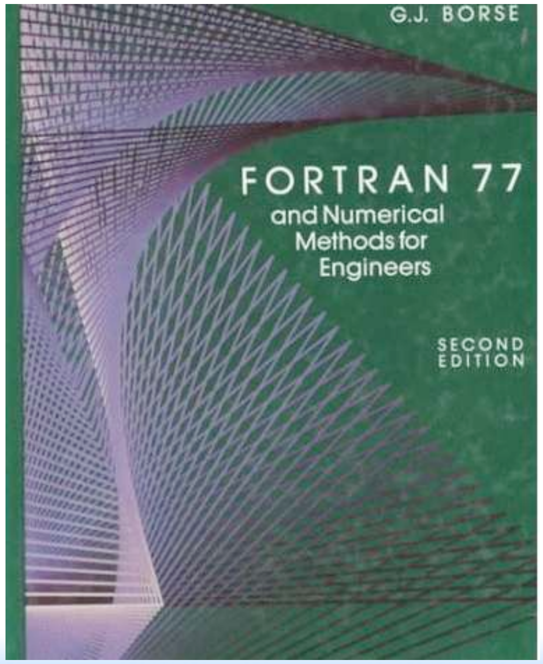
WESLEY J. CHUN

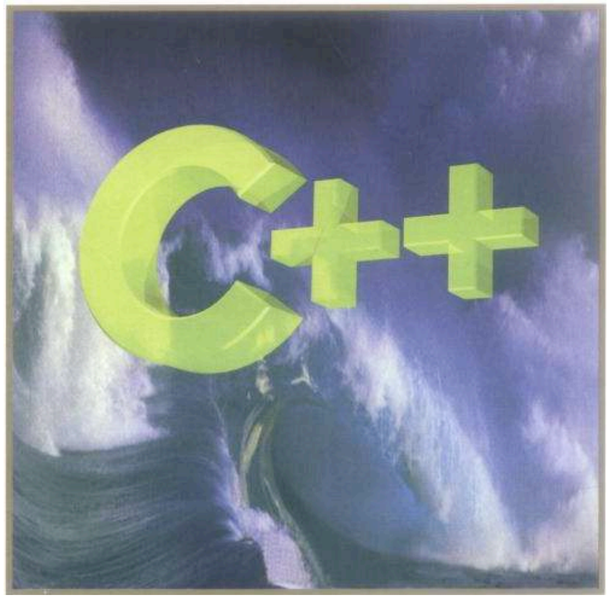
ALOHA

ALOHA ~~Google~~ translate

From: UFO To: Helicity Translate

Type text or a website address or translate a document.

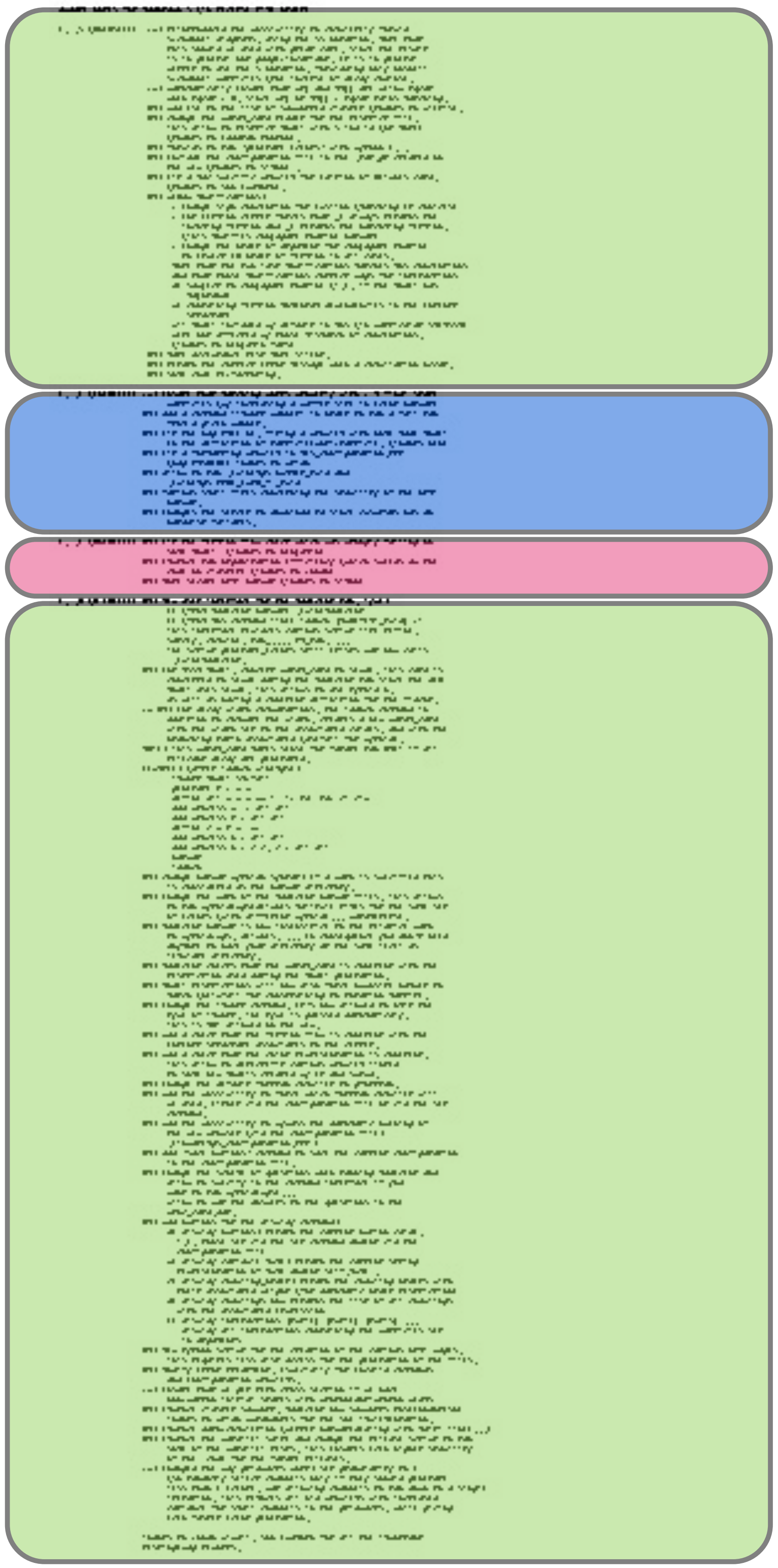




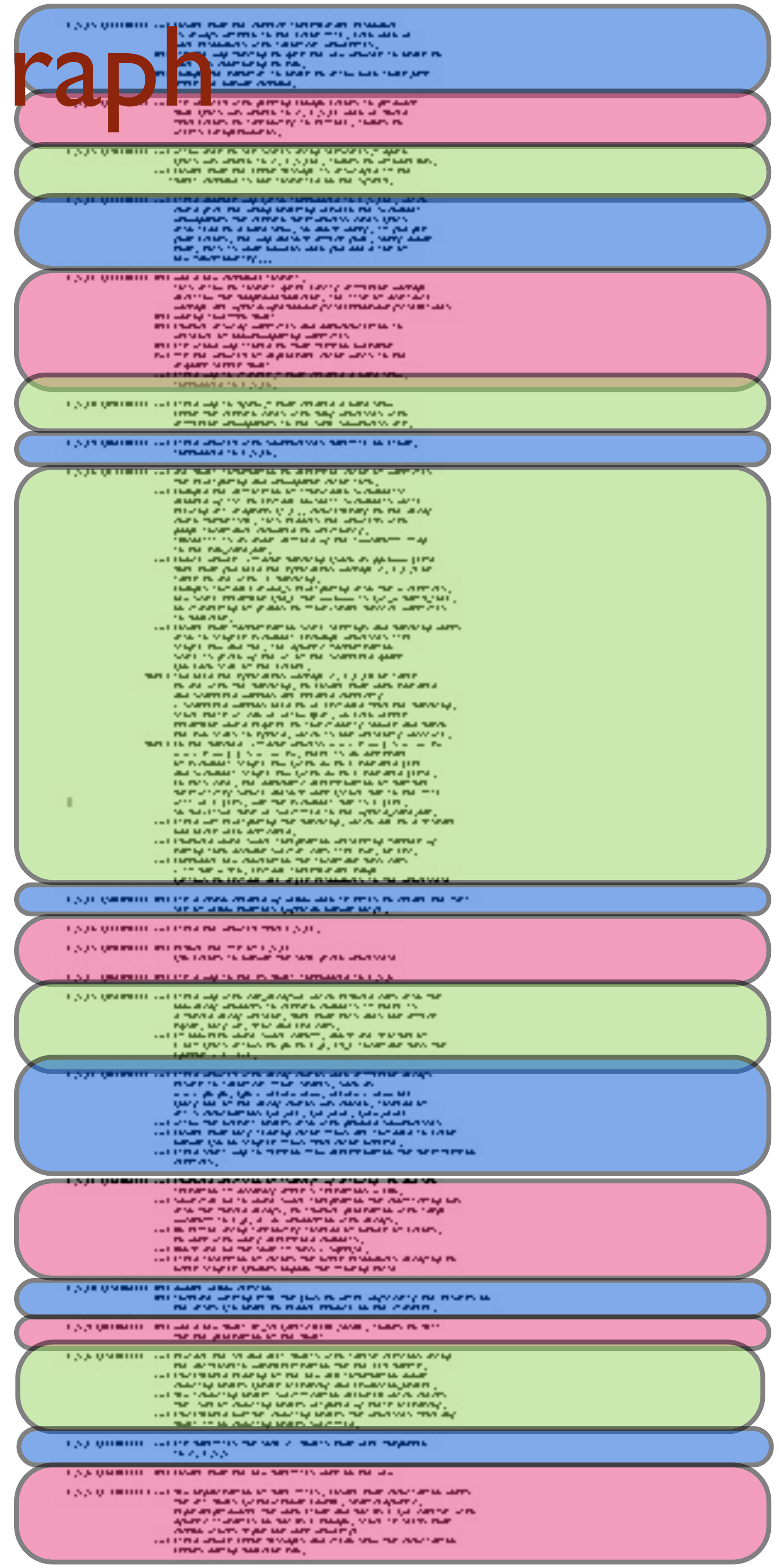
MADGRAPH 5

MadGraph

MadGraph



1.4.0



MadGraph

Not possible to detail everything

1.4.0

MadGraph

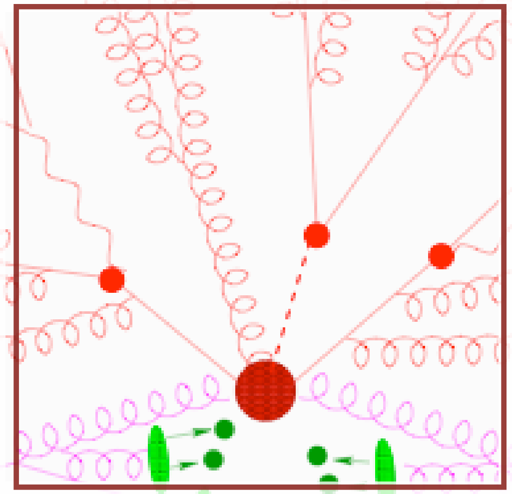
- Original MadGraph by Tim Stelzer was written in Fortran, first version from 1994

hep-ph/9401258

1. High-Q^2 Scattering

2. Parton Shower

☞ where new physics lies



☞ process dependent

☞ first principles description

3. Hadronization

4. Underlying Event

MadGraph

- Original MadGraph by Tim Stelzer was written in Fortran, first version from 1994 hep-ph/9401258
- Event generation by MadEvent using the single diagram enhanced multichannel integration technique in 2002 (Stelzer, Maltoni) hep-ph/0208156

Master formula

$$\int \hat{\sigma}_{ab\to X}(\hat{s},\ldots)\, f_a(x_1) f_b(x_2)\, dx_1 dx_2 d\Phi_{FS}$$

Parton level cross section | Parton density functions | Phase space integral

- Parton level cross section from matrix element
- Parton density (or distribution) functions:
  Process independent, determined by particle type

MadGraph

MadGraph

- Original MadGraph by Tim Stelzer was written in Fortran, first version from 1994

hep-ph/9401258

MadGraph

- Original MadGraph by Tim Stelzer was written in Fortran, first version from 1994 hep-ph/9401258
- Event generation by MadEvent using the single diagram enhanced multichannel integration technique in 2002 (Stelzer, Maltoni) hep-ph/0208156

MadGraph

- Original MadGraph by Tim Stelzer was written in Fortran, first version from 1994 hep-ph/9401258
- Event generation by MadEvent using the single diagram enhanced multichannel integration technique in 2002 (Stelzer, Maltoni) hep-ph/0208156
- Support for BSM (and many other improvements) in MG/ME 4 (2006) arXiv:0706.2334, arXiv:0809.2410

MadGraph

- Original MadGraph by Tim Stelzer was written in Fortran, first version from 1994 hep-ph/9401258
- Event generation by MadEvent using the single diagram enhanced multichannel integration technique in 2002 (Stelzer, Maltoni) hep-ph/0208156
- Support for BSM (and many other improvements) in MG/ME 4 (2006) arXiv:0706.2334, arXiv:0809.2410
- Rewritten in Python in 2011: MG5 arXiv:1106.0522
  - ➡ Full automatic support for BSM

MadGraph

- Original MadGraph by Tim Stelzer was written in Fortran, first version from 1994 hep-ph/9401258
- Event generation by MadEvent using the single diagram enhanced multichannel integration technique in 2002 (Stelzer, Maltoni) hep-ph/0208156
- Support for BSM (and many other improvements) in MG/ME 4 (2006) arXiv:0706.2334, arXiv:0809.2410
- Rewritten in Python in 2011: MG5 arXiv:1106.0522
  - ➡ Full automatic support for BSM
- First public version of aMC@NLO in 2013 (See more tomorrow!)

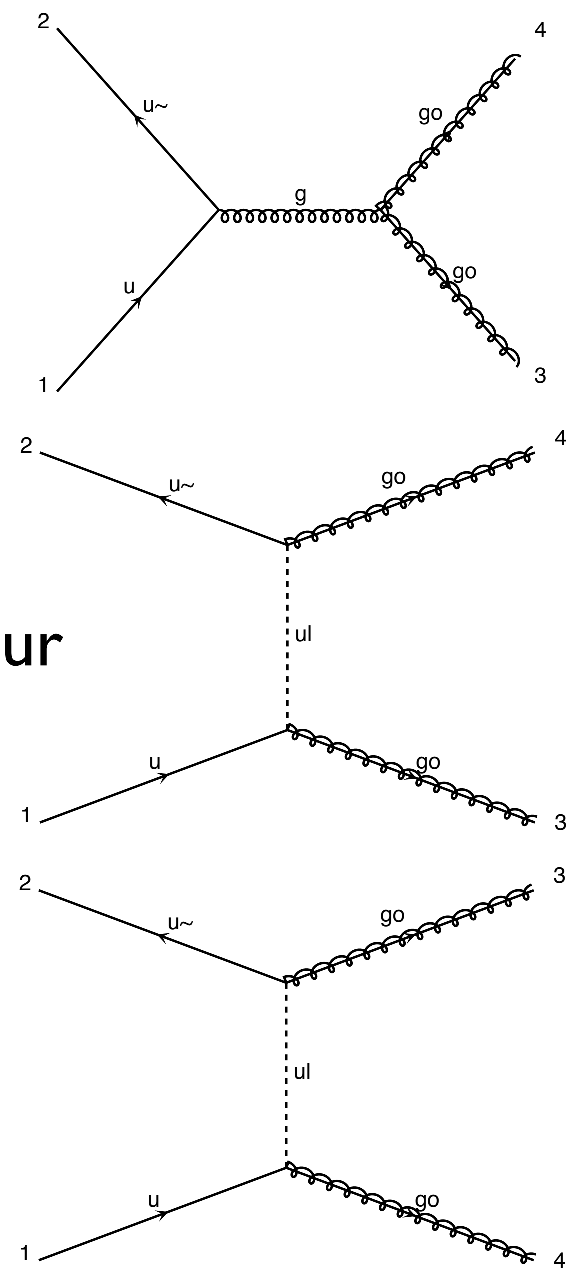
Decay chains

- p p > t t~ w+, (t > w+ b, w+ > l+ vl), \
  (t~ > w- b~, w- > j j), \
  w+ > l+ vl
- Separately generate core process and each decay
  - Decays generated with the decaying particle as resulting wavefunction
- Iteratively combine decays and core processes
- Difficulty: Multiple diagrams in decays

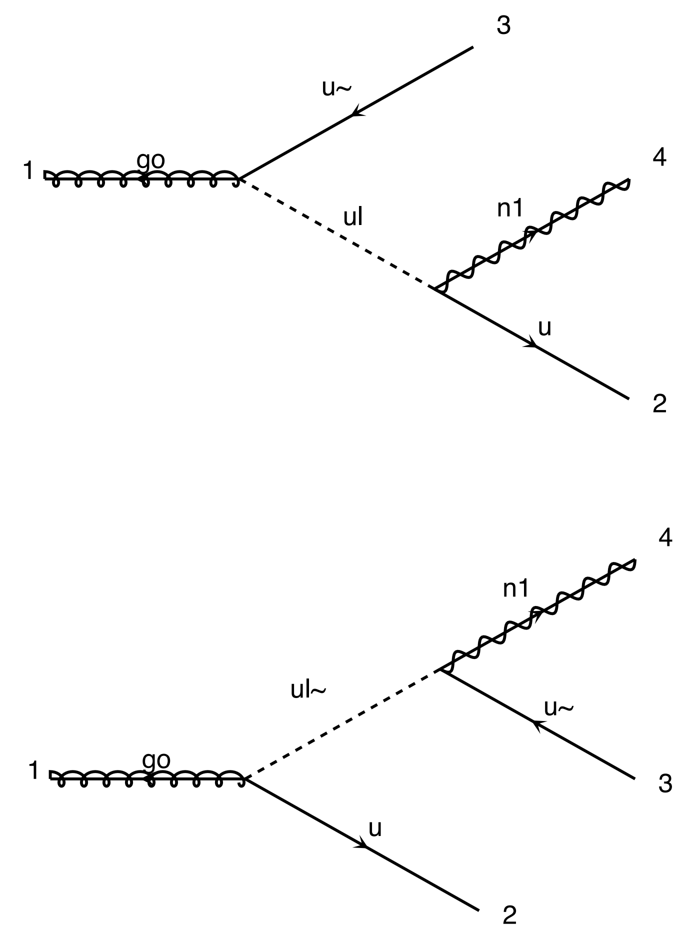
Decay chains

- If multiple diagrams in decays, need to multiply together core process and decay diagrams:

u u~ > go go / ur



x



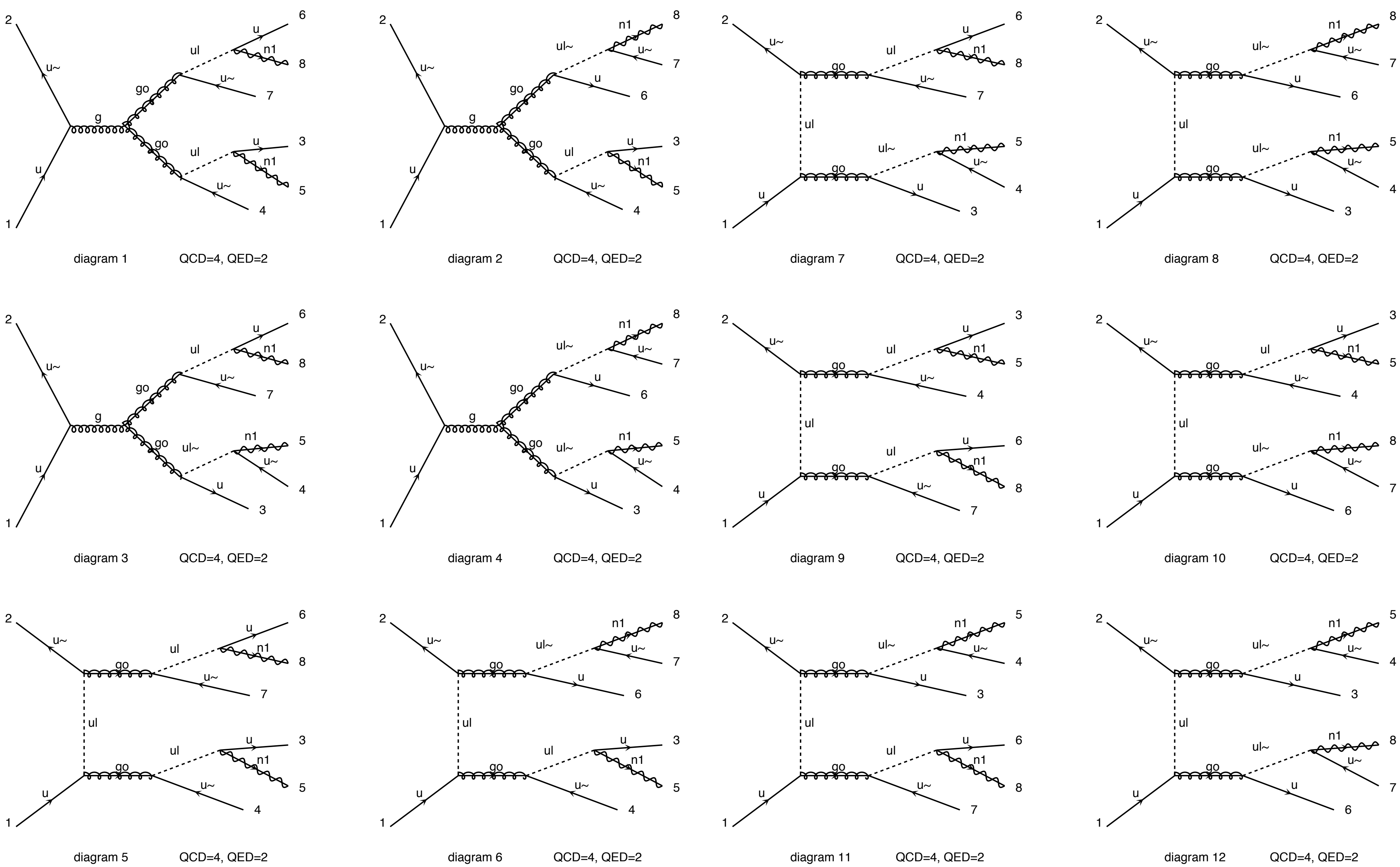
go > u u~ n1 / ur

(to the second power since both gluinos decay)

Decay chains

- If multiple diagrams in decays, need to multiply together core process and decay diagrams:

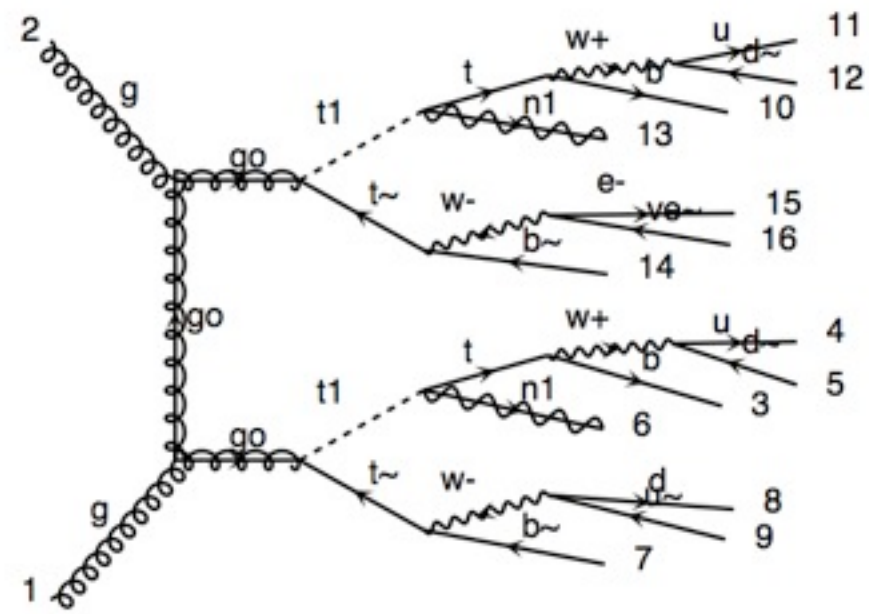
u u~ > go go / ur, go > u u~ n1 / ur



Decay chains

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

Decay chains



Online Event Generation

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

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

Available Results

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

Main Page

Thanks to developments in MadEvent, also (very) long decay chains possible to simulate directly in MadGraph!

Output formats in MadGraph 5

- Thanks to UFO/ALOHA, we now have automatic helicity amplitude routines in any language
  - ➡ So it makes sense to have also matrix element output in multiple languages!
- Presently implemented: Fortran, C++, Python
  - ➡ Fortran - for MadEvent and Standalone
  - ➡ C++ - for Pythia 8 and Standalone
  - ➡ Python - for internal use in MG5 (checks of gauge, perturbation and Lorentz invariance)

Pythia 8 Matrix Element output

- ➡ Library of process .h and .cc files, sorted by model
  + all needed model and helicity amplitude files
  + example main file (for user convenience!)
- ➡ Run as standard internal Pythia processes
- ➡ Allows using Pythia for ANY (2→1,2,3) process in ANY model at the push of a key!

Sigma_sm_qq_ttx.h

```
#include "SigmaProcess.h"
#include "Parameters_sm.h"

using namespace std;

namespace Pythia8
{
//==========================================================
// A class for calculating the matrix elements for
// Process: u u~ > t t~
// Process: c c~ > t t~
// Process: d d~ > t t~
// Process: s s~ > t t~
//----------------------------------------------------------
class Sigma_sm_qq_ttx : public Sigma2Process
{
  public:

    // Constructor.
    Sigma_sm_qq_ttx() {}

    // Initialize process.
    virtual void initProc();

    // Calculate flavour-independent parts of cross section.
    virtual void sigmaKin();

    // Evaluate sigmaHat(sHat).
    virtual double sigmaHat();

    // Select flavour, colour and anticolour.
    virtual void setIdColAcol();
 ...
```

MADGRAPH 5
Life Demonstration

Examples shown

- p p > t t~
  This gives only (the dominant) QCD vertices, and ignores (the negligible) QED vertices.
- p p > t t~ QED=2
  This gives both QED and QCD vertices.
- p p > w+ j j, w+ > l+ vl
  More complicated example.

More syntax examples

- p p > t t~ j QED=2: Generate all combinations of processes for particles defined in multiparticle labels p / j, including up to two QED vertices (and unlimited QCD vertices)
- p p > t t~, (t > b w+, w+ > l+ vl), t~ > b~ j j :
  - Only diagrams compatible with given decay
  - Only t / t~ and W+ close to mass shell in event generation
- p p > w+ w- / h : Exclude any diagrams with h
- p p > w+ w- $ h : Exclude on-shell h in event generation (but retain interference effects)

Summary: Simulations with MG5

- UFO + ALOHA + MG5:
  - ➡ ANY BSM is available
  - ➡ HELAS Routines ⇒ very fast
- MG5
  - ➡ decay chains
  - ➡ nice interface
  - ➡ several output formats
  - ➡ easy to use

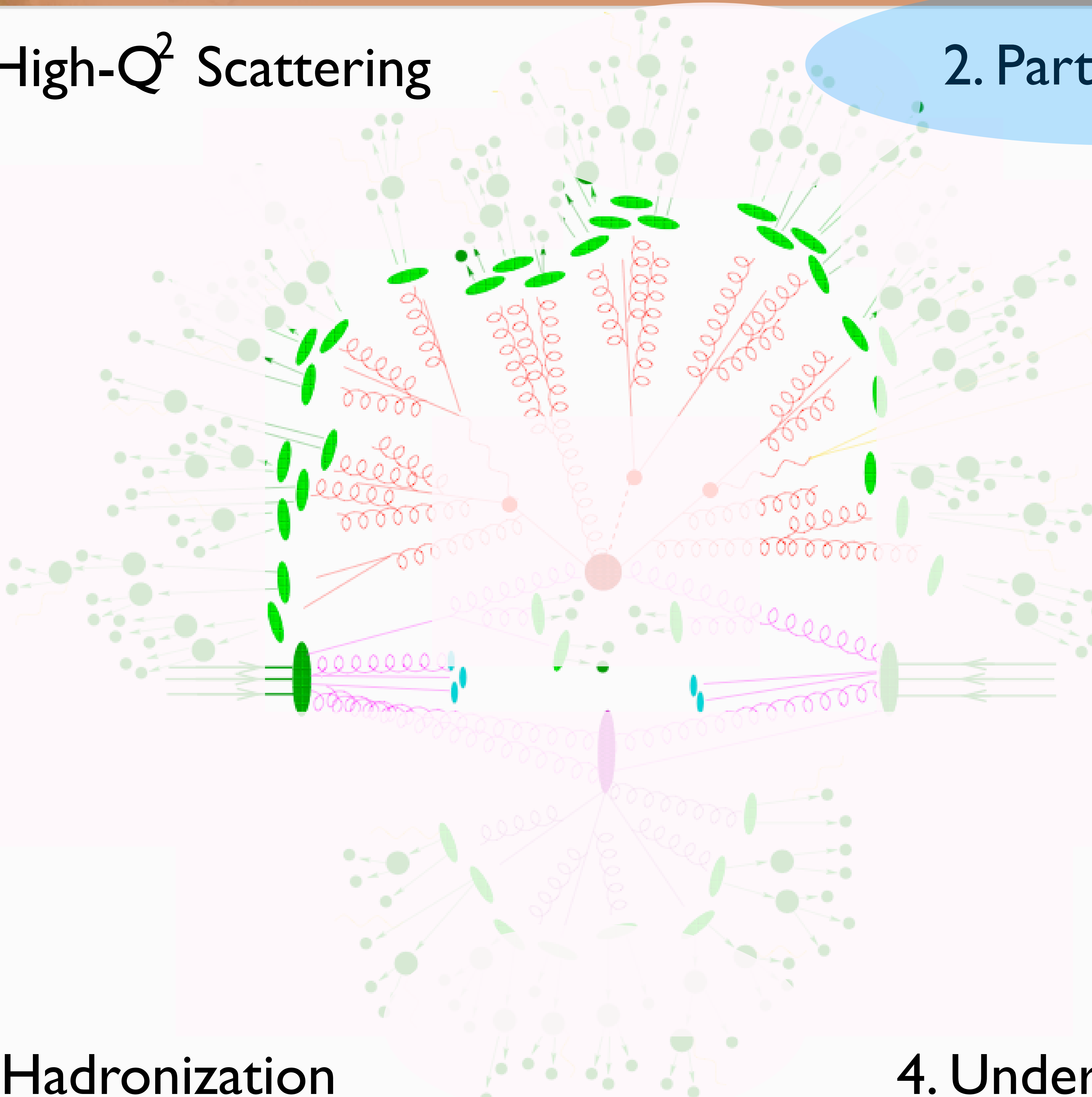
Jet Matching

Outline:

- Parton Showers
- MLM Matching with MadGraph and Pythia
- Validating the Matching

1. High-Q^2 Scattering

2. Parton Shower



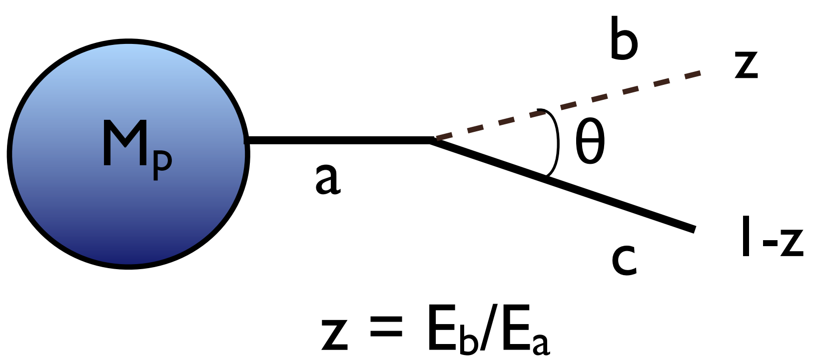
3. Hadronization

4. Underlying Event

Parton Shower basics

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

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

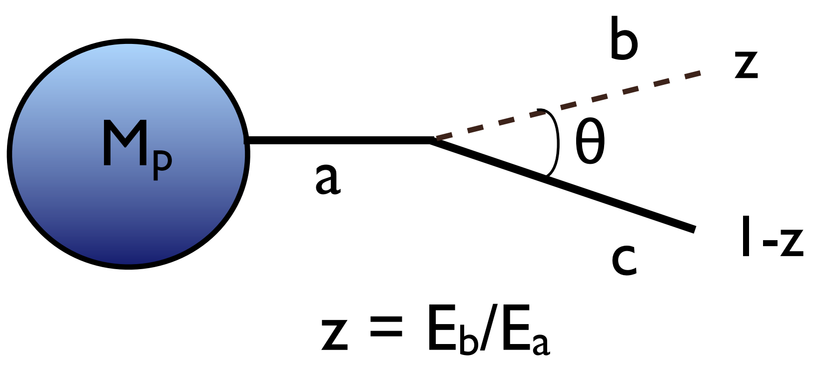


Parton Shower basics

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

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

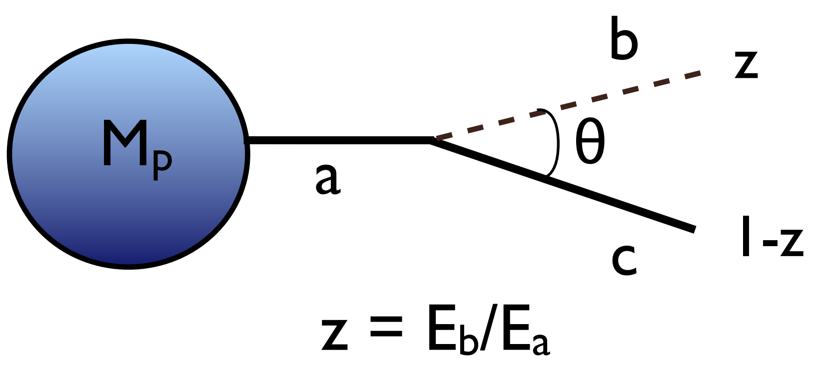
soft



Parton Shower basics

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

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

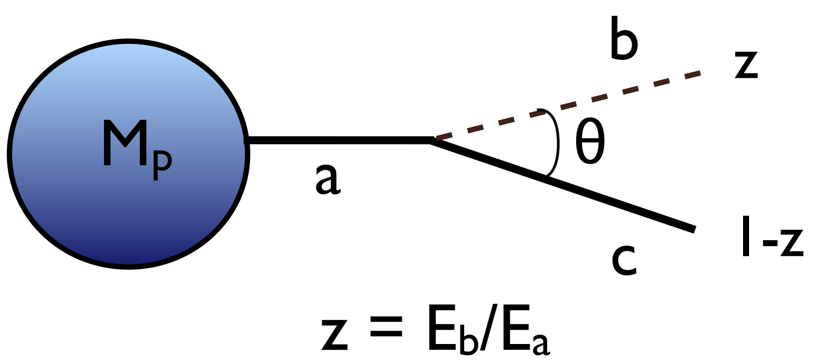


soft and collinear divergencies

Parton Shower basics

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

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



soft and collinear divergencies

Collinear factorization:

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

when θ is small.

Parton Shower basics

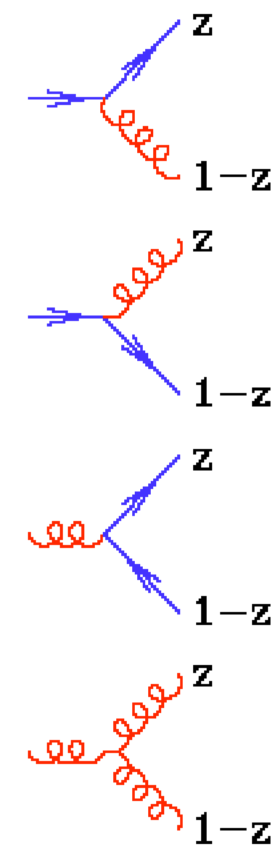
The spin averaged (unregulated) splitting functions for the various types of branching are:

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

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

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

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



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

Parton Shower basics

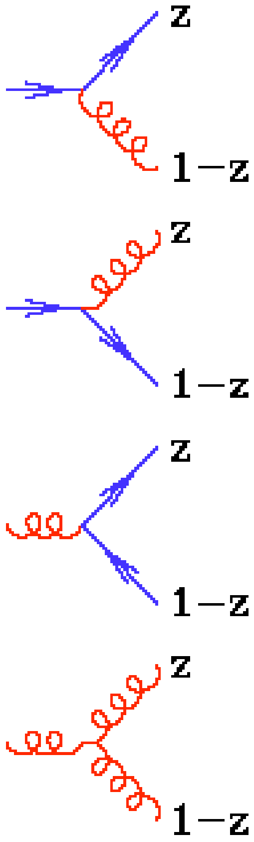
The spin averaged (unregulated) splitting functions for the various types of branching are:

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

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

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

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

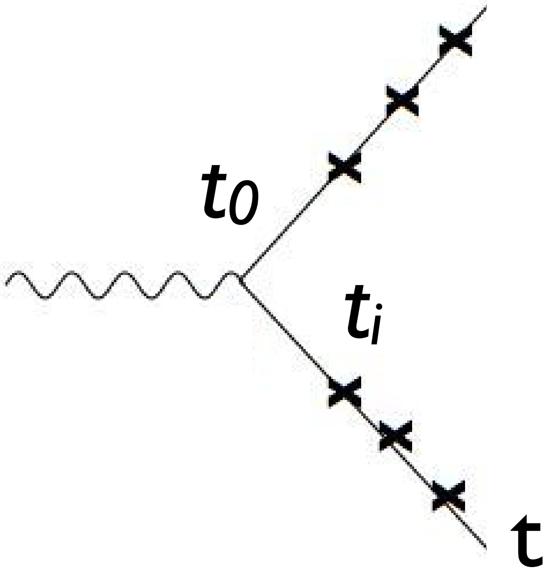


$$C_F = \frac{4}{3}, C_A = 3, T_R = \frac{1}{2}.$$

Comments:

* Gluons radiate the most
* There are soft divergences in z=1 and z=0.
* P_{qg} has no soft divergences.

Parton Shower basics



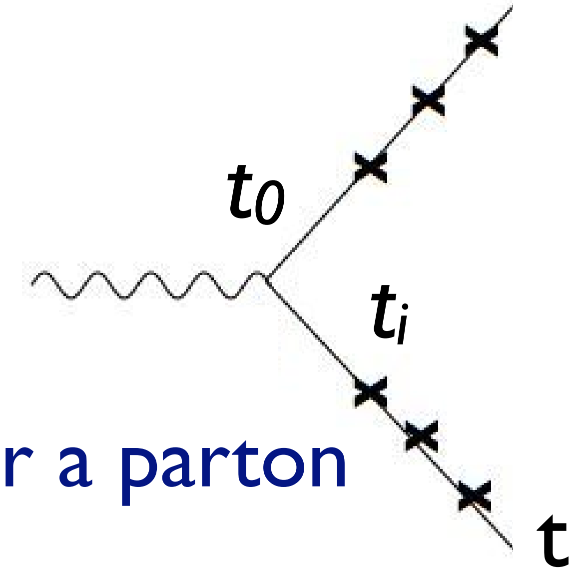
\mathcal{
{\delta t

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

{\color[rgb
{\delta t}{

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

Parton Shower basics



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

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

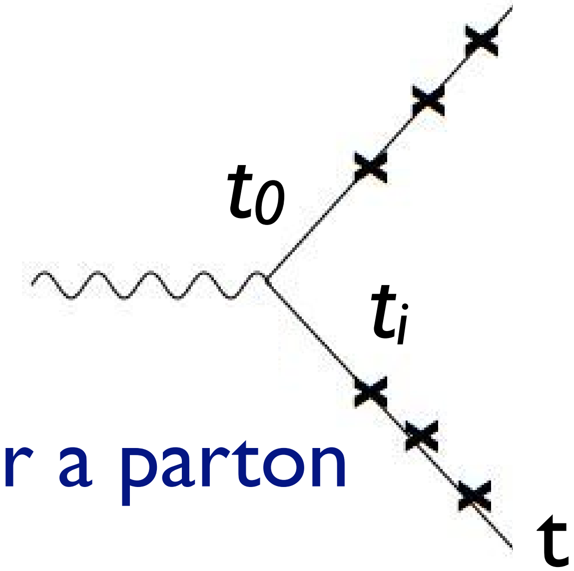
\mathcal{P
\delta t}

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

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

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

Parton Shower basics



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

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

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

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

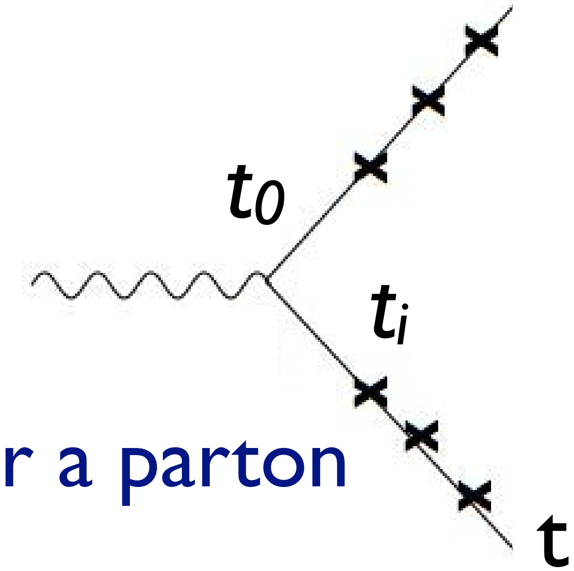
\mathcal{[
\delta t[

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

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

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

Parton Shower basics



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

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

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

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

- This is the famous "Sudakov form factor"

\mathcal{
\delta t

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

{\color[rgb
{\delta t}{

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

Final-state parton showers

Final-state parton showers

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

Final-state parton showers

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

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

Final-state parton showers

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

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

Final-state parton showers

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

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

Final-state parton showers

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

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

Final-state parton showers

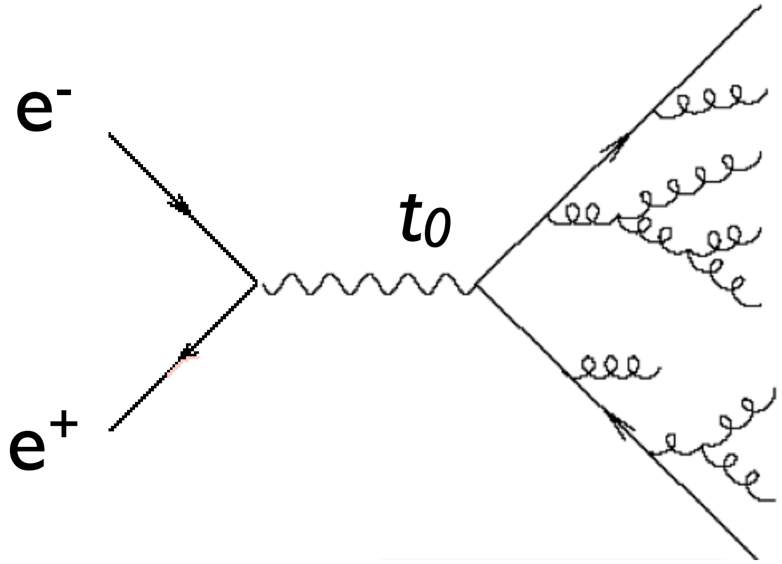
With the Sudakov form factor, we can now implement a final-state parton shower in a Monte Carlo event generator!

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

Final-state parton showers

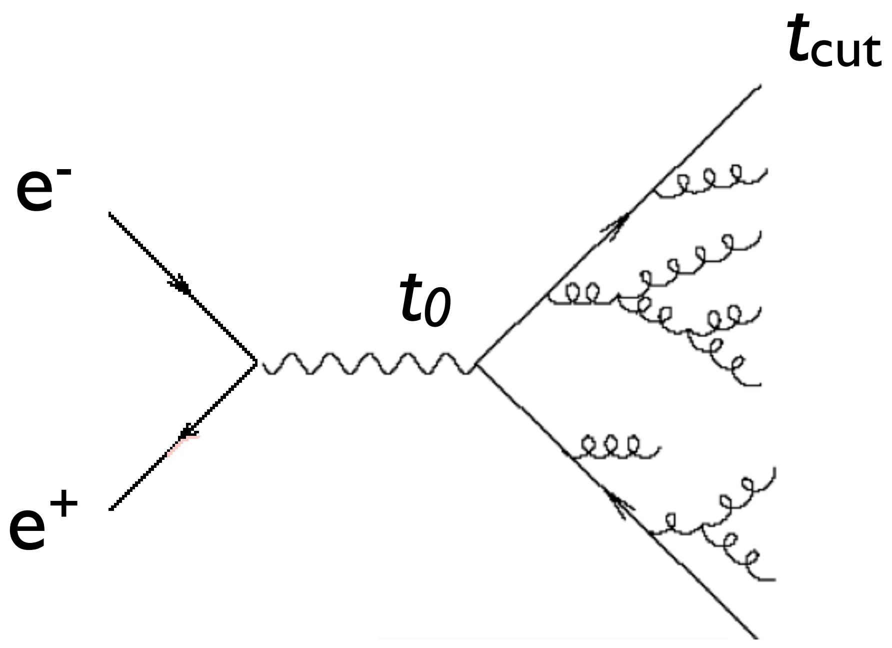
Final-state parton showers

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



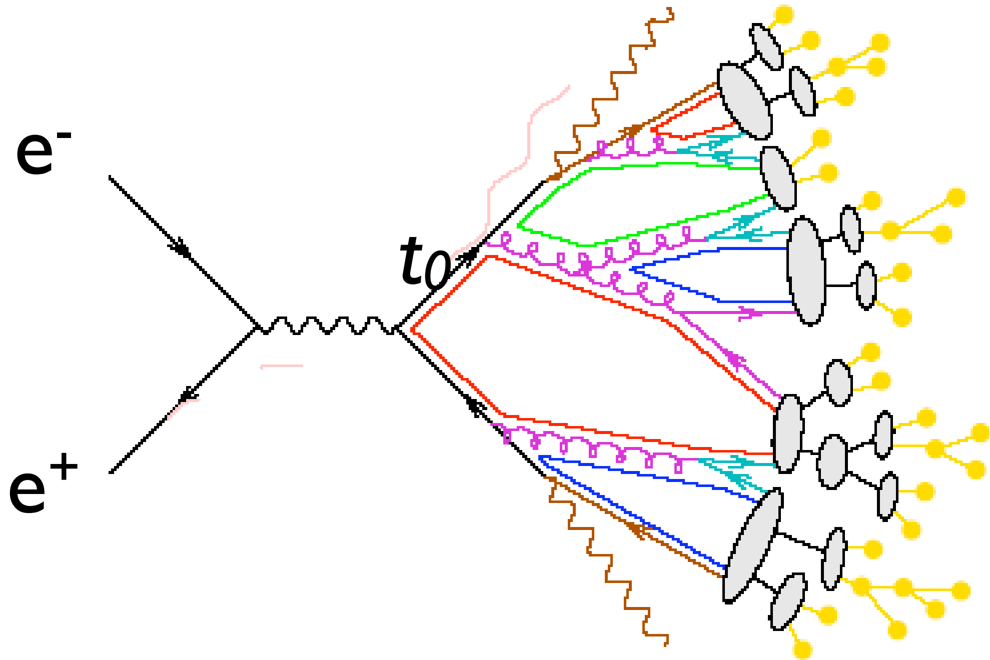
Final-state parton showers

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



Final-state parton showers

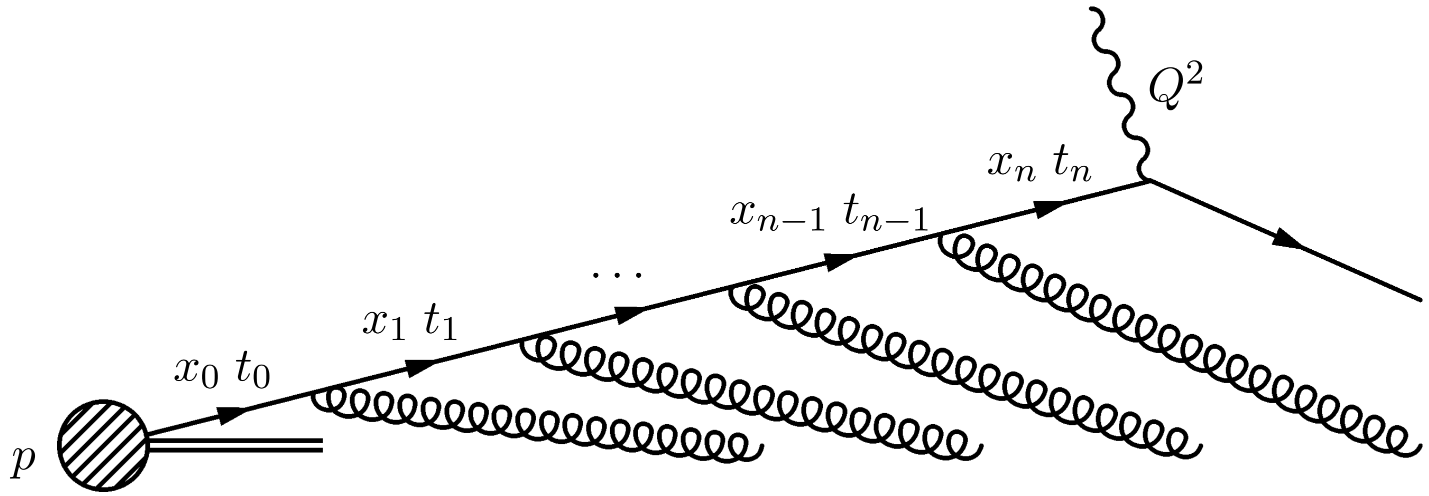
- The result is a "cascade" or "shower" of partons with ever smaller virtualities.
- The cutoff scale t_{cut} is usually set close to 1 GeV, the scale where non-perturbative effects start dominating over the perturbative parton shower.
- At this point, phenomenological models are used to simulate how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



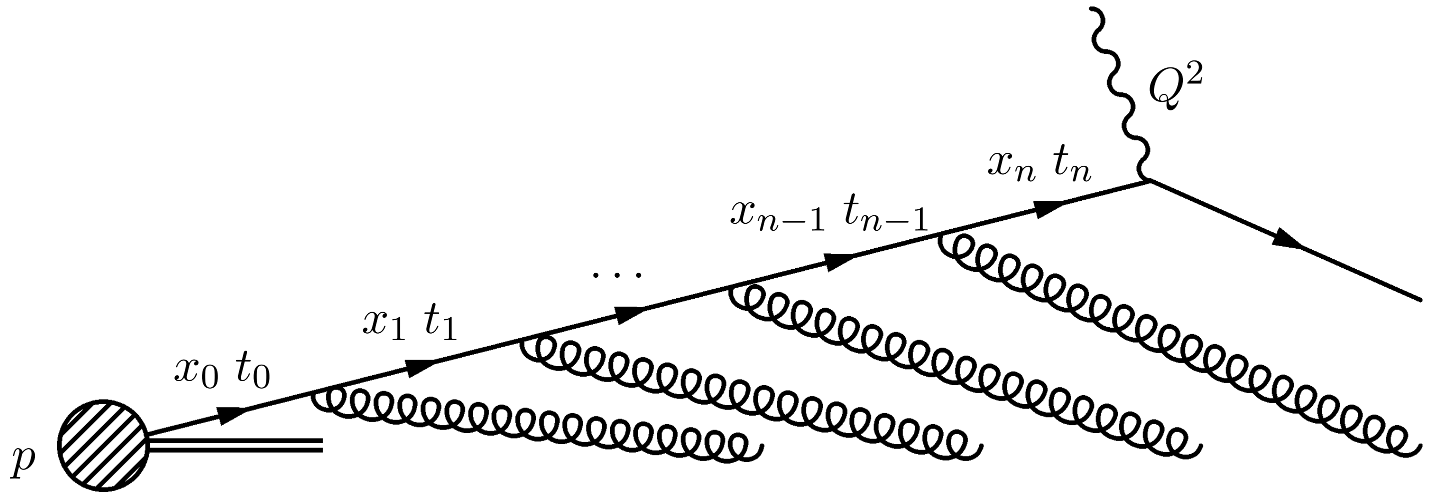
Initial-state parton splittings

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

Initial-state parton splittings



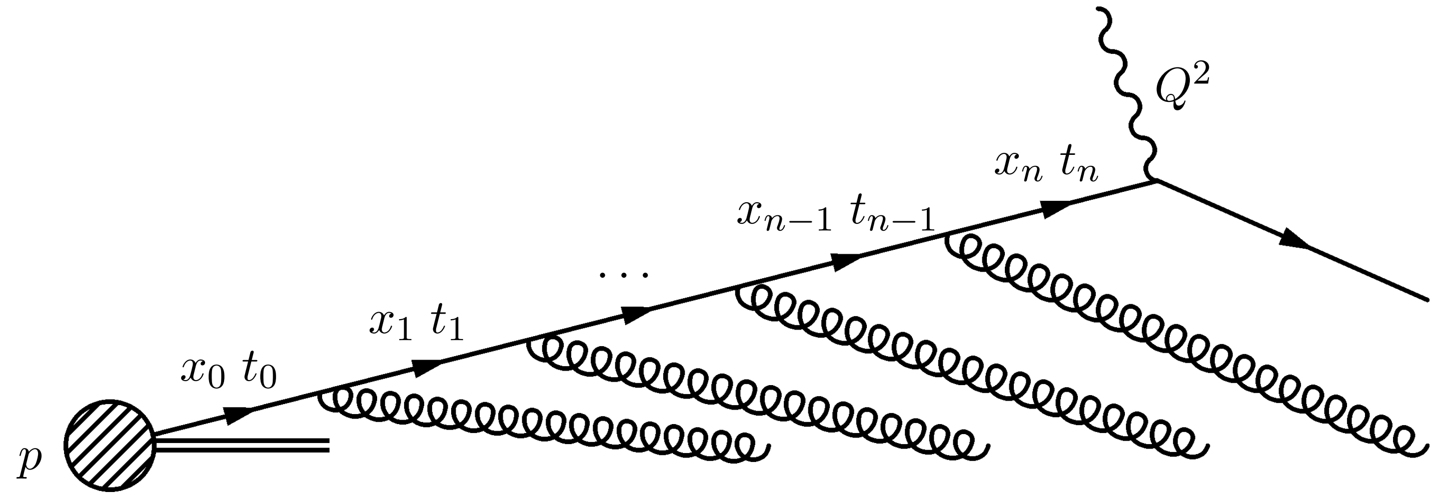
Initial-state parton splittings



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

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

Initial-state parton splittings



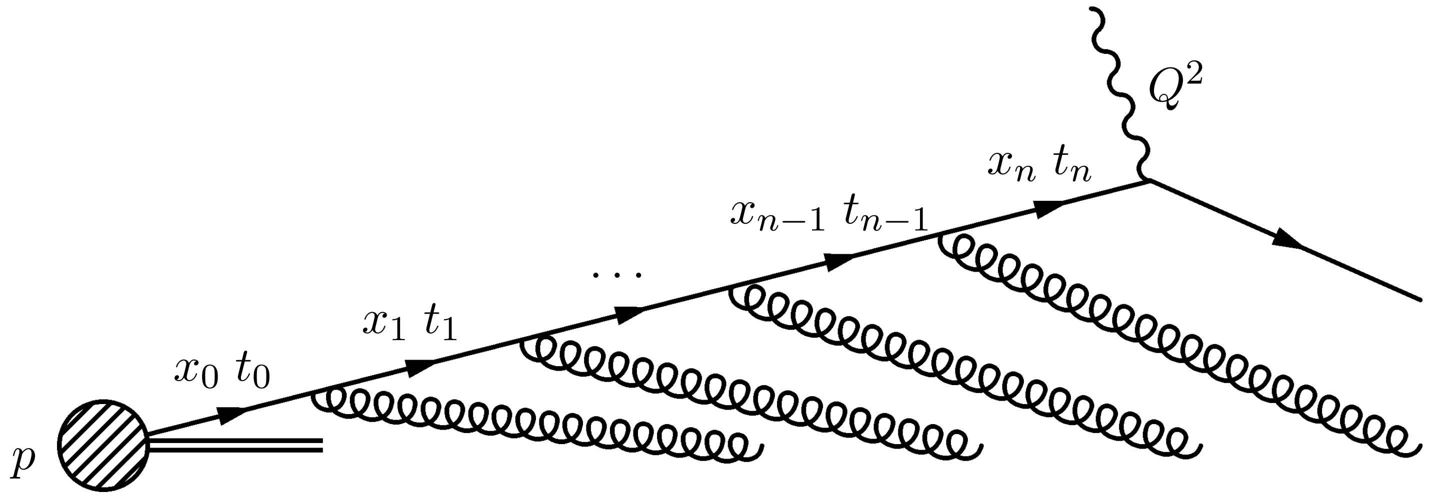
- Start with a quark PDF $f_0(x)$ at scale t_0. After a single parton emission, the probability to find the quark at virtuality $t > t_0$ is

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

- After a second emission, we have

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

Initial-state parton splittings



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

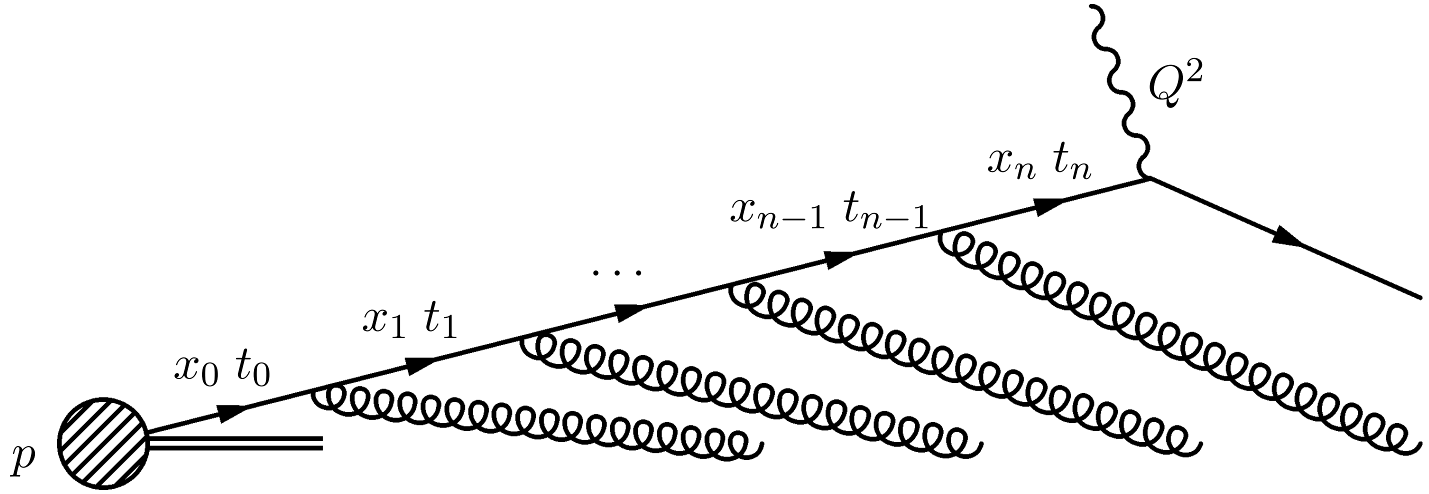
$$f(x,t) = f_0(x) + \int_{t_0}^{t} \frac{dt'}{t'} \frac{\alpha_s}{2\pi} \int_x^1 \frac{dz}{z} P(z) f_0\left(\frac{x}{z}\right)$$

- After a second emission, we have

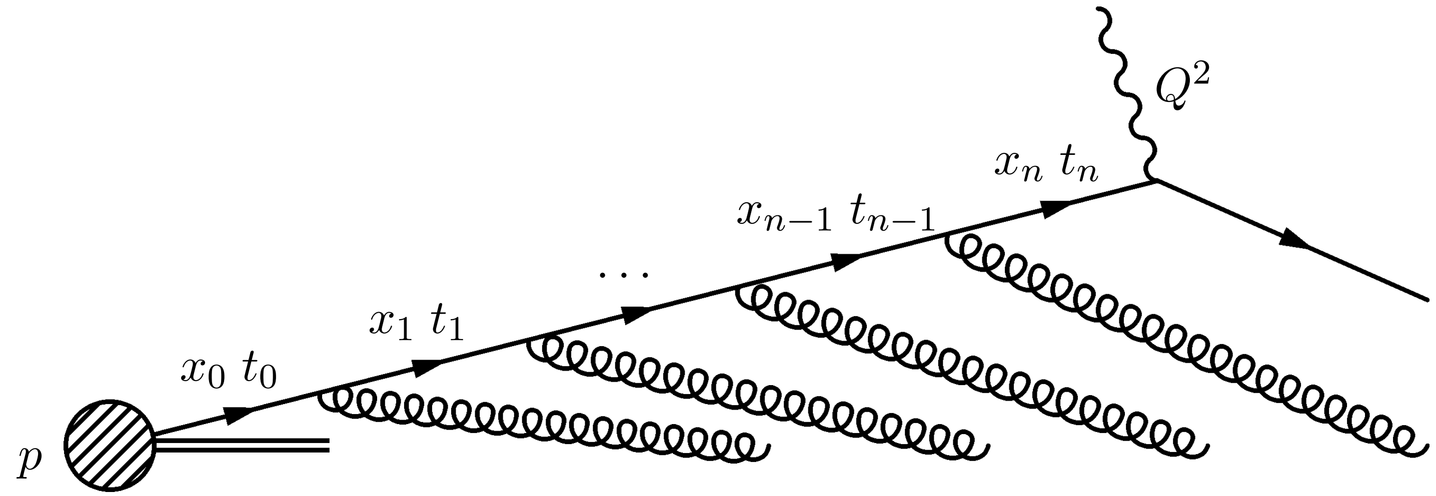
$$f(x,t) = f_0(x) + \int_{t_0}^{t} \frac{dt'}{t'} \frac{\alpha_s}{2\pi} \int_x^1 \frac{dz}{z} P(z) \left\{ f_0\left(\frac{x}{z}\right) + \int_{t_0}^{t'} \frac{dt''}{t''} \frac{\alpha_s}{2\pi} \int_{x/z}^1 \frac{dz'}{z'} P(z') f_0\left(\frac{x}{zz'}\right) \right\}$$

f(x/z, t')

The DGLAP equation

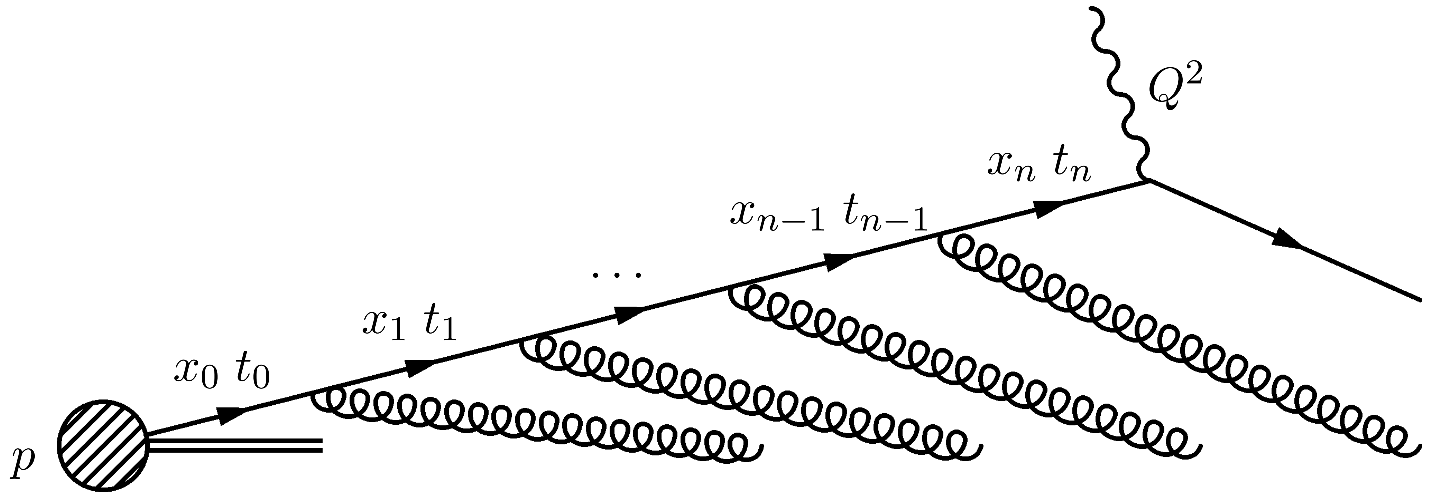


The DGLAP equation



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

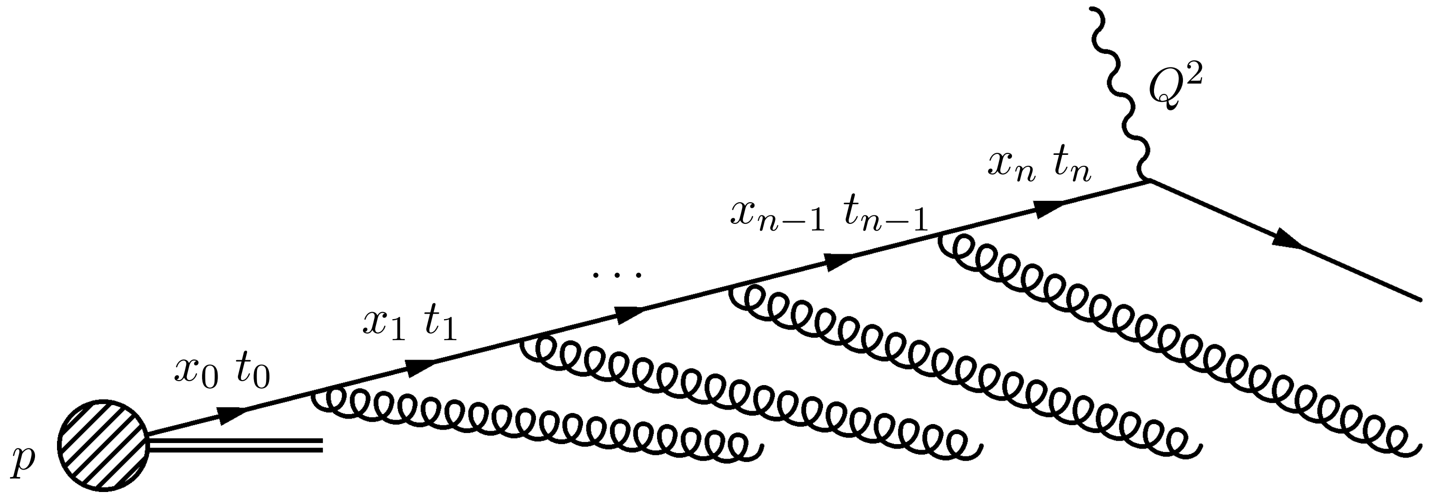
The DGLAP equation



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

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

The DGLAP equation

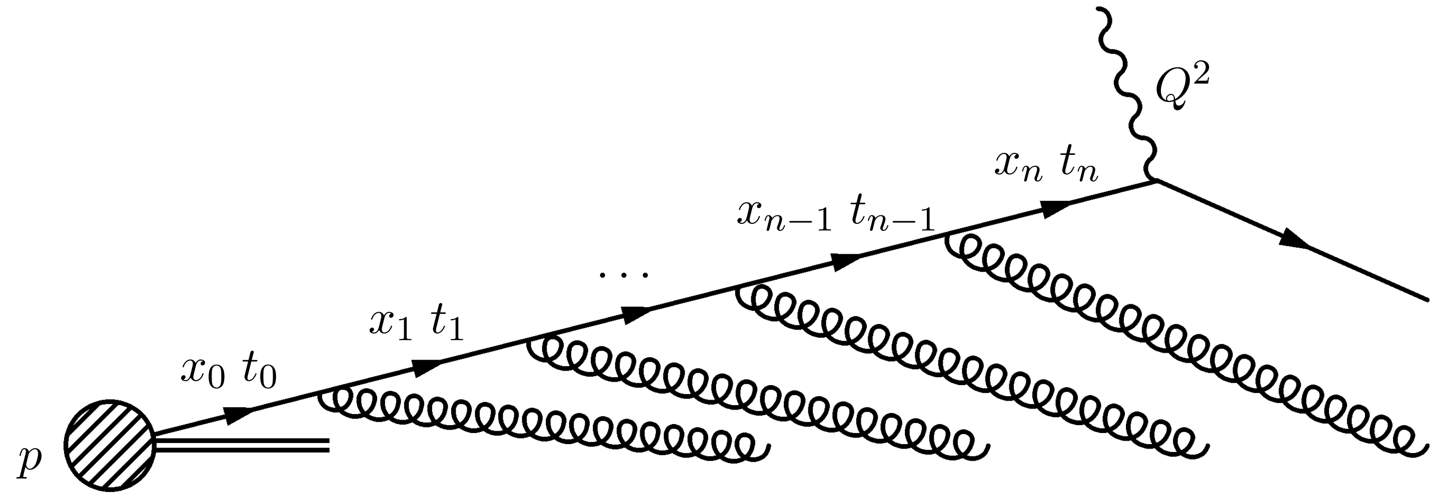


- So for multiple parton splittings, we arrive at an integral equation:

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

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

The DGLAP equation



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

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

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

- These starting PDFs are fitted to experimental data.

Initial-state parton showers

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

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

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

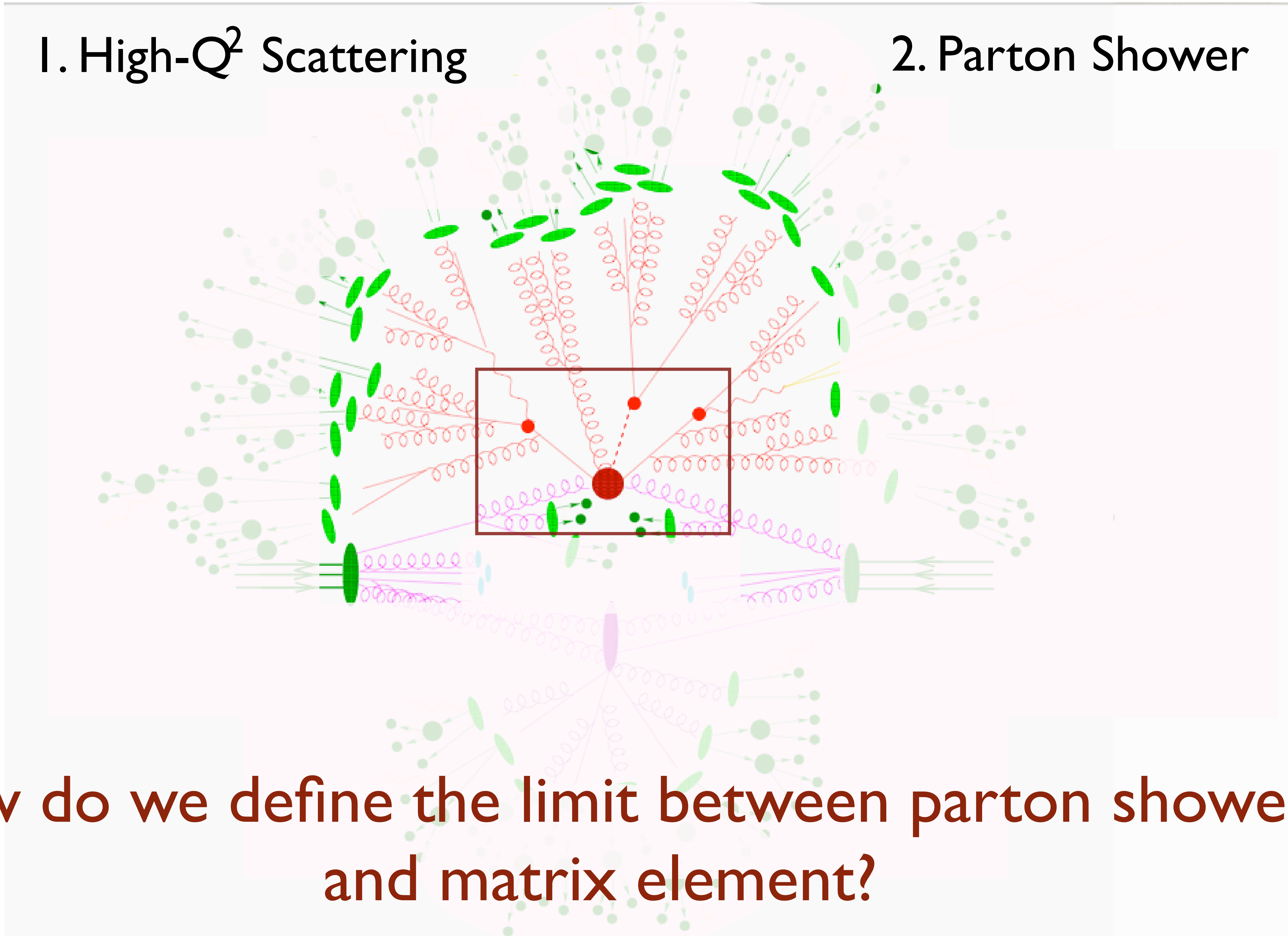
  This represents the probability that parton *i* will stay at the same *x* (no splittings) when evolving from t_1 to t_2.

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

Parton Shower MC event generators

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

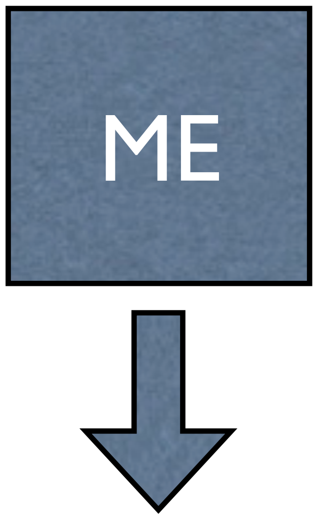
Back to our favorite piece of art!



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

Matrix Elements vs. Parton Showers

Matrix Elements vs. Parton Showers



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

Matrix Elements vs. Parton Showers

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

Matrix Elements vs. Parton Showers

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

Approaches are complementary: merge them!

Matrix Elements vs. Parton Showers

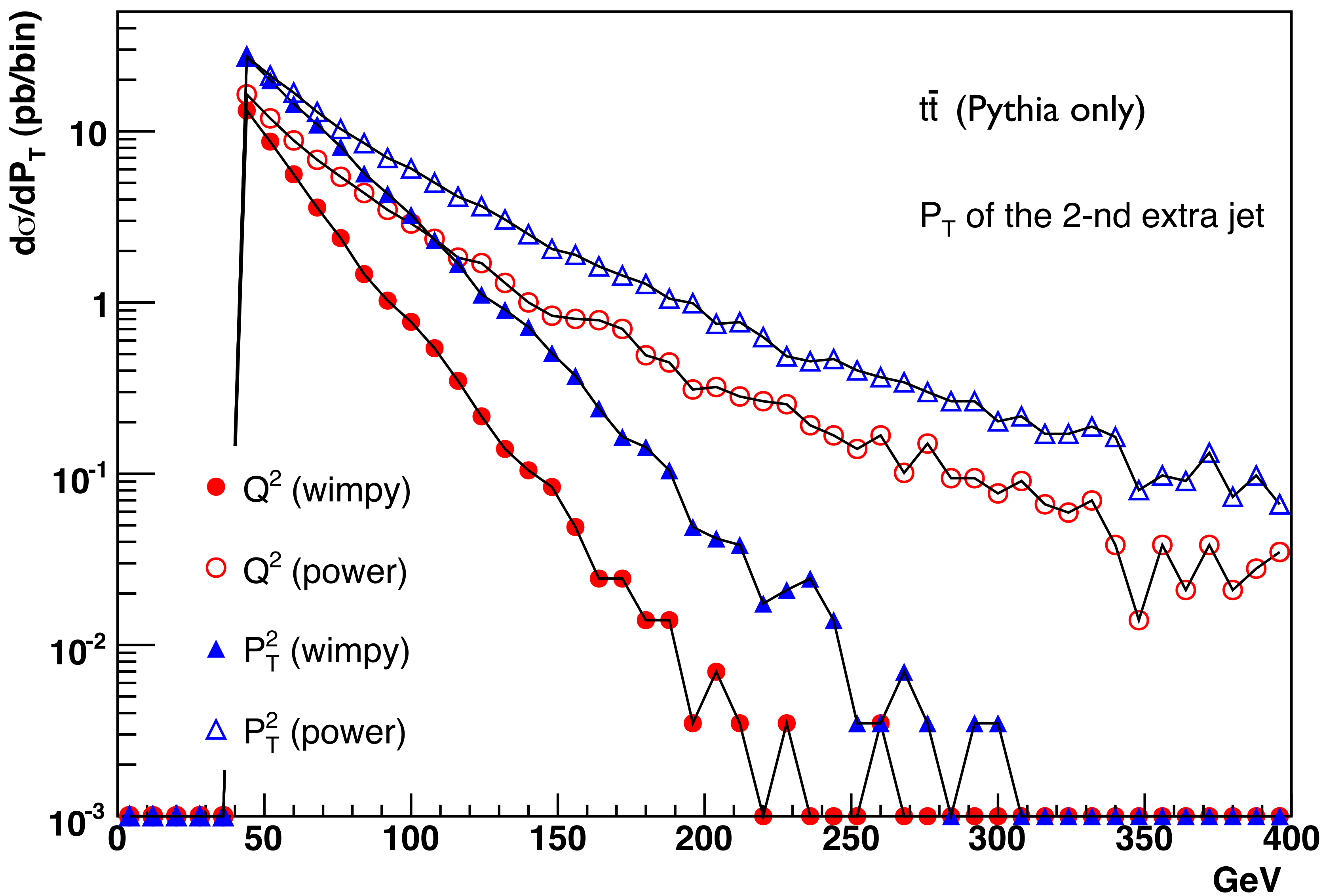
ME	Shower MC
1. Fixed order calculation	1. Resums logs to all orders
2. Computationally expensive	2. Computationally cheap
3. Limited number of particles	3. No limit on particle multiplicity
4. Valid when partons are hard and well separated	4. Valid when partons are collinear and/or soft
5. Quantum interference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

Approaches are complementary: merge them!

Difficulty: avoid double counting, ensure smooth distributions

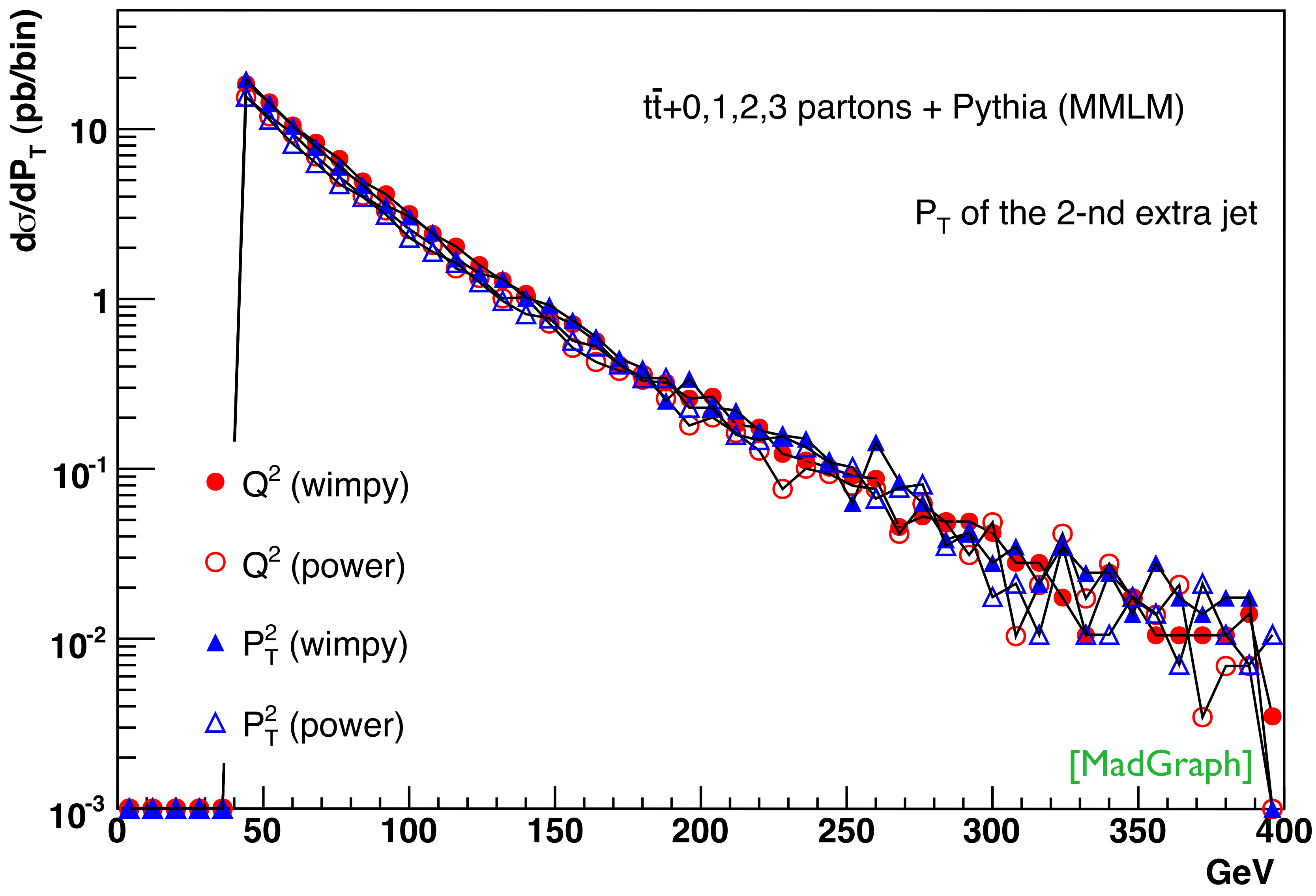
PS alone vs matched samples

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

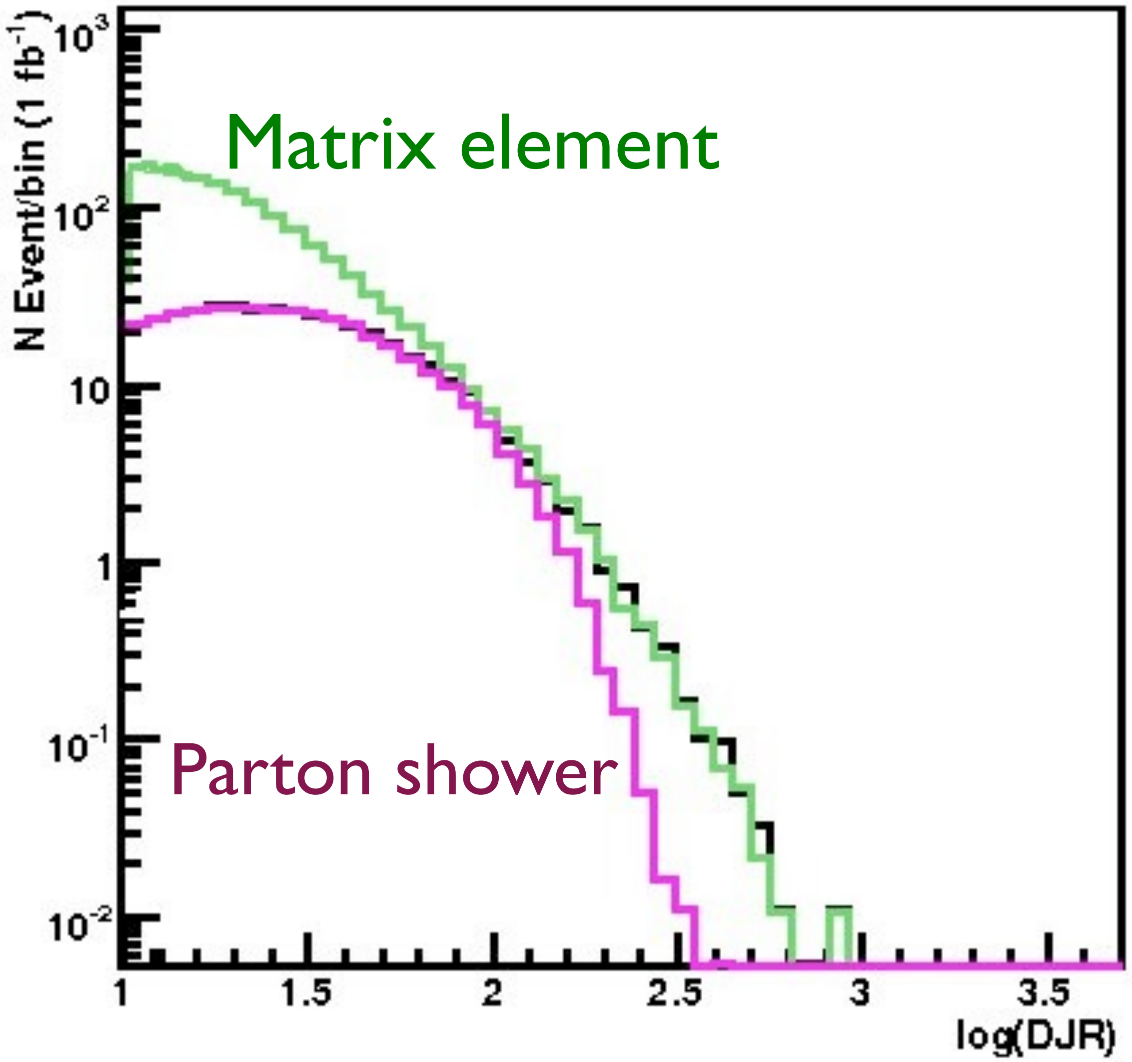


PS alone vs ME matching

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



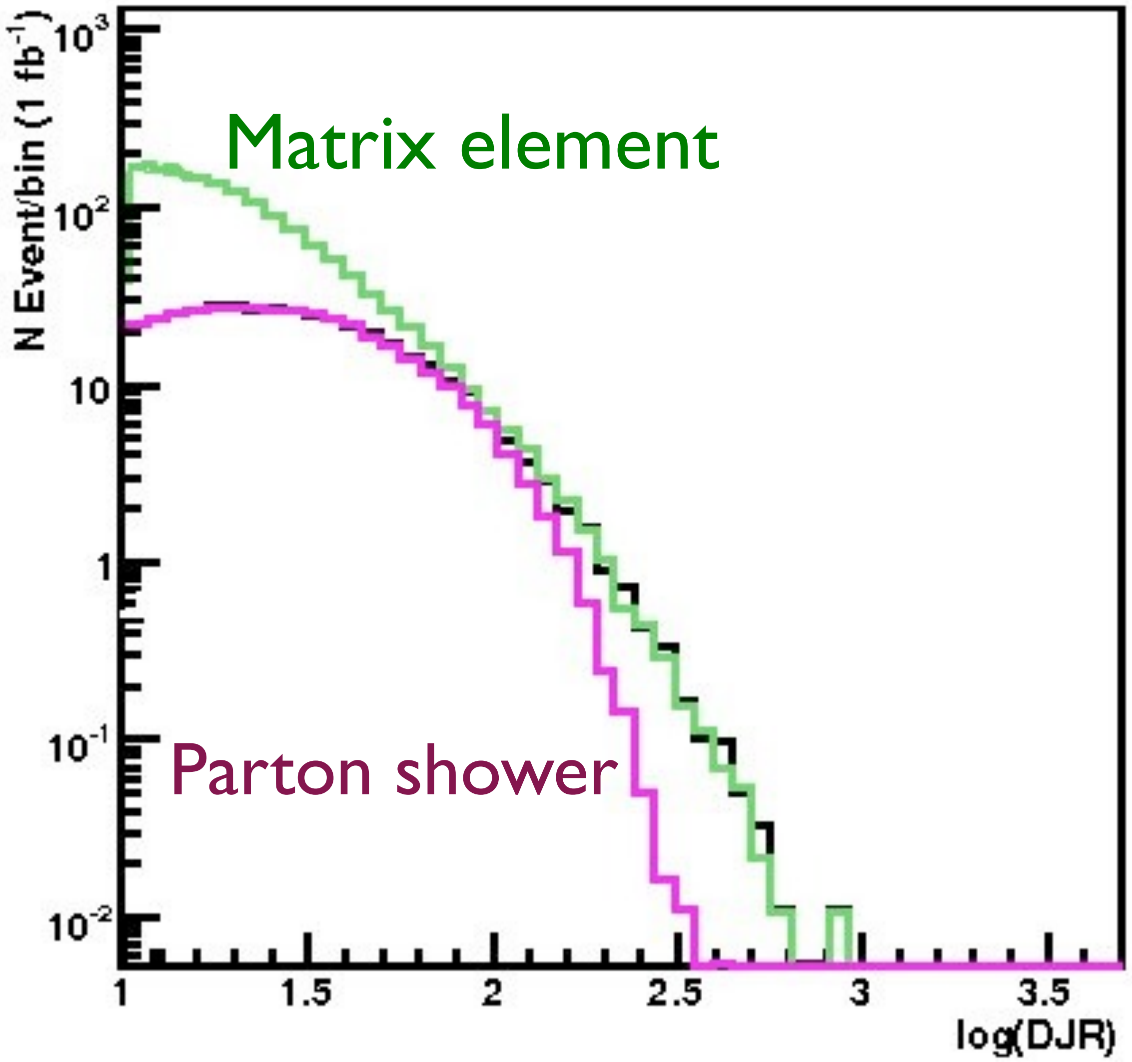
Goal for ME-PS merging/matching



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

Goal for ME-PS merging/matching

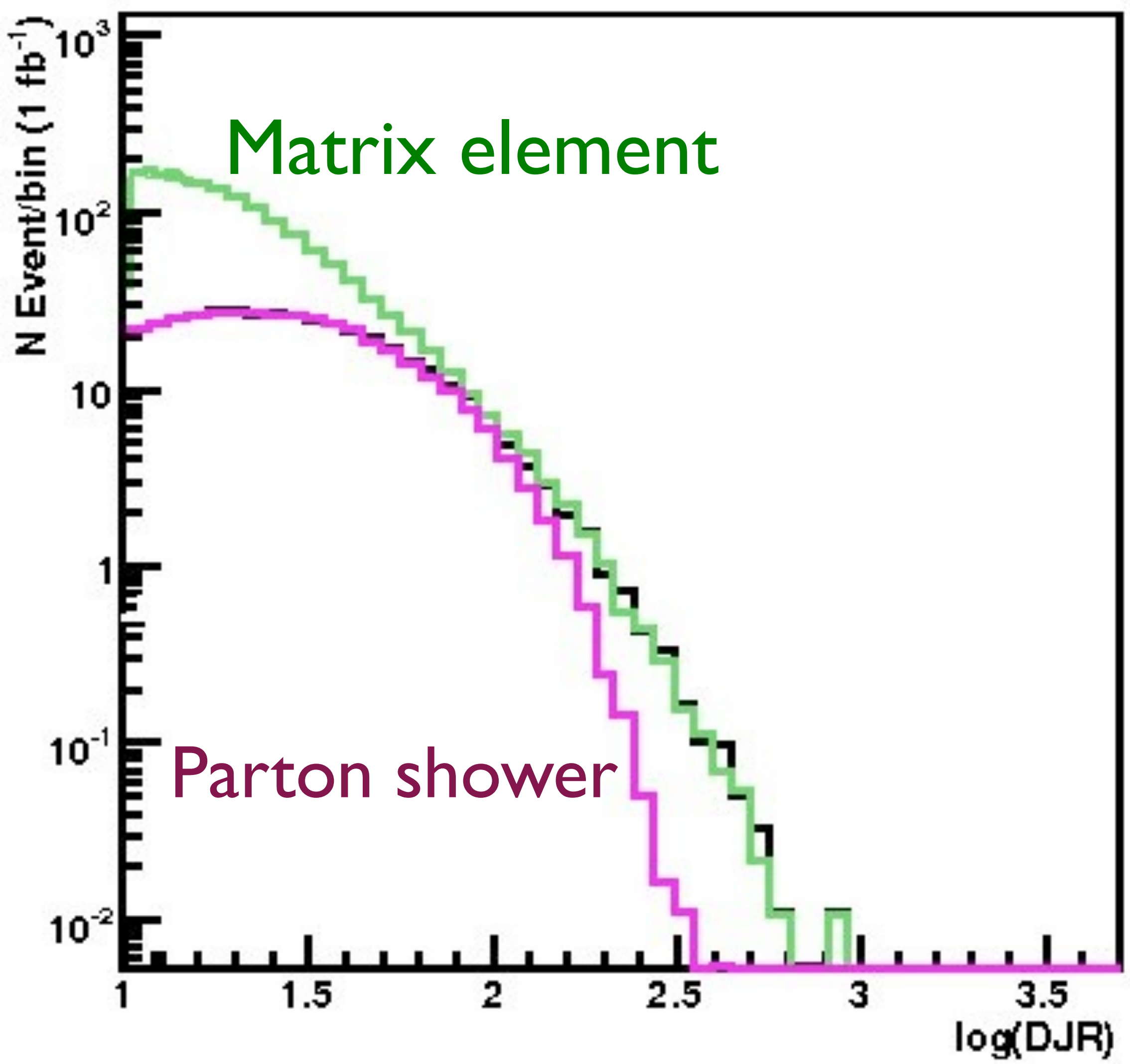
- Regularization of matrix element divergence



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

Goal for ME-PS merging/matching

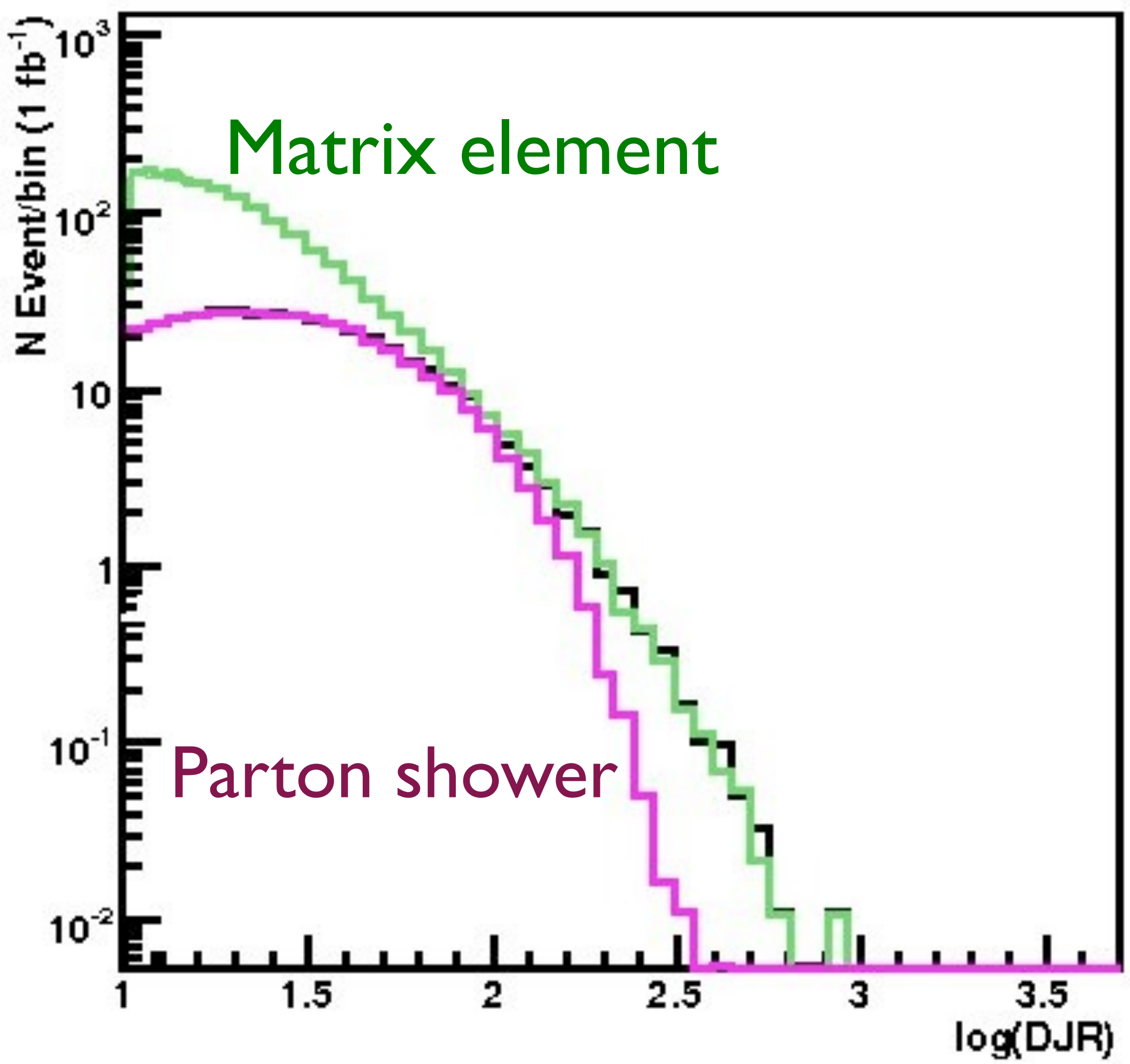
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta



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

Goal for ME-PS merging/matching

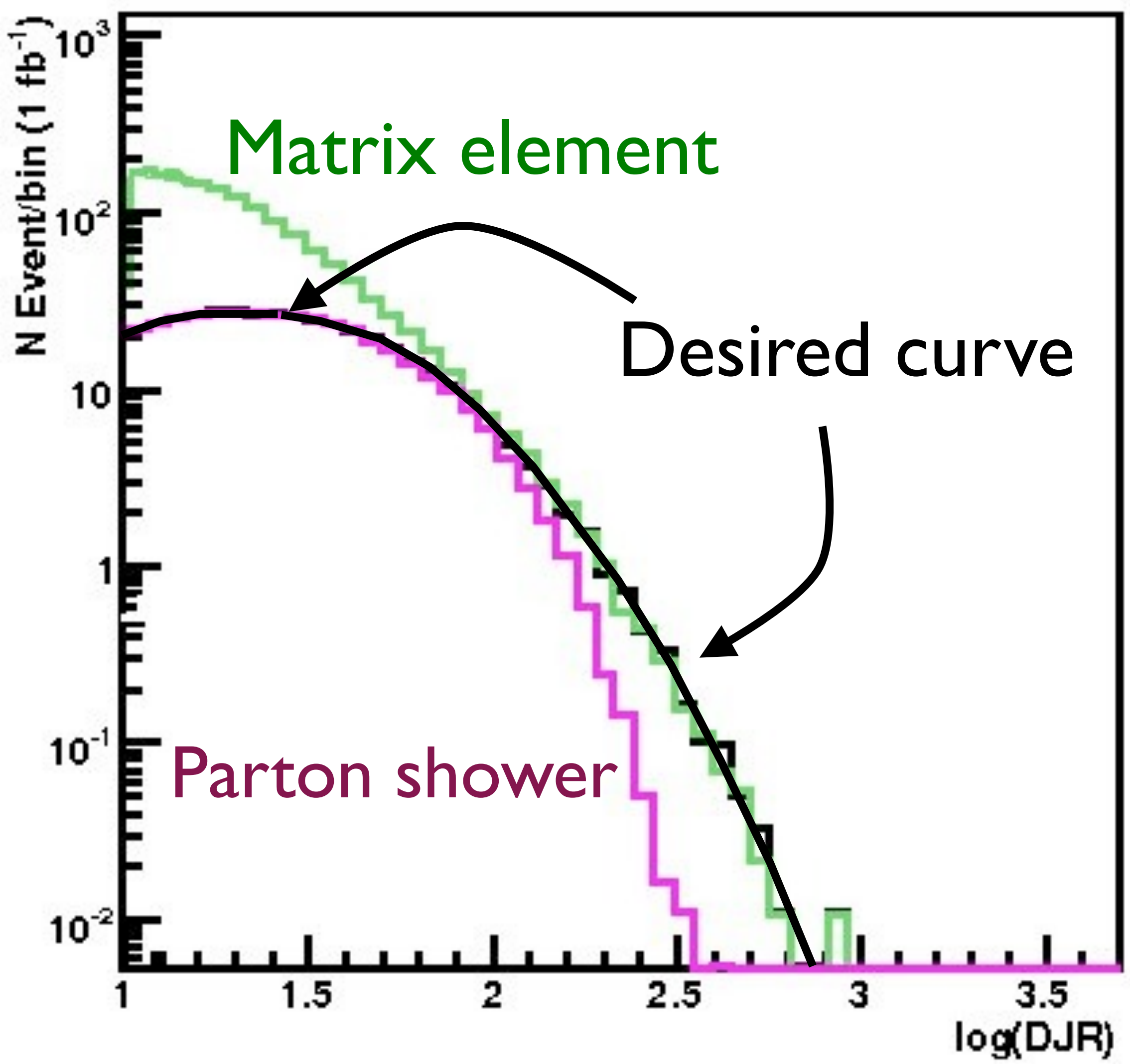
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



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

Goal for ME-PS merging/matching

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



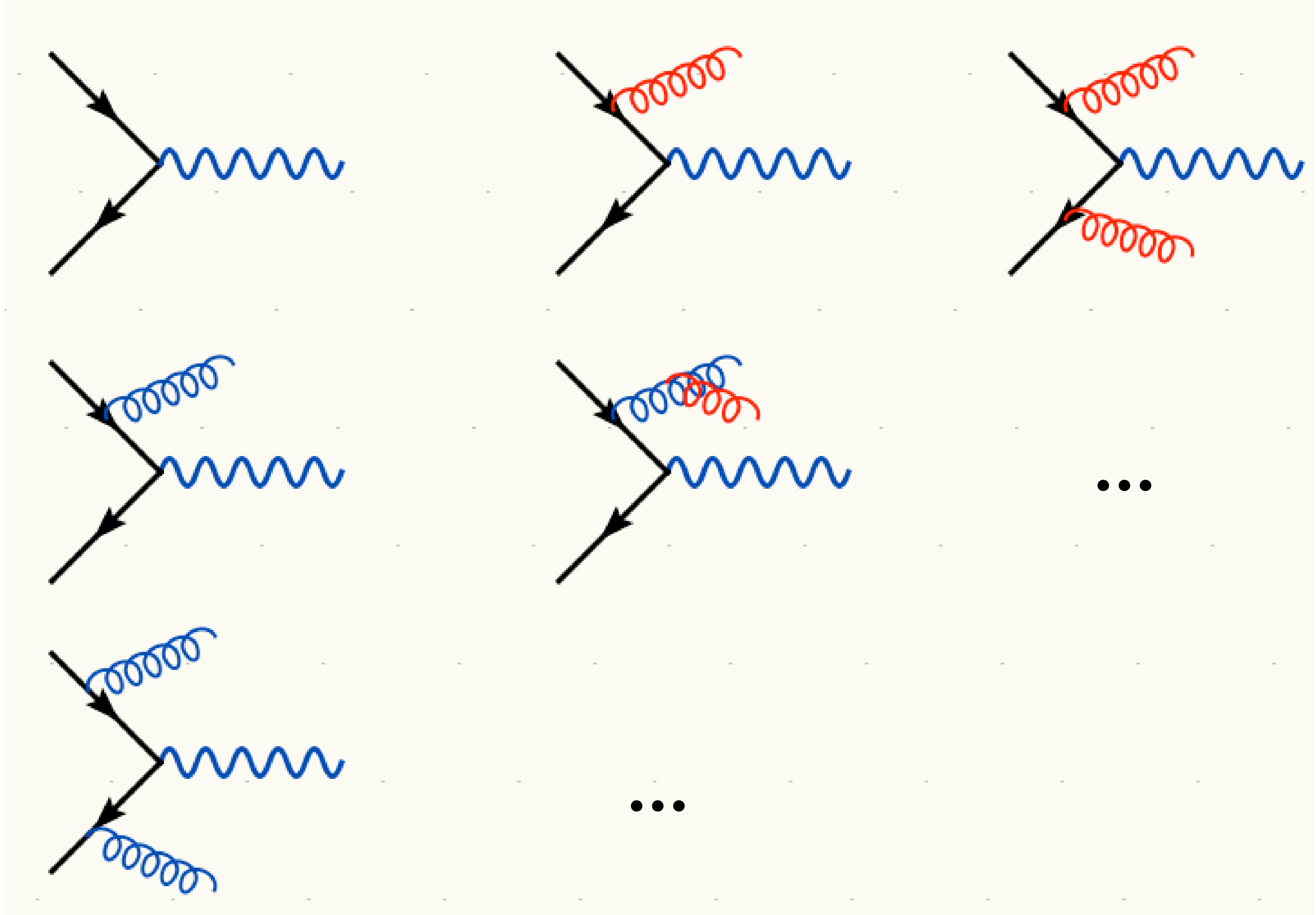
2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Merging ME with PS

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

PS →

ME ↓

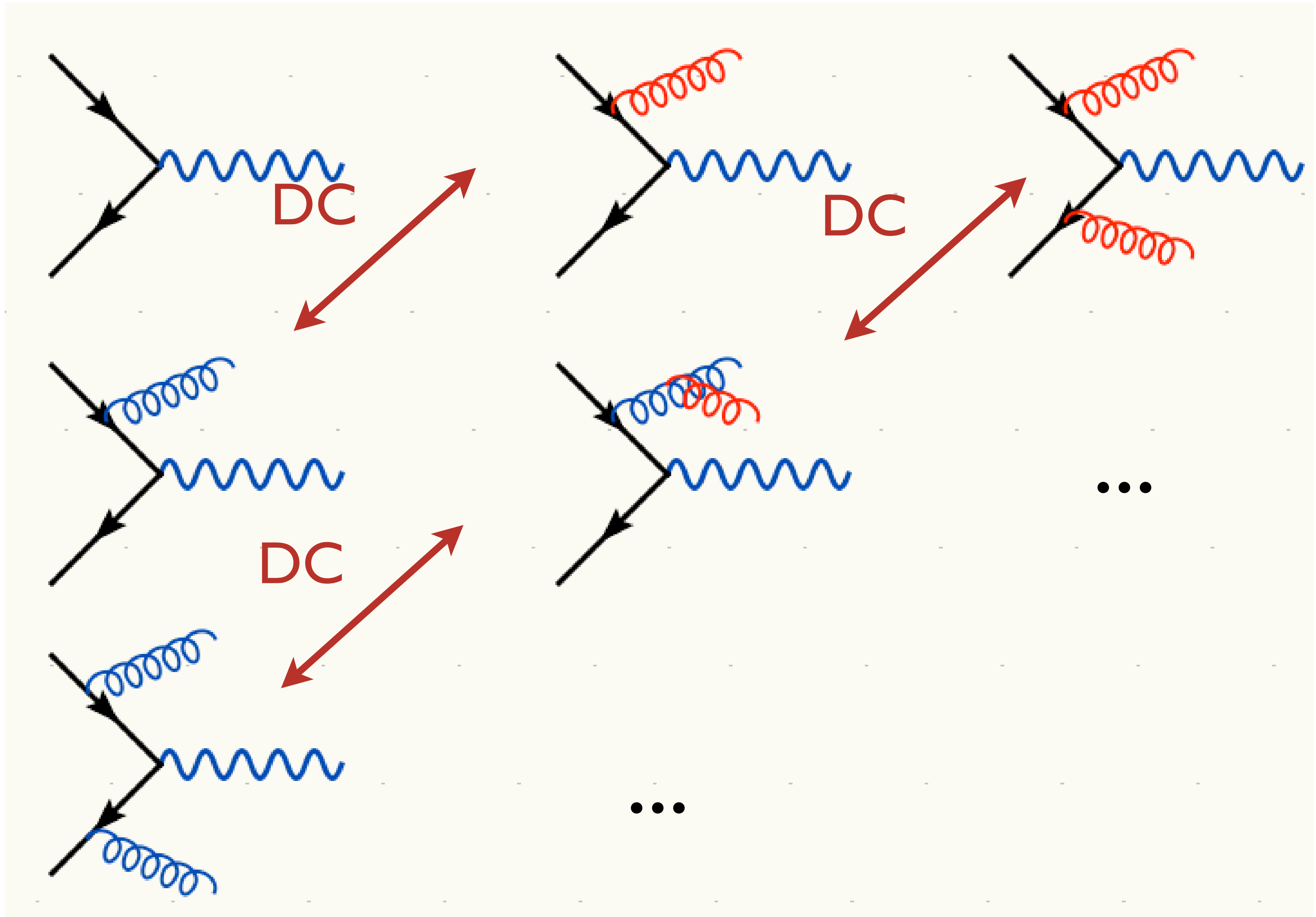


Merging ME with PS

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

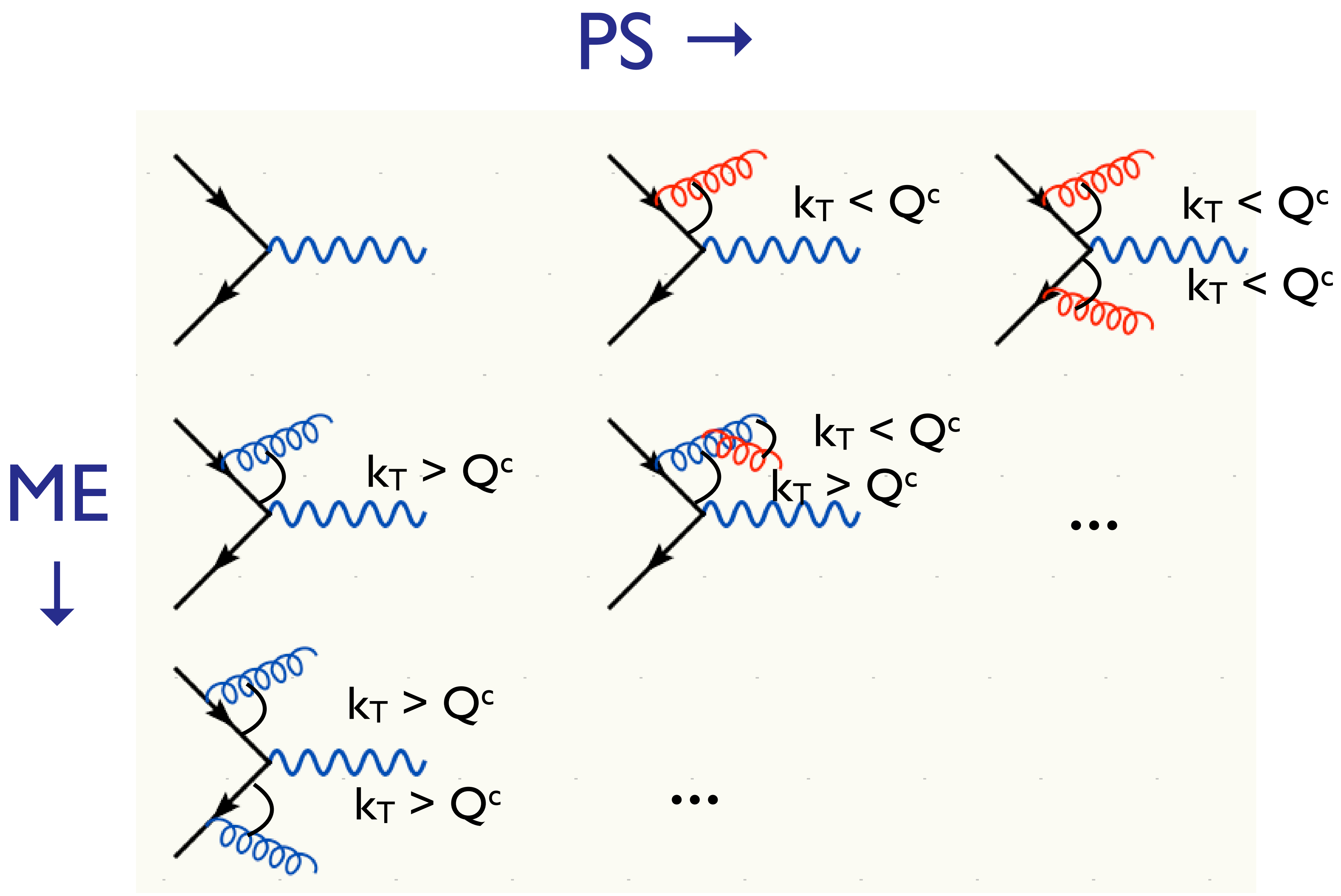
PS →

ME ↓



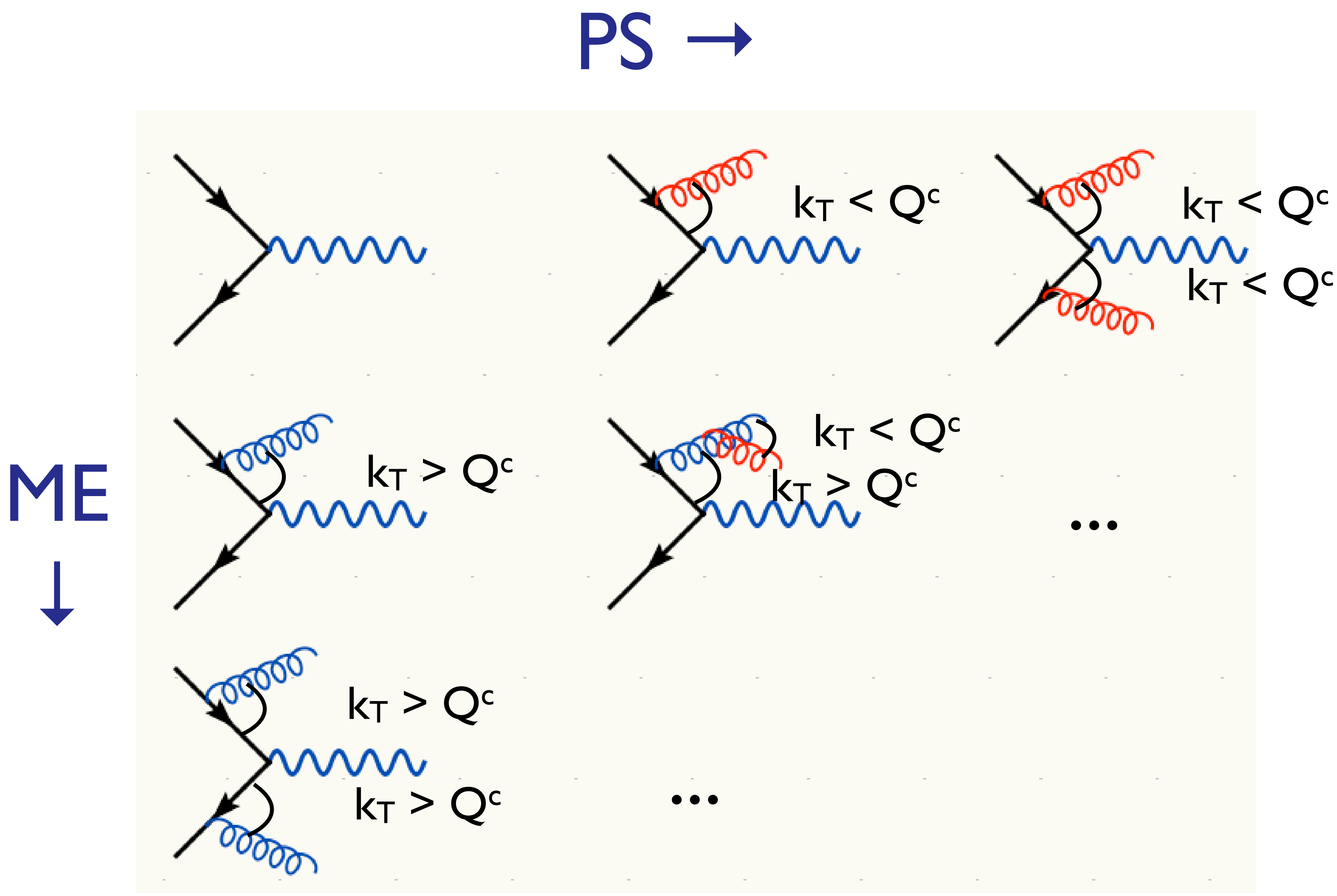
Merging ME with PS

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



Merging ME with PS

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

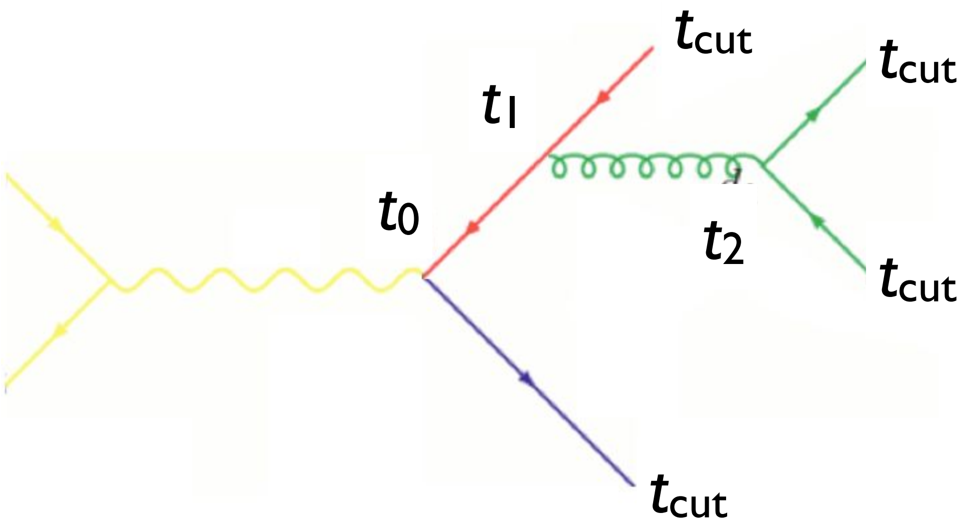


Double counting between ME and PS easily avoided using phase space cut between the two: PS below cutoff, ME above cutoff.

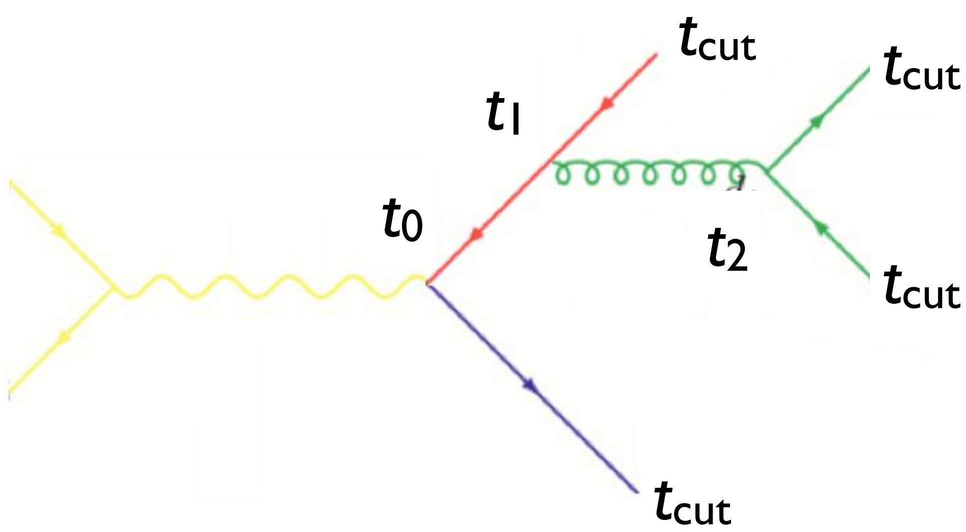
Merging ME with PS

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

Merging ME with PS

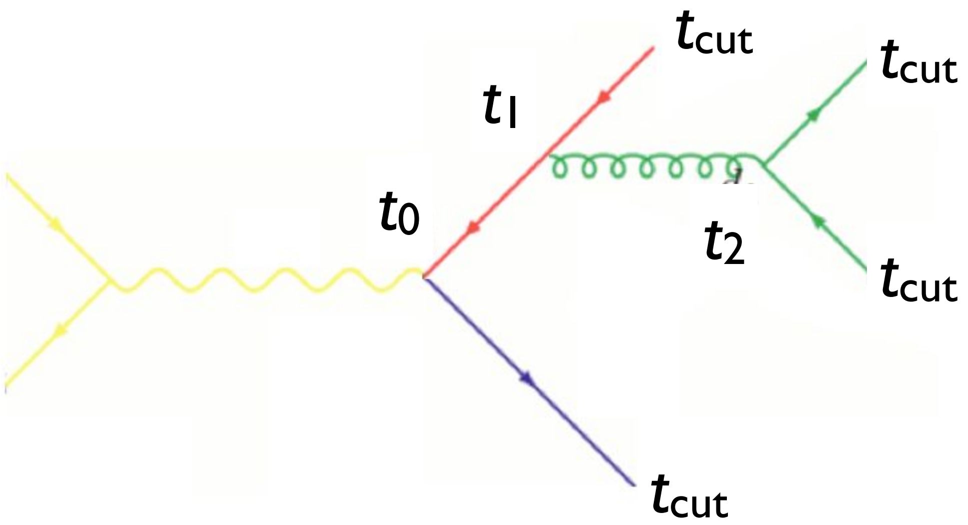


Merging ME with PS



- How does the PS generate the configuration above?

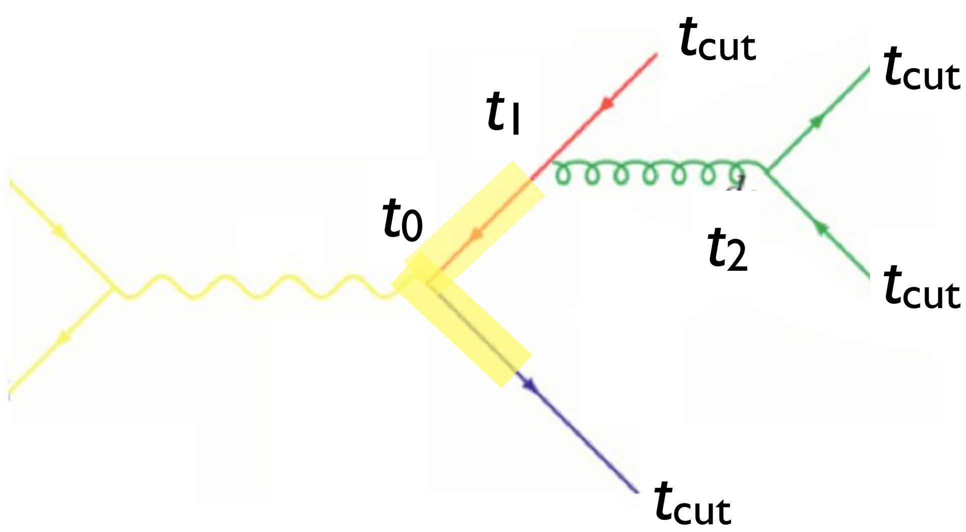
Merging ME with PS



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

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

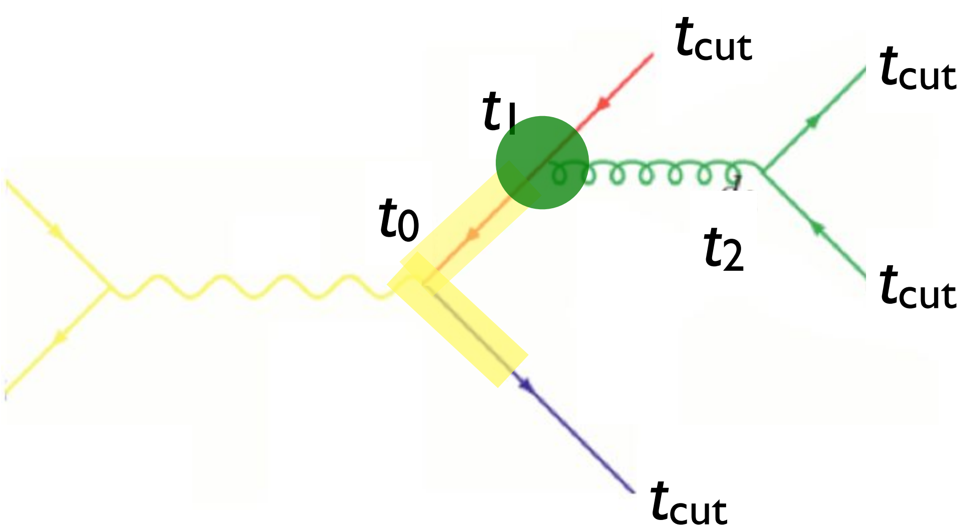
Merging ME with PS



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

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

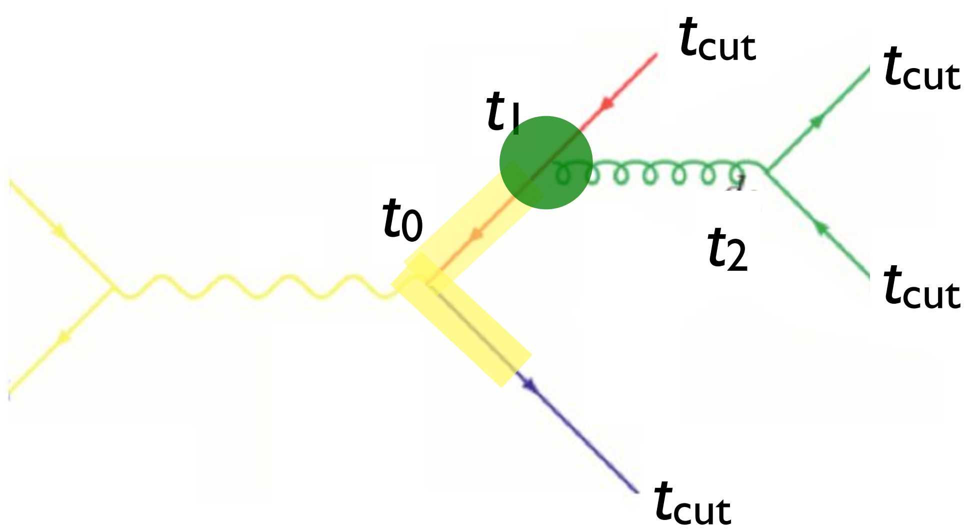
Merging ME with PS



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

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

Merging ME with PS



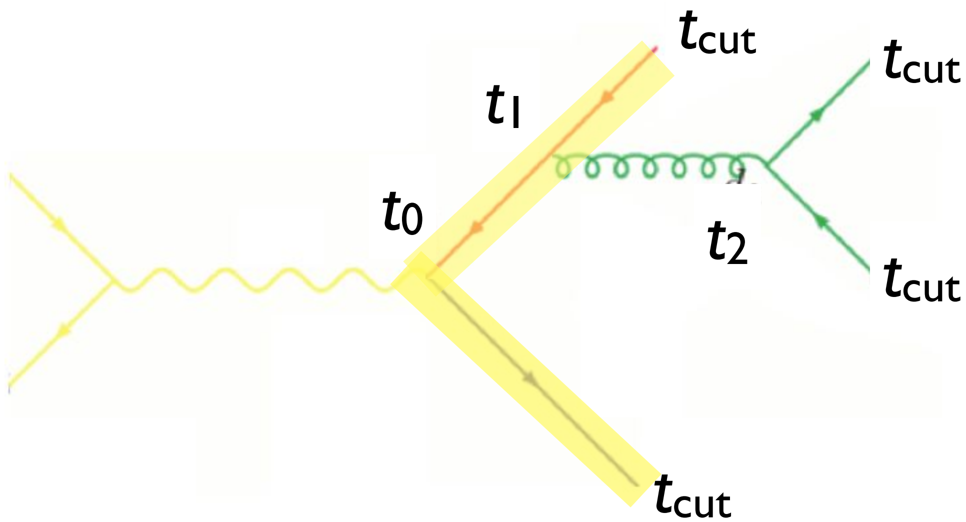
- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

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

and for the whole tree

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

Merging ME with PS



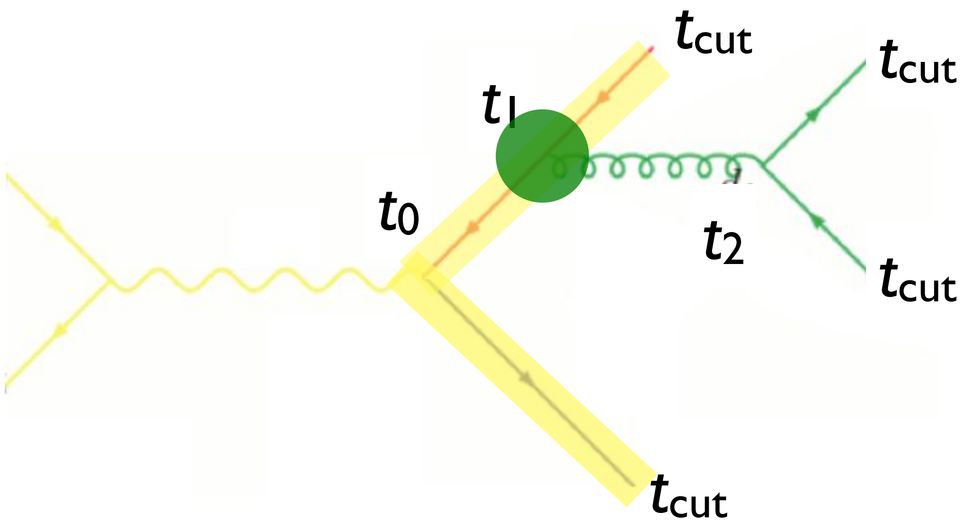
- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

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

and for the whole tree

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

Merging ME with PS



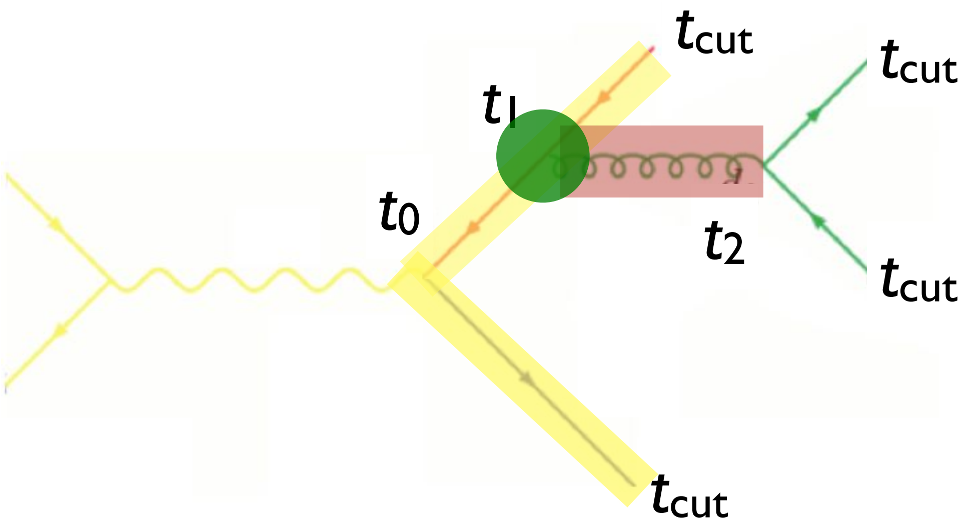
- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

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

and for the whole tree

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

Merging ME with PS



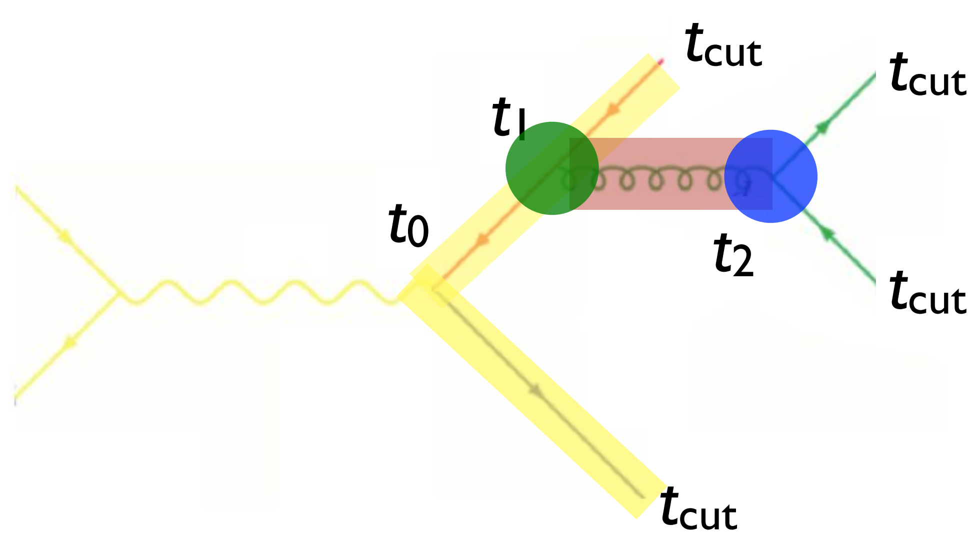
- How does the PS generate the configuration above?
- Probability for the splitting at t1 is given by

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

and for the whole tree

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

Merging ME with PS



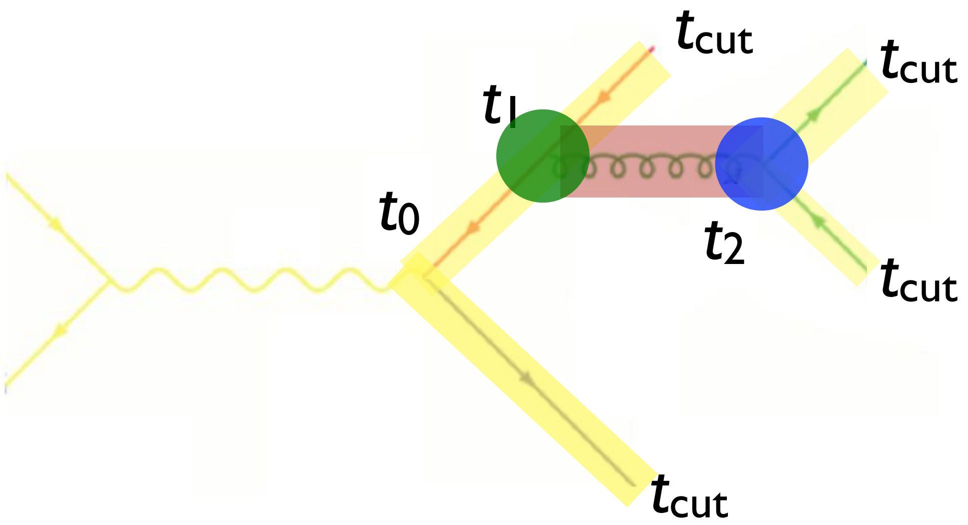
- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

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

and for the whole tree

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

Merging ME with PS



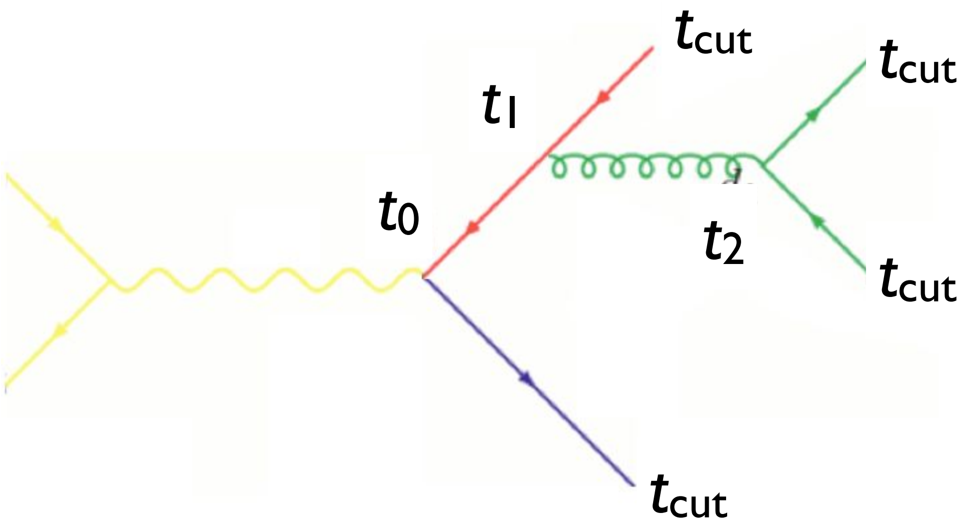
- How does the PS generate the configuration above?
- Probability for the splitting at t_1 is given by

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

and for the whole tree

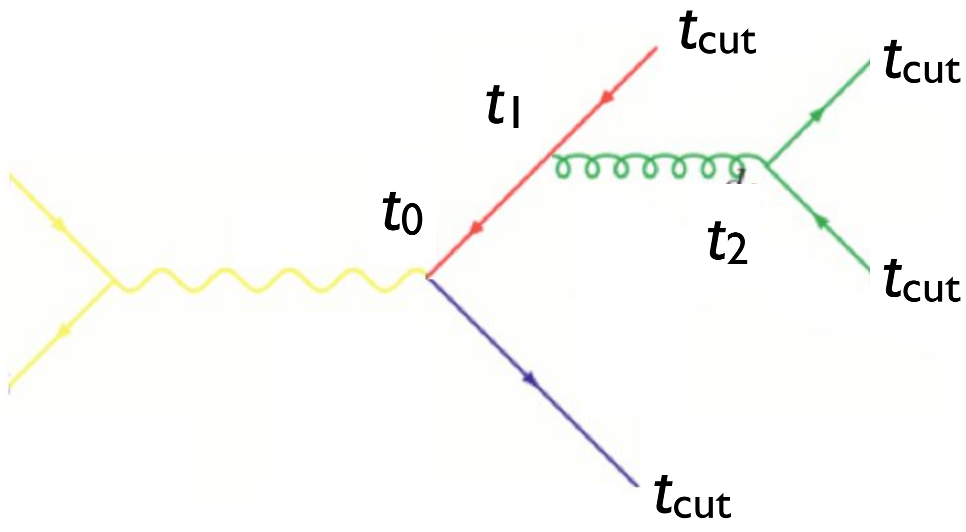
$$(\Delta_q(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1) (\Delta_q({\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z) \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$

Merging ME with PS



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

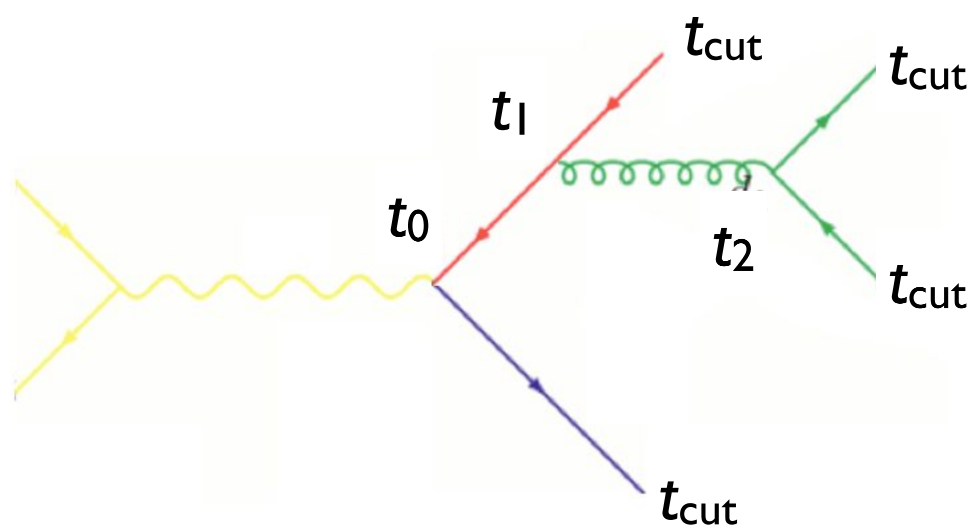
Merging ME with PS



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

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

Merging ME with PS

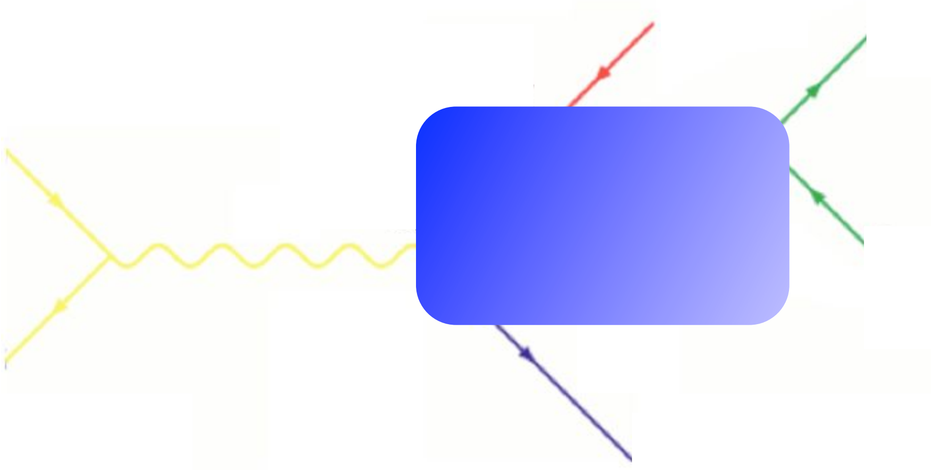


$$(\Delta_q(t_{\mathrm{cut}}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(_{\mathrm{cut}}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} P_{gq}(z) \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$

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

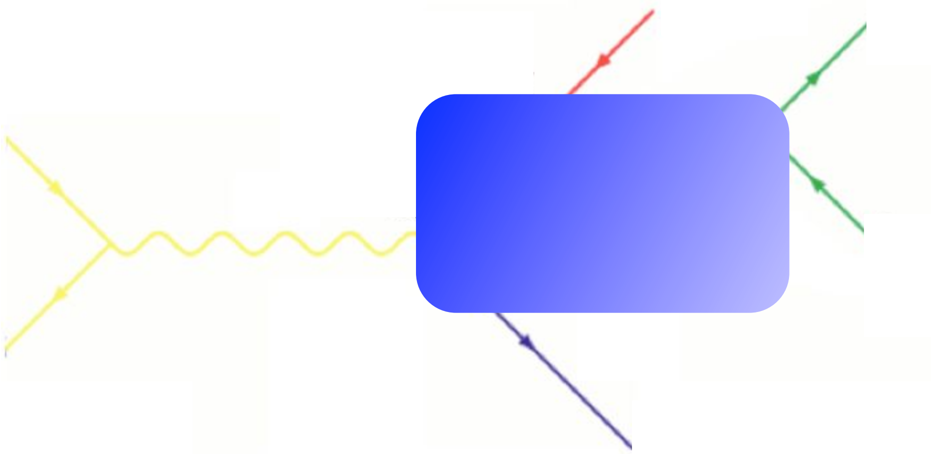
Sudakov suppression due to disallowing additional radiation above the scale t_{cut}

Merging ME with PS



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

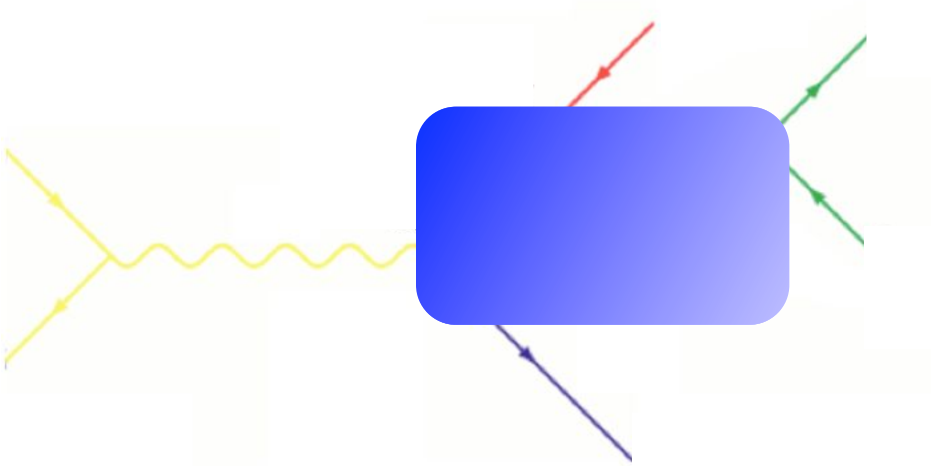
Merging ME with PS



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

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

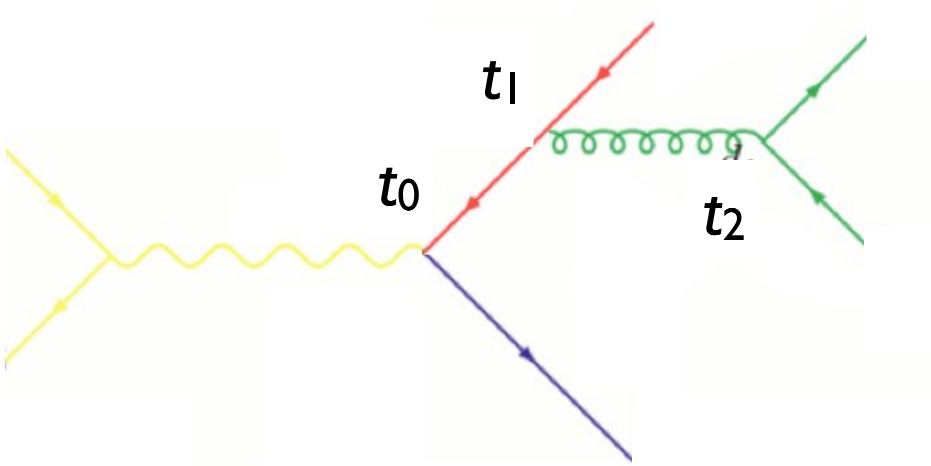
Merging ME with PS



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

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

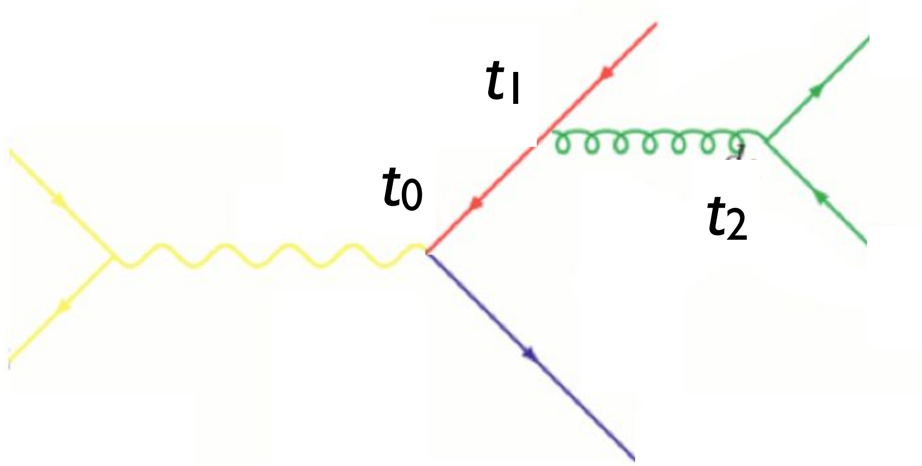
Merging ME with PS



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

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

Merging ME with PS



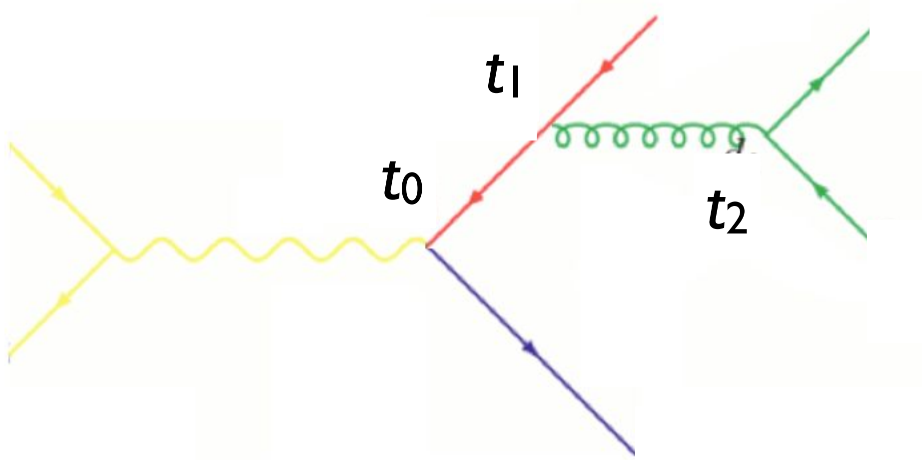
$$|\mathcal{M}|^2(\hat{s}, p_3, p_4, \ldots)$$

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

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

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

Merging ME with PS

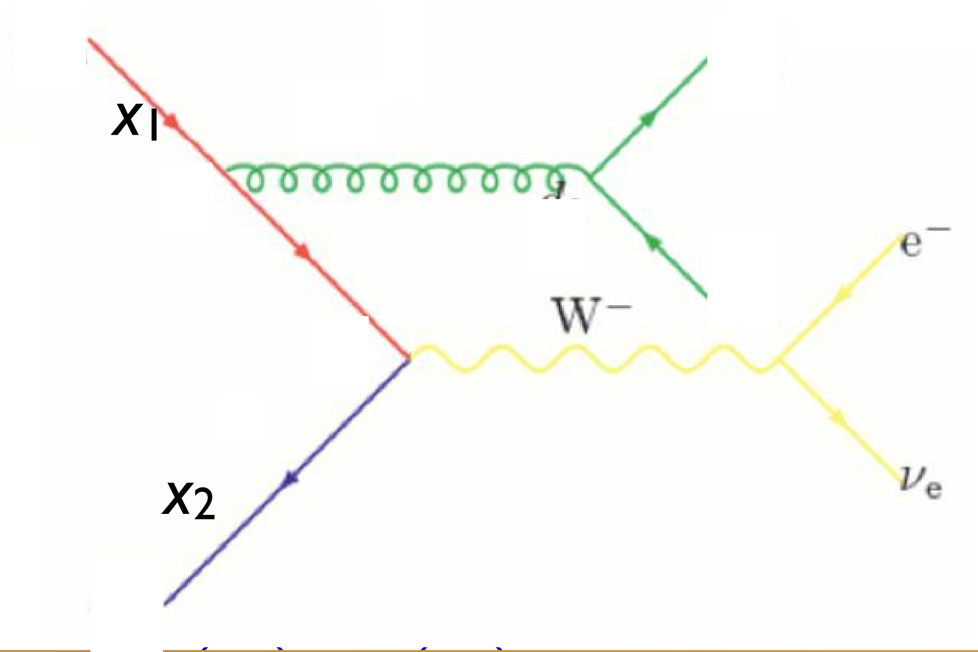


$$|\mathcal{M}|^2(\hat{s}, p_3, p_4, \ldots)$$

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

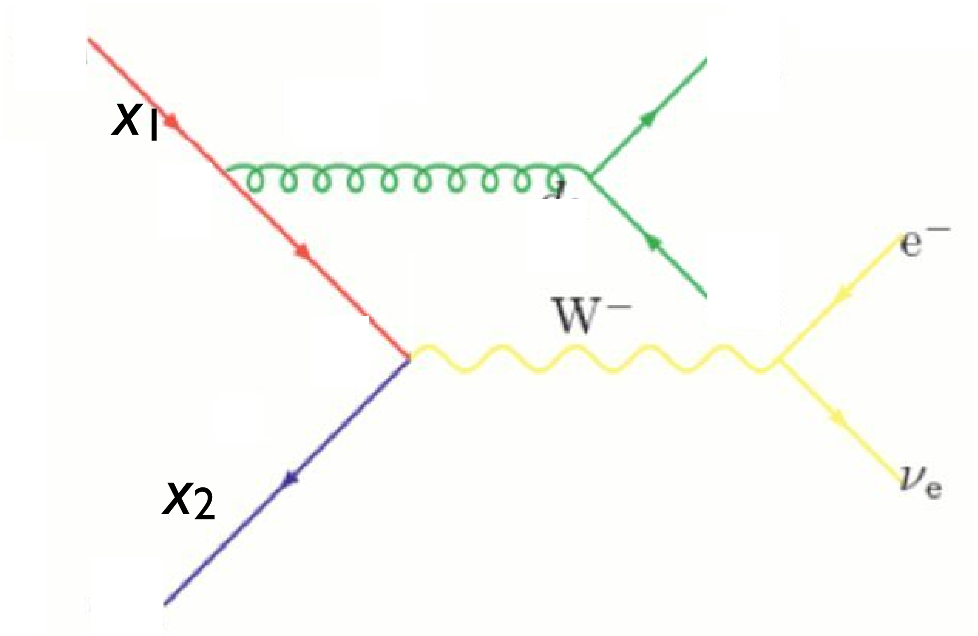
1. Cluster the event using some clustering algorithm
   - this gives us a corresponding "parton shower history"
2. Reweight α_s in each clustering vertex with the clustering scale
   $$|\mathcal{M}|^2 \to |\mathcal{M}|^2 \frac{\alpha_s(t_1)}{\alpha_s(t_0)} \frac{\alpha_s(t_2)}{\alpha_s(t_0)}$$
3. Use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

Matching for initial state radiation



Matching for initial state radiation

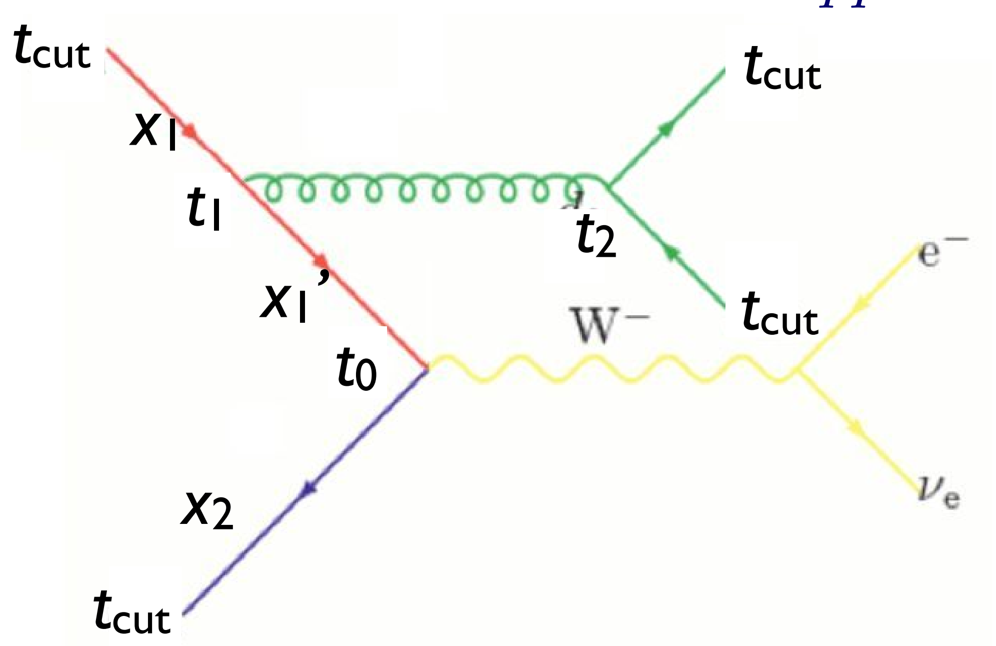
- We are of course not interested in e^+e^- but p-p(bar) - what happens for initial state radiation?



Matching for initial state radiation

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

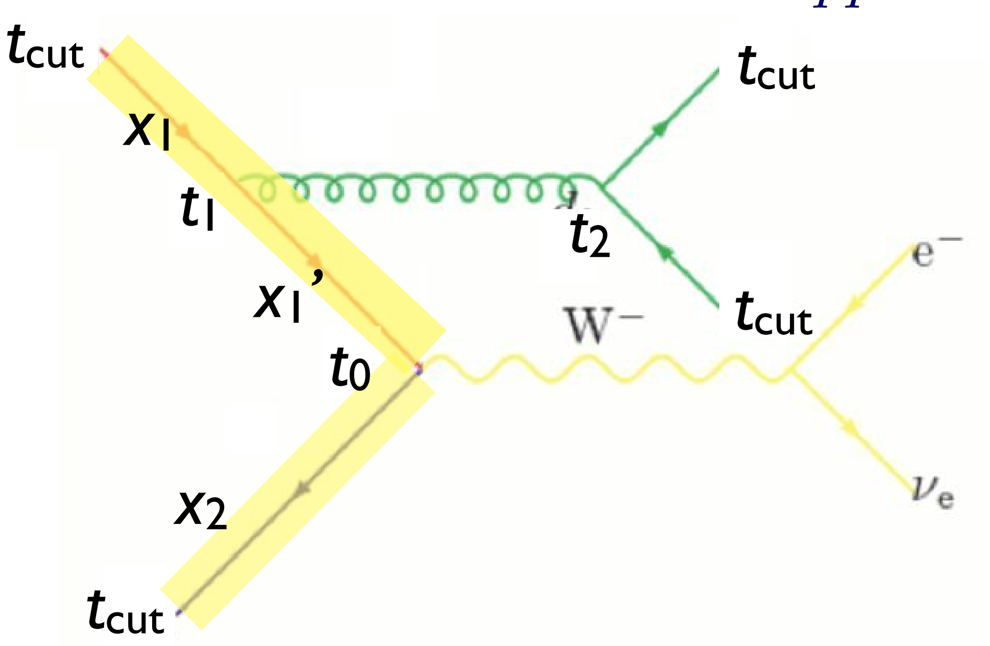
$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut},t_0))^2 \Delta_g(t_2,t_1)(\Delta_q(t_{\rm cut},t_2))^2 \frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1,t_1)}{f_q(x_1',t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$
$$\times\hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s},\ldots)f_q(x_1',t_0)f_{\bar{q}}(x_2,t_0)$$



Matching for initial state radiation

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

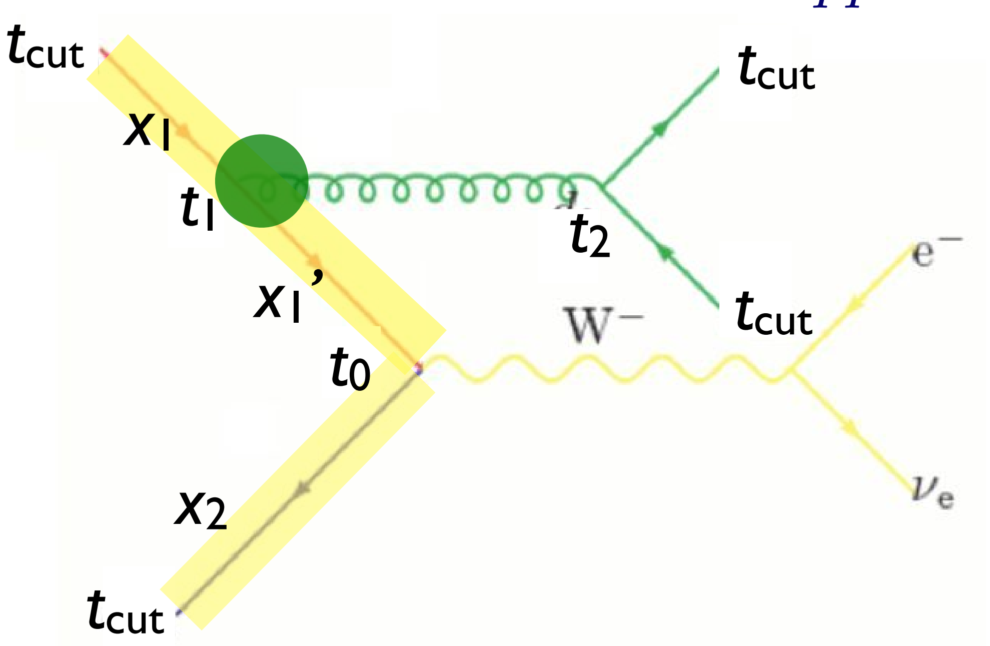
$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} \frac{P_{gq}(z)}{z} \frac{f_q(x_1, t_1)}{f_q(x_1', t_1)} \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z') \times \hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s}, \ldots) f_q(x_1', t_0) f_{\bar{q}}(x_2,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



Matching for initial state radiation

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

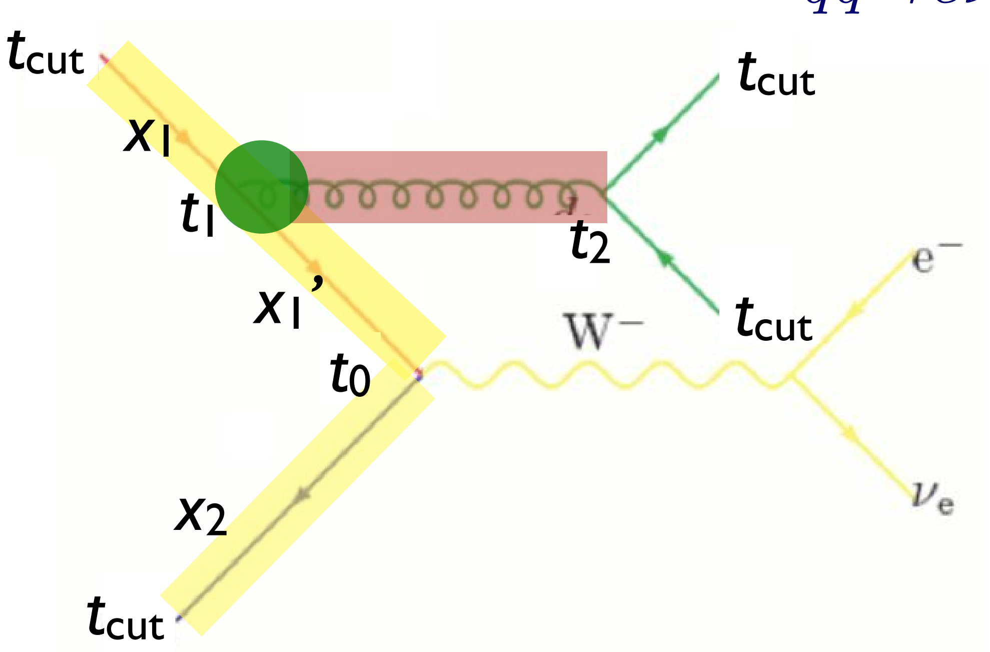
$$\mathcal{P} = (\Delta_{Iq}(t_{\rm cut}, t_0))^2 \Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2 \frac{\alpha_s(t_1)}{2\pi} \frac{P_{gq}(z)}{z} \frac{f_q(x_1, t_1)}{f_q(x_1', t_1)} \frac{\alpha_s(t_2)}{2\pi} P_{qg}(z')$$
$$\times \hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s}, \ldots) f_q(x_1', t_0) f_{\bar{q}}(x_2,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



Matching for initial state radiation

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

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

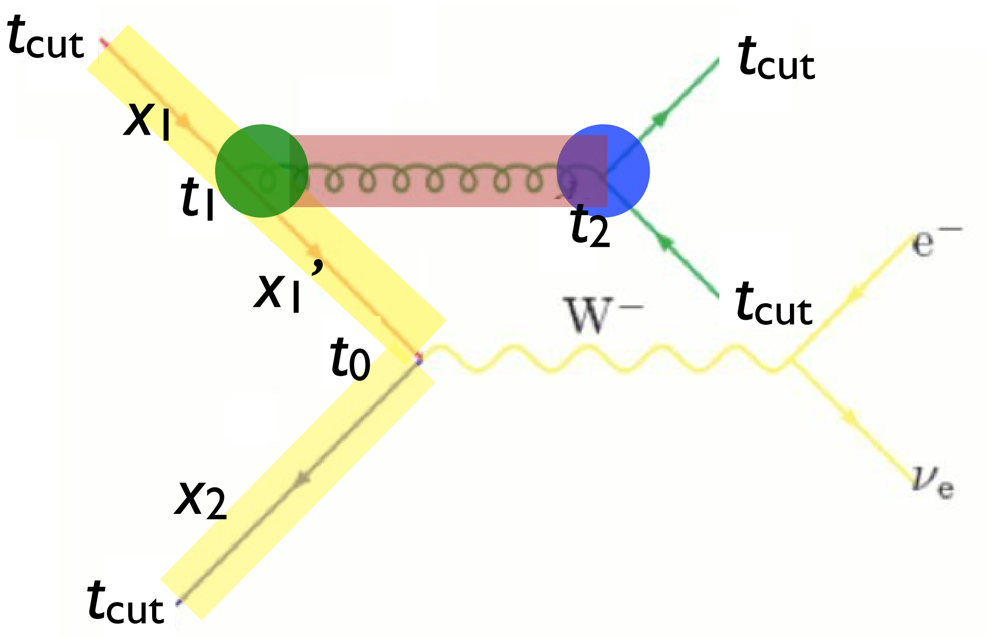


Matching for initial state radiation

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

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

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

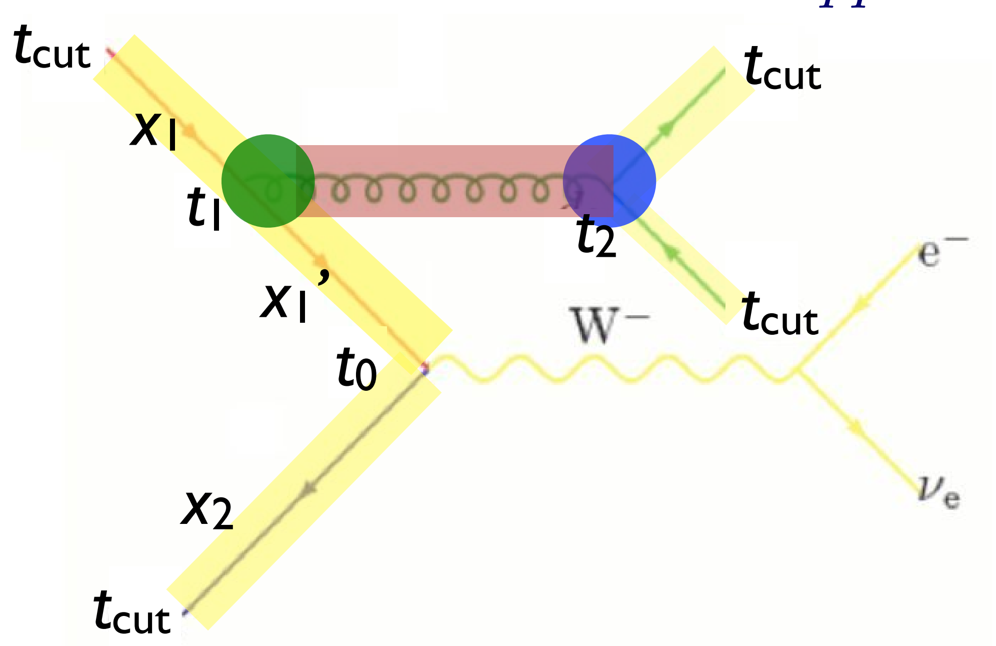


Matching for initial state radiation

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

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

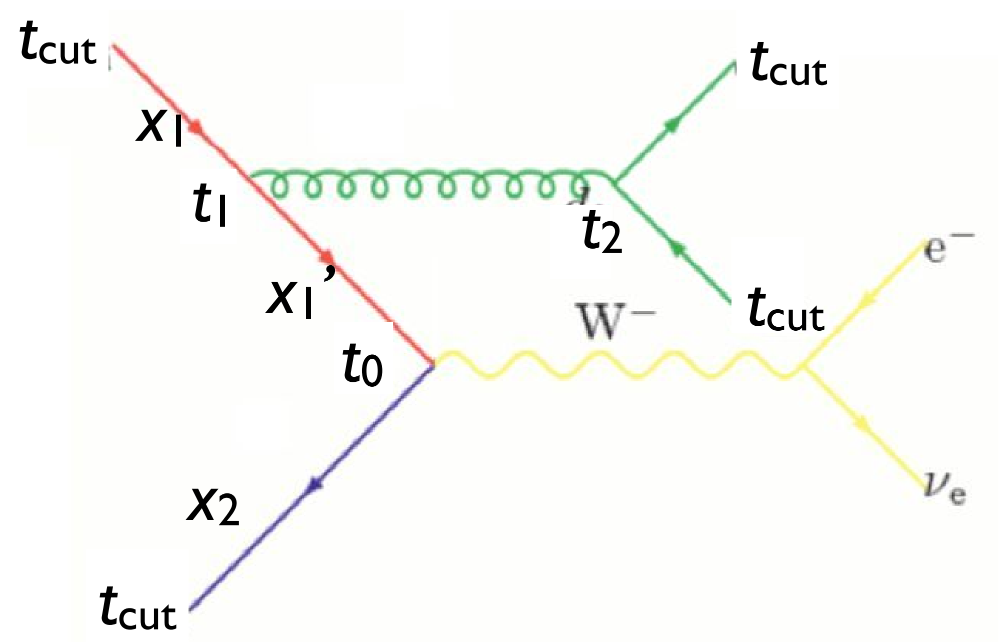
$$\times \hat{\sigma}_{q\bar{q}\to e\nu}(\hat{s},\ldots) f_q(x_1', t_0) f_{\bar{q}}(x_2, t_0)$$



Matching for initial state radiation

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

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

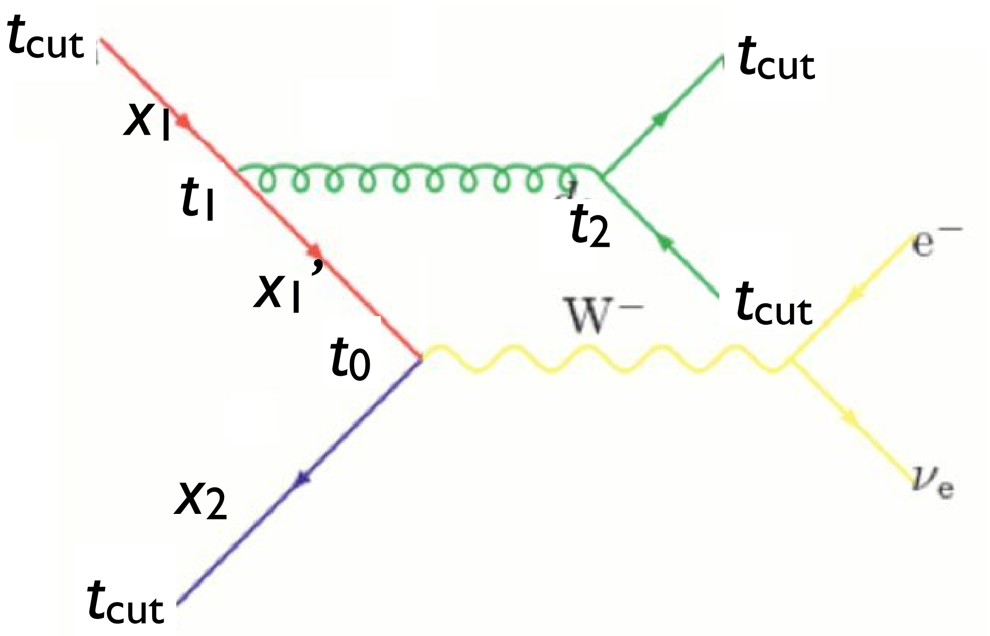


Matching for initial state radiation

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

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

ME with α_s evaluated at the scale of each splitting



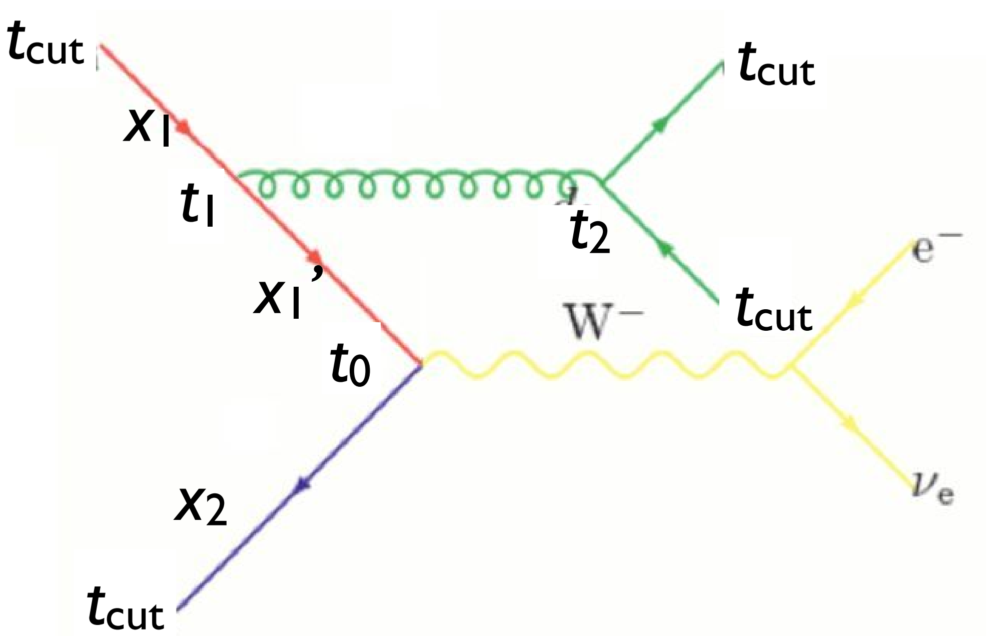
Matching for initial state radiation

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

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

ME with α_s evaluated at the scale of each splitting

PDF reweighting



Matching for initial state radiation

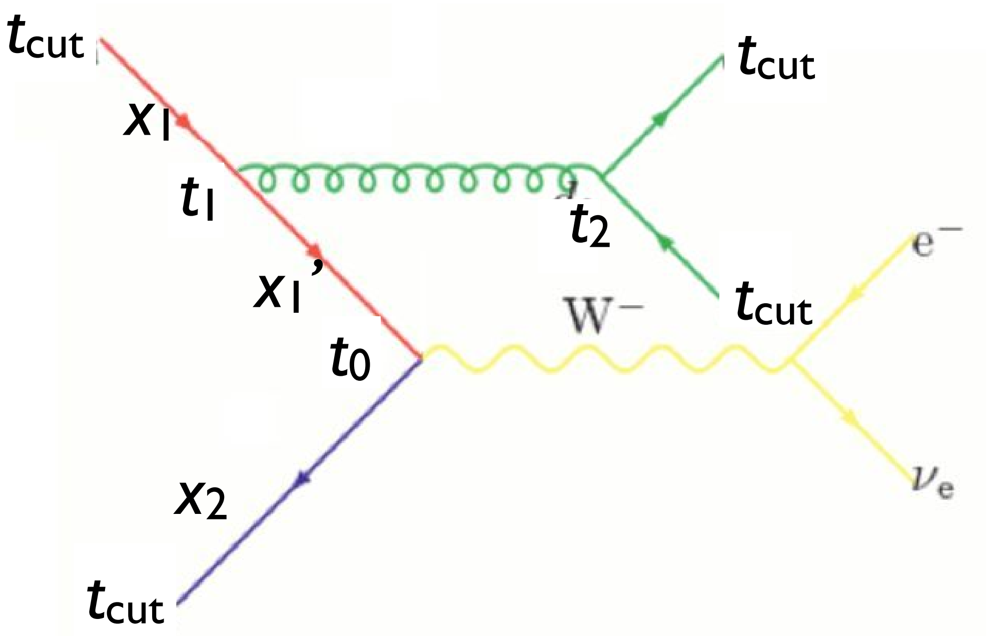
$$(\Delta_{Iq}(t_{\rm cut},t_0))^2\Delta_g(t_2,t_1)(\Delta_q(t_{\rm cut},t_2))^2\frac{\alpha_s(t_1)}{2\pi}\frac{P_{gq}(z)}{z}\frac{f_q(x_1,t_1)}{f_q(x_1',t_1)}\frac{\alpha_s(t_2)}{2\pi}P_{qg}(z')$$

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

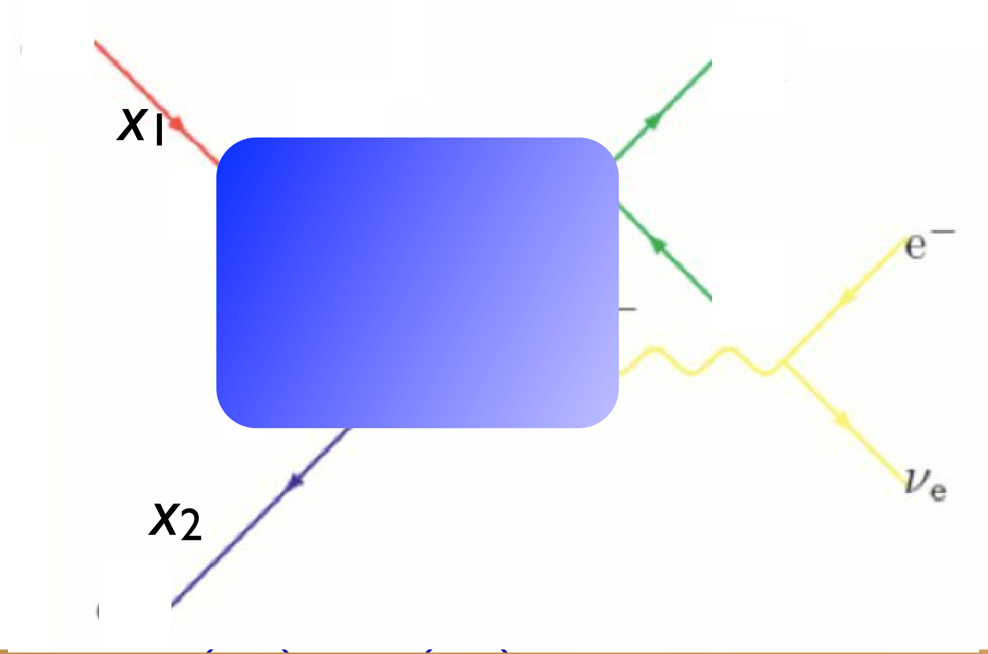
ME with α_s evaluated at the scale of each splitting

PDF reweighting

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

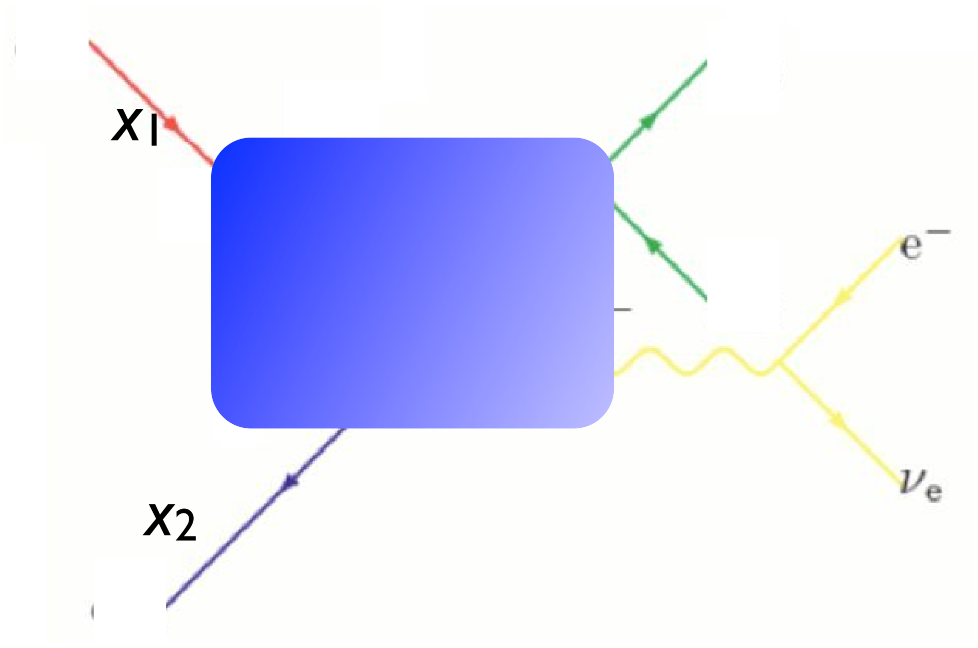


Matching for initial state radiation



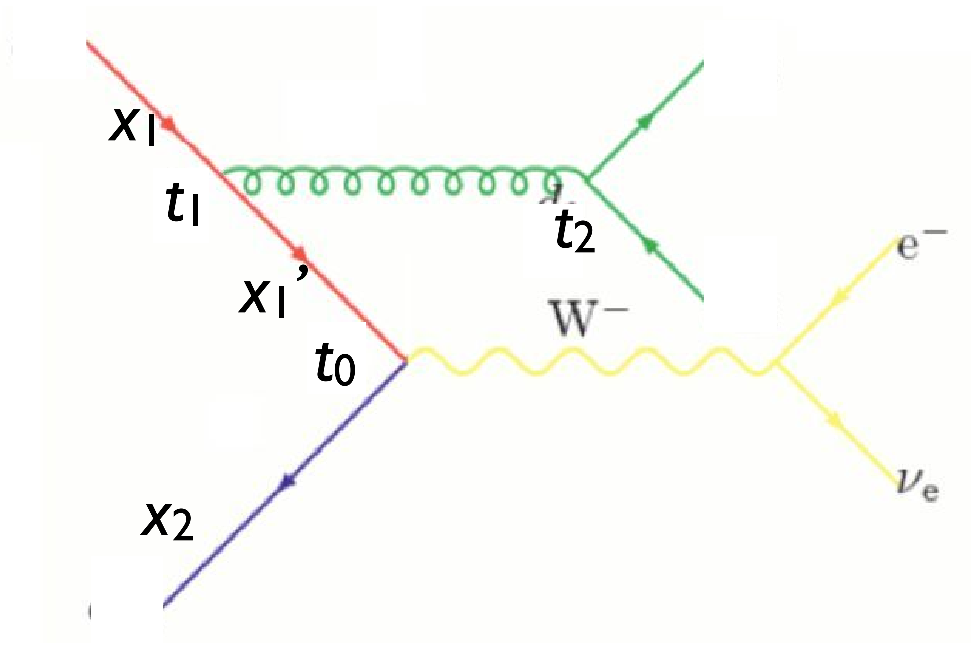
Matching for initial state radiation

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



Matching for initial state radiation

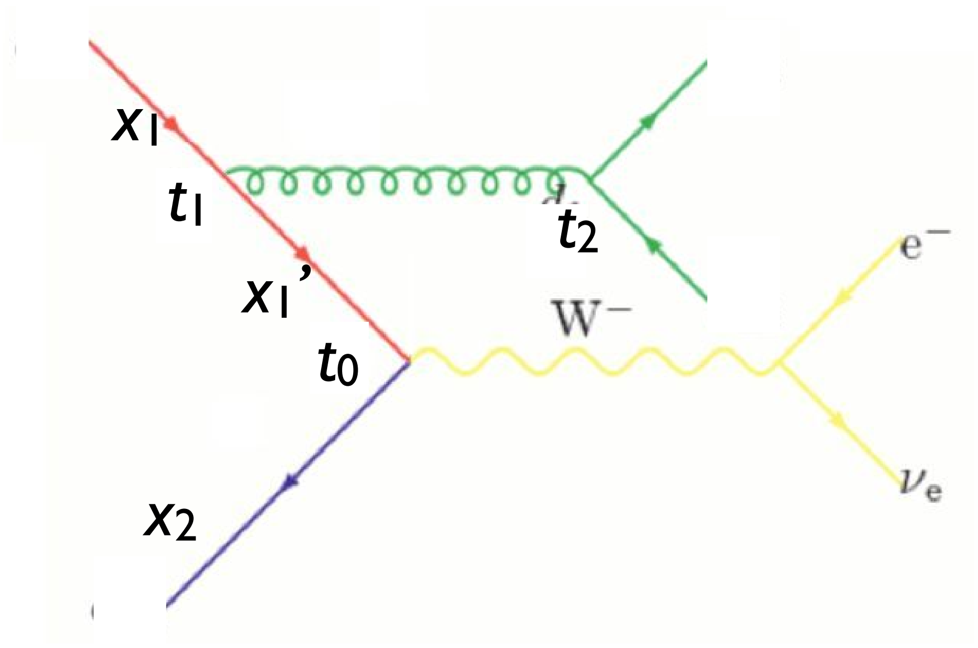
- Again, use a clustering scheme to get a parton shower history



Matching for initial state radiation

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

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

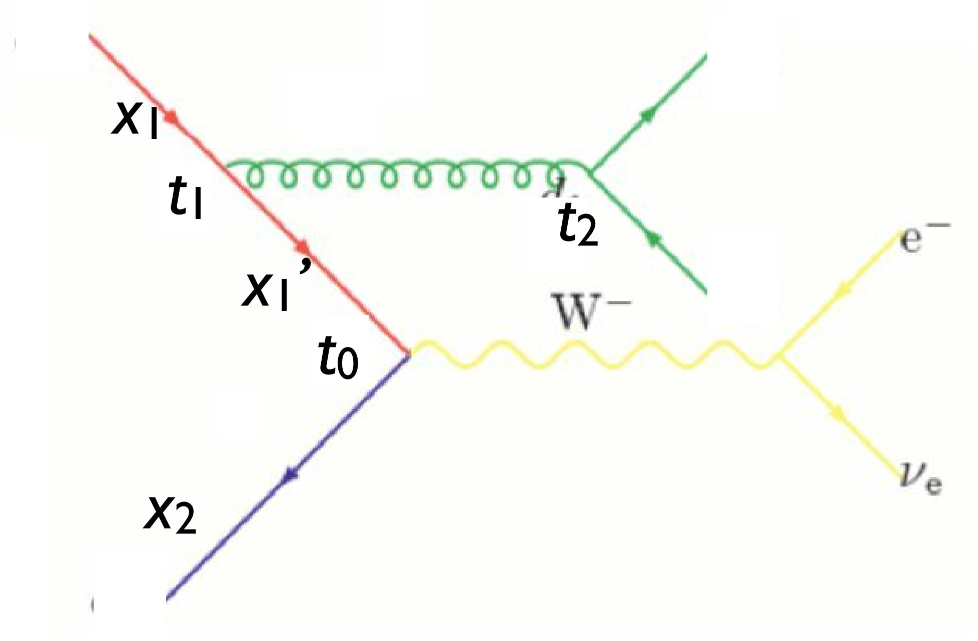


Matching for initial state radiation

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

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

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



K_T clustering schemes

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

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

and for hadron collisions, the minimum of:

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

and

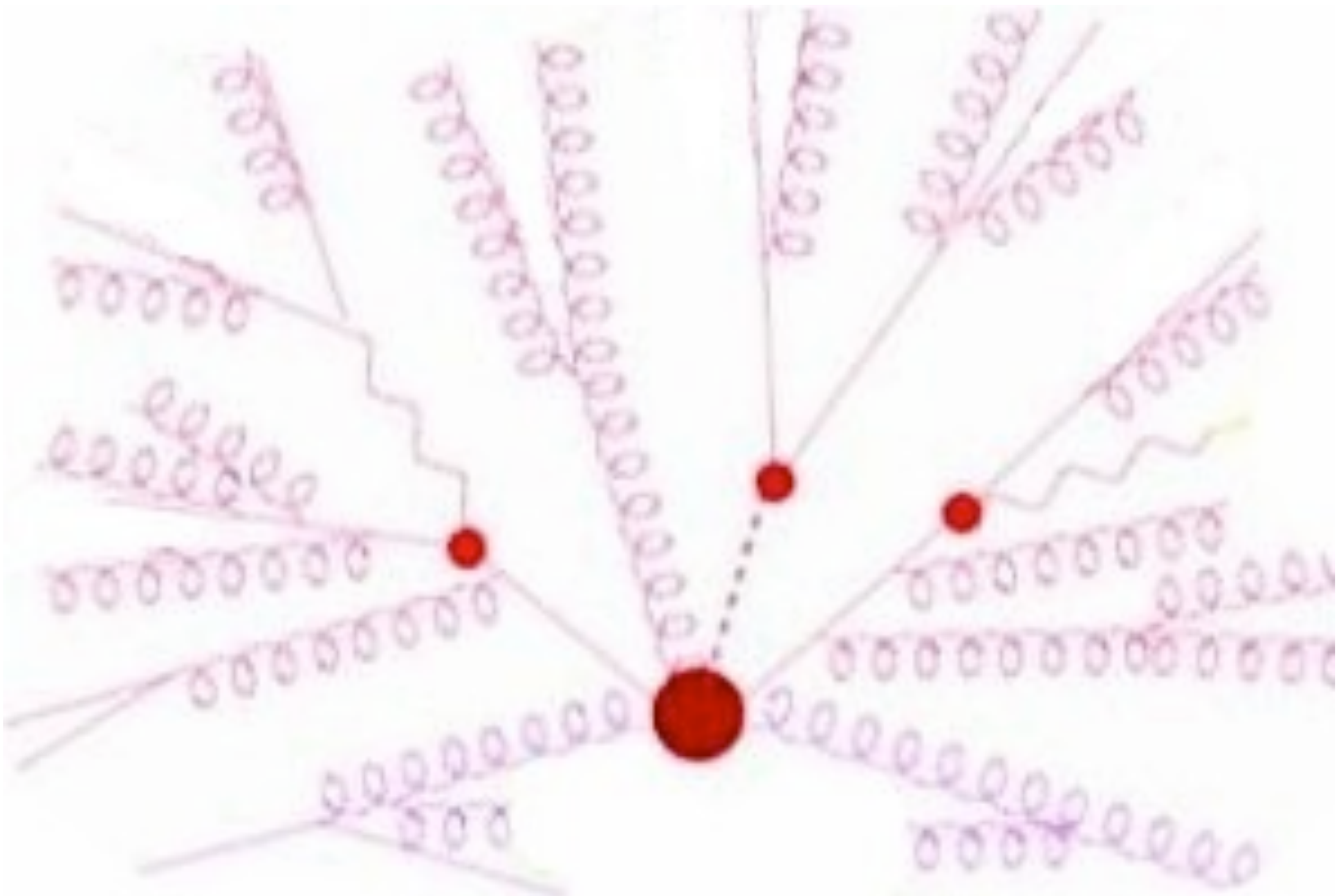
$$k_{Tij}^2 = \max(m_i^2, m_2^2) + \min(p_{Ti}^2, p_{Tj}^2)R_{ij}$$

with

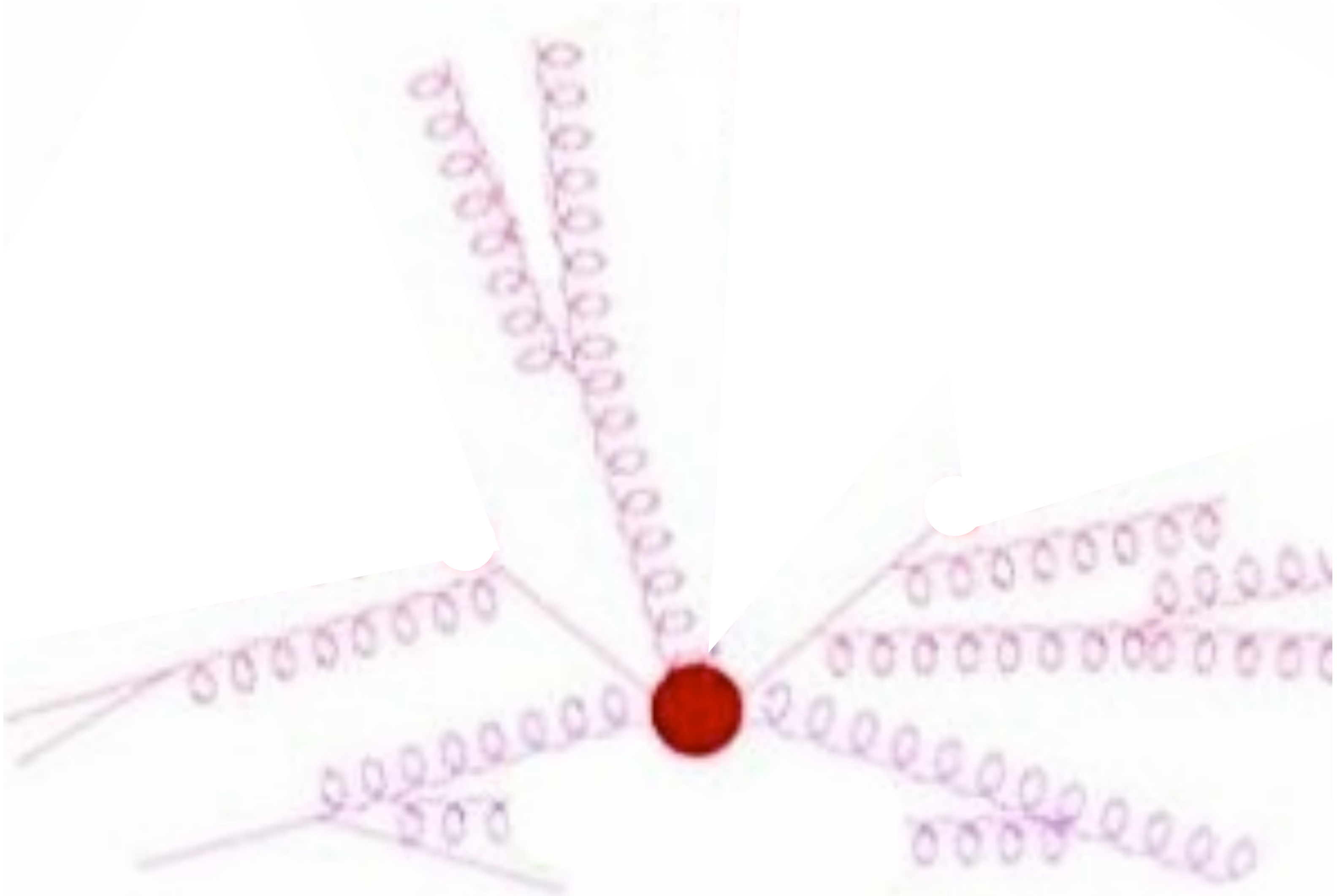
$$R_{ij} = 2[\cosh(y_i - y_j) - \cos(\phi_i - \phi_j)] \simeq (\Delta y)^2 + (\Delta\phi)^2$$

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

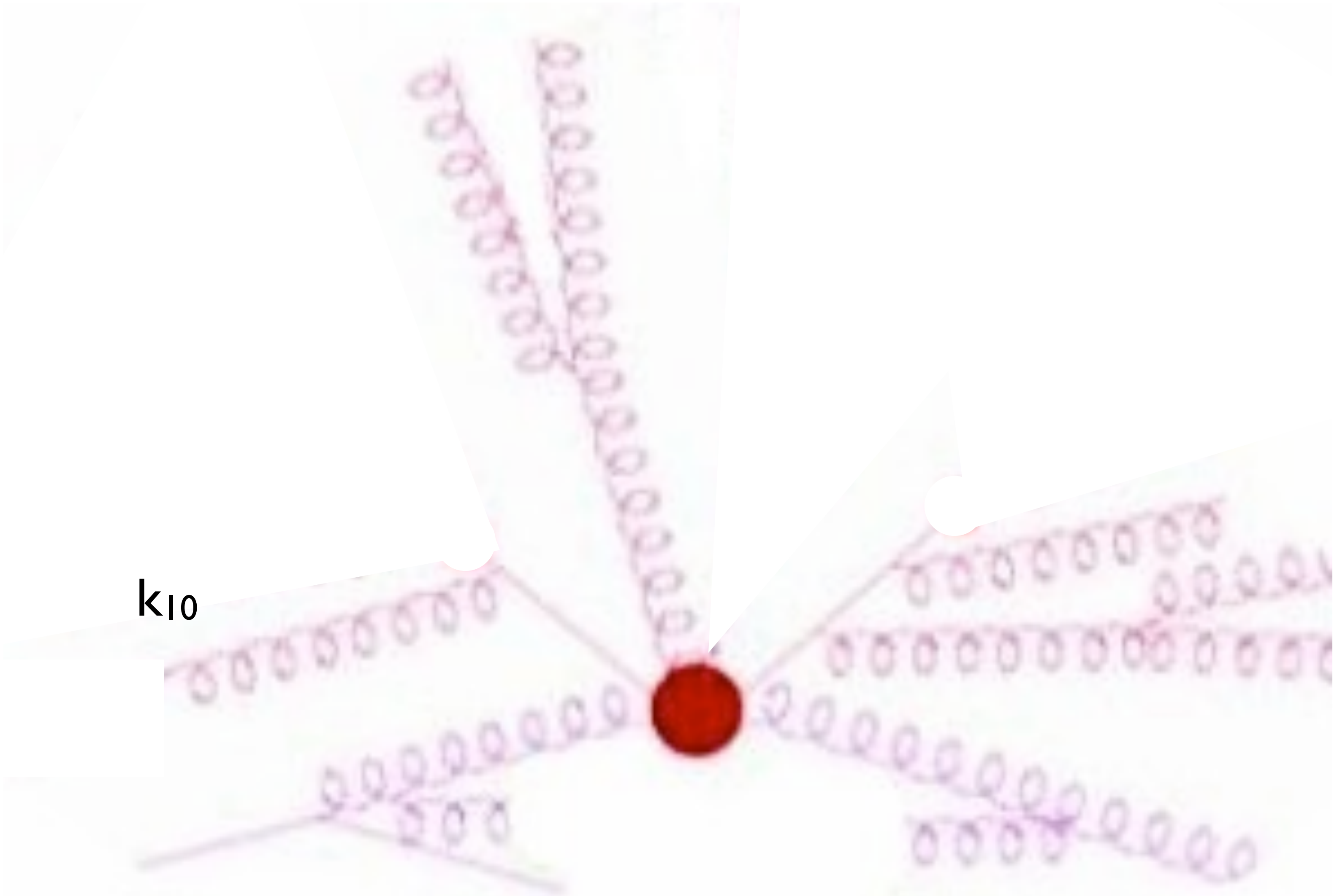
Clustering example



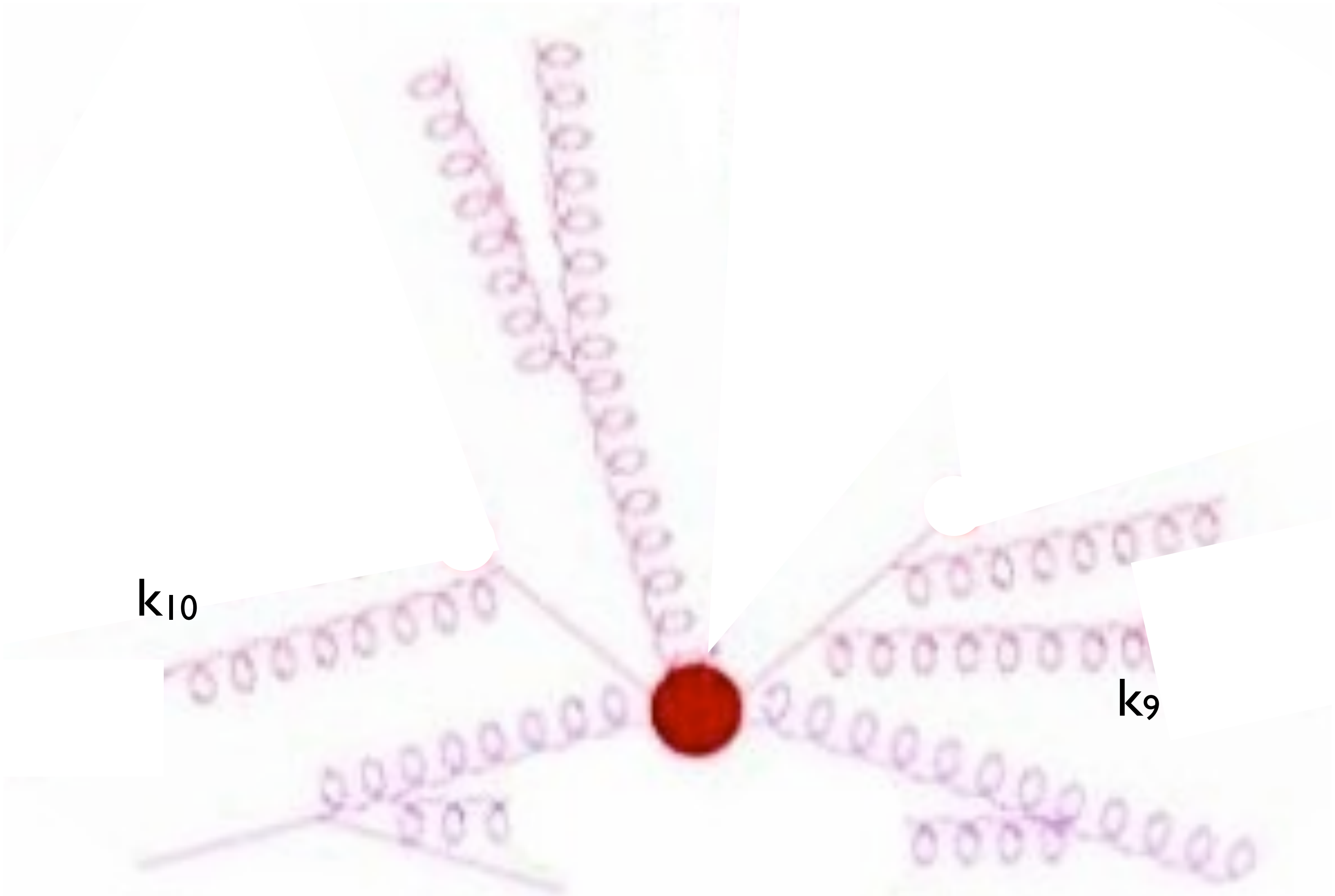
Clustering example



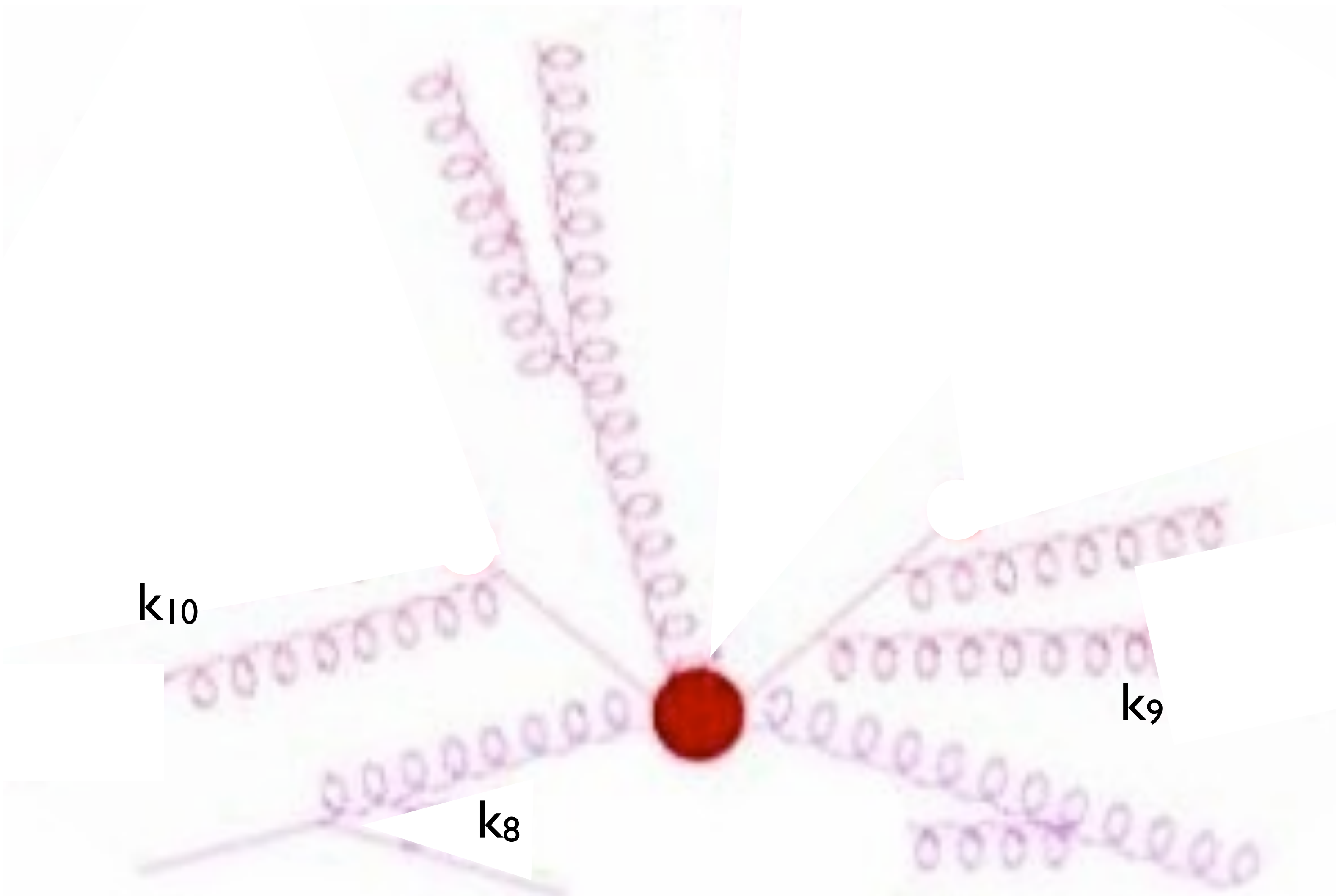
Clustering example



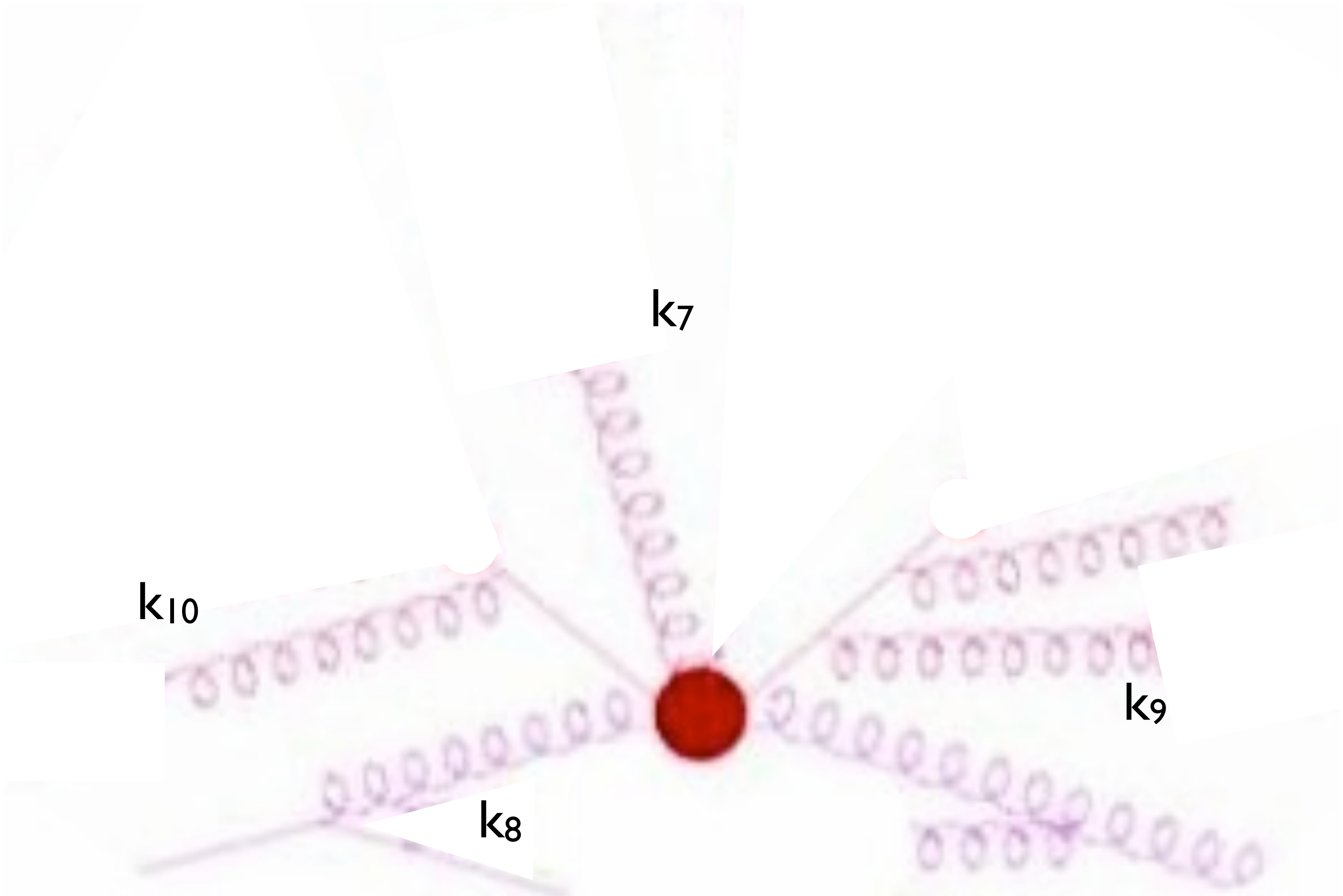
Clustering example



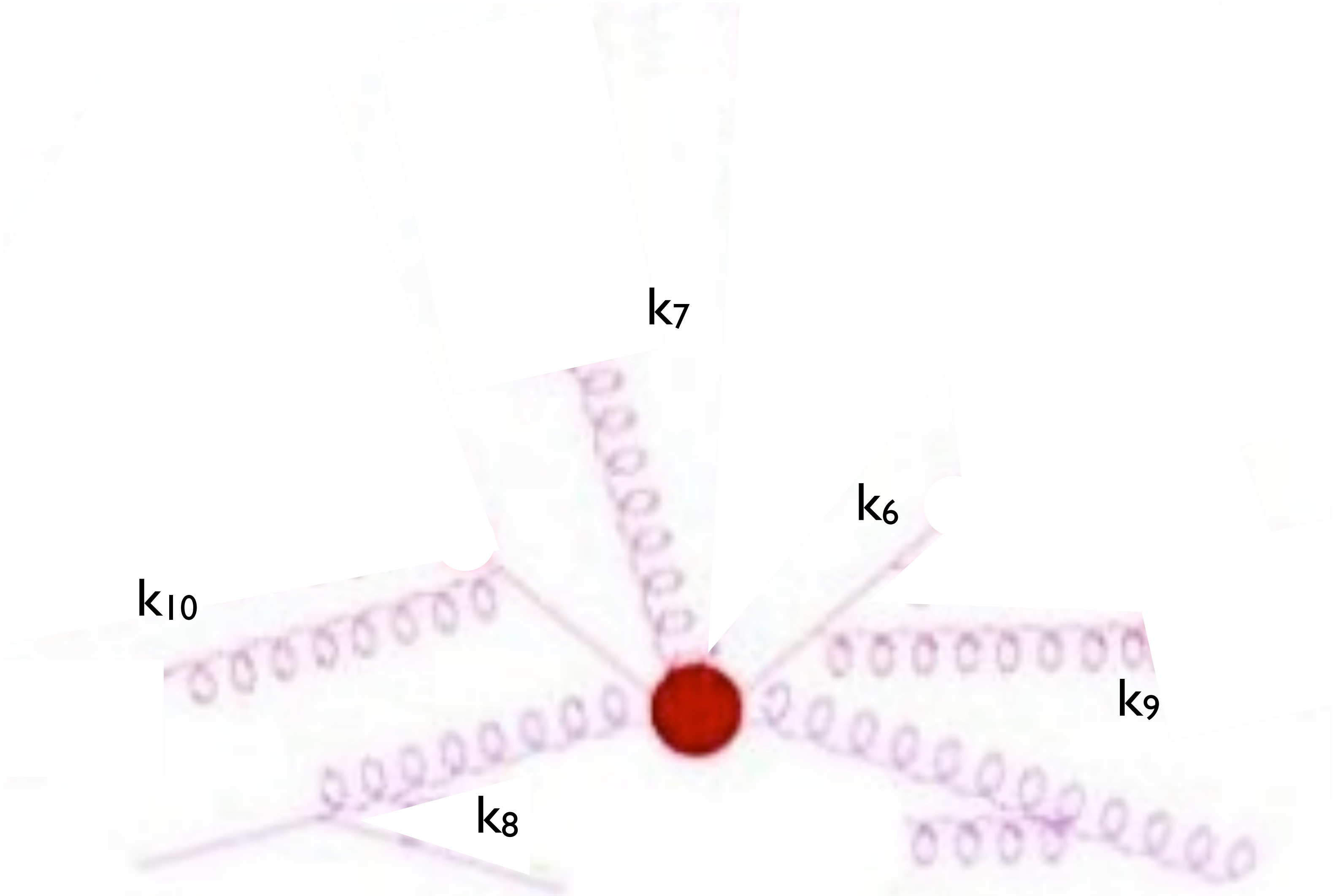
Clustering example



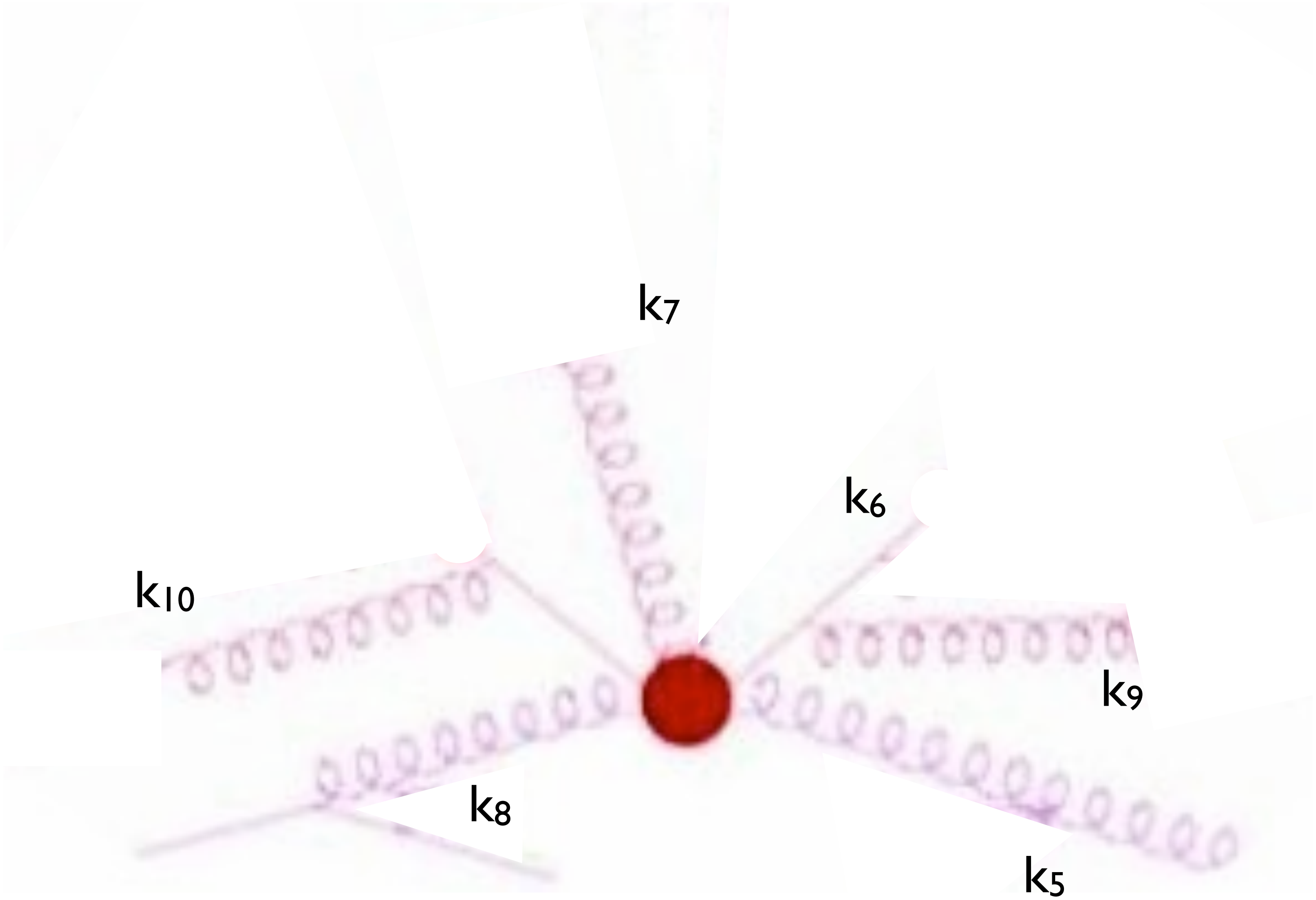
Clustering example



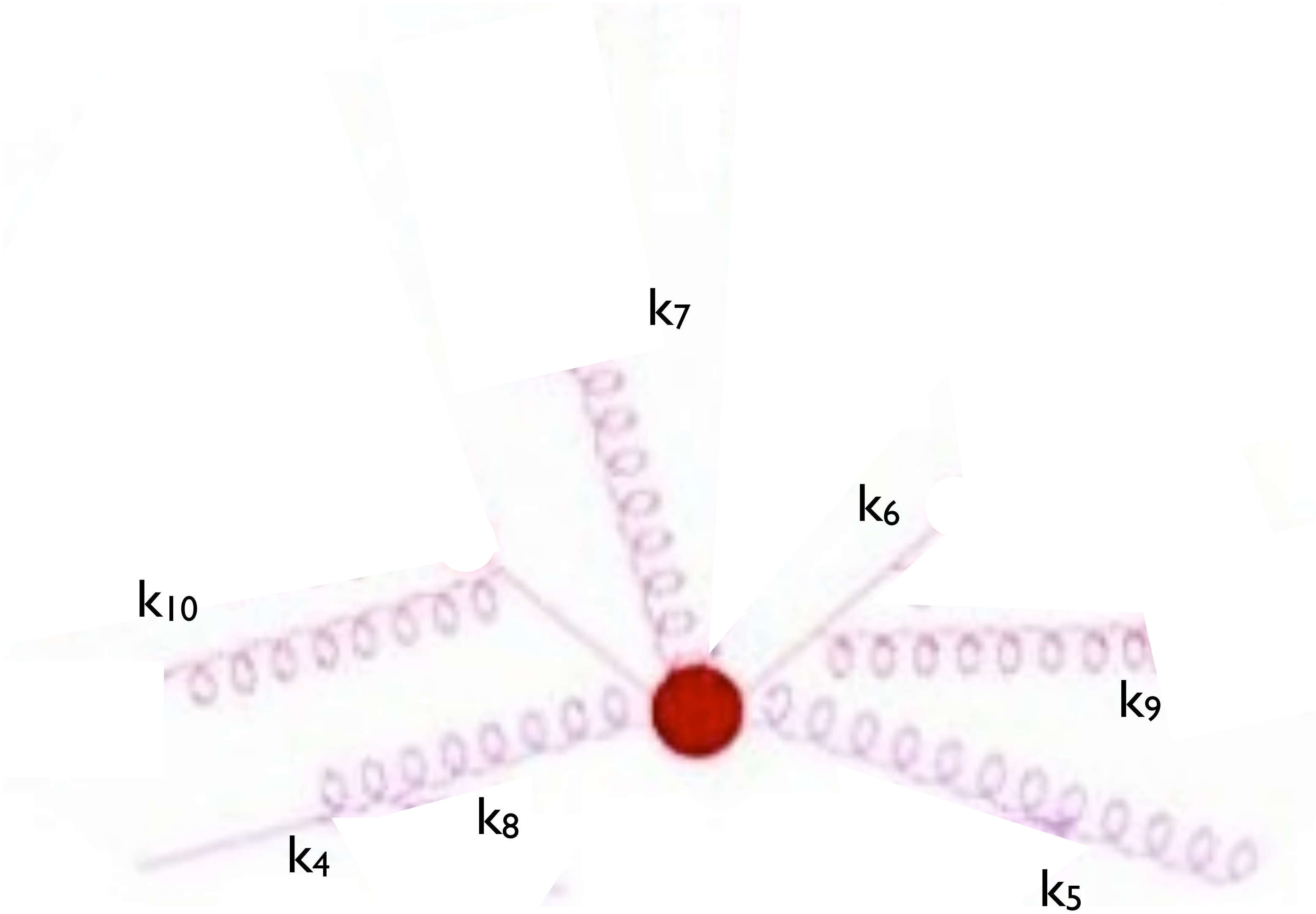
Clustering example



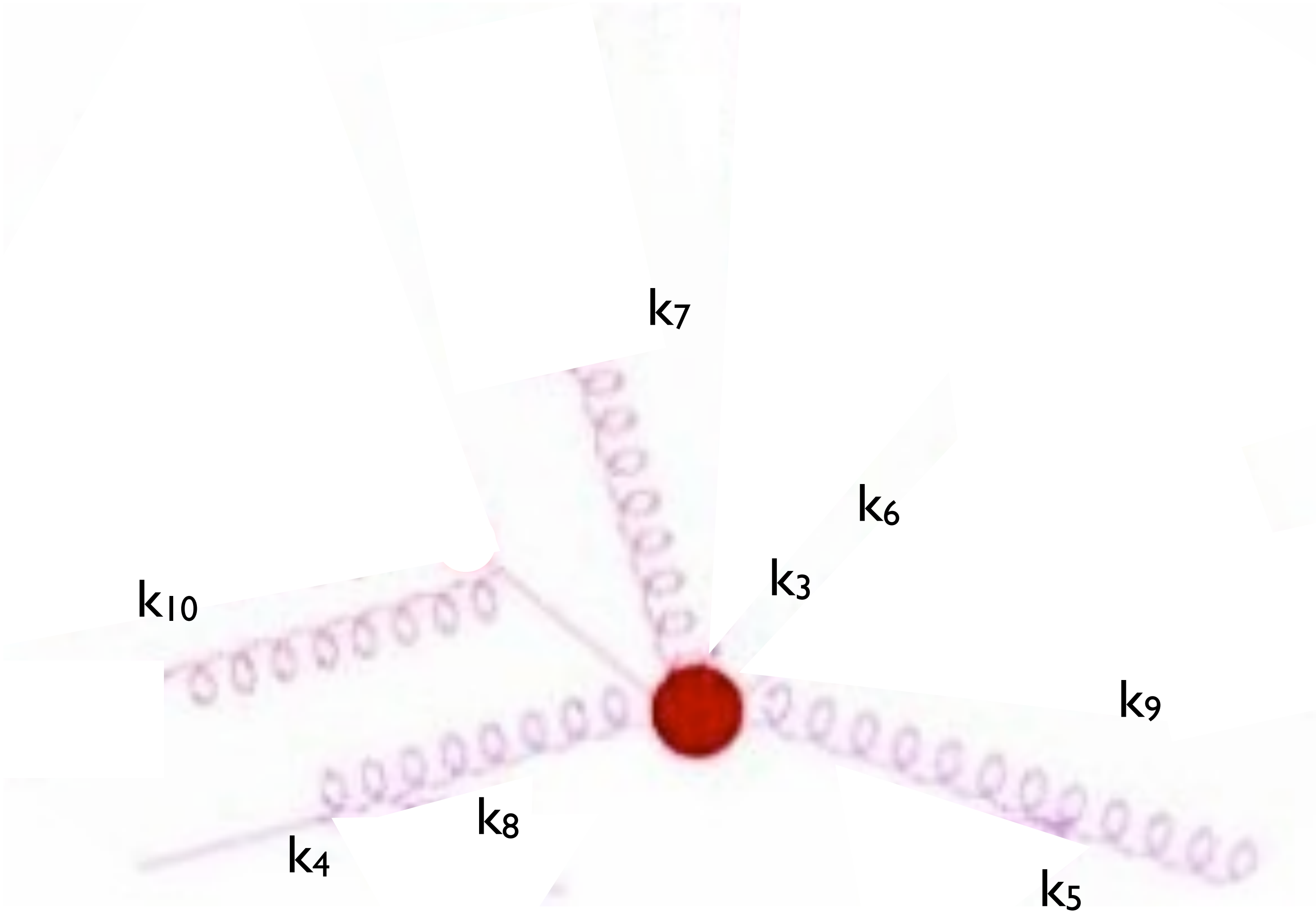
Clustering example



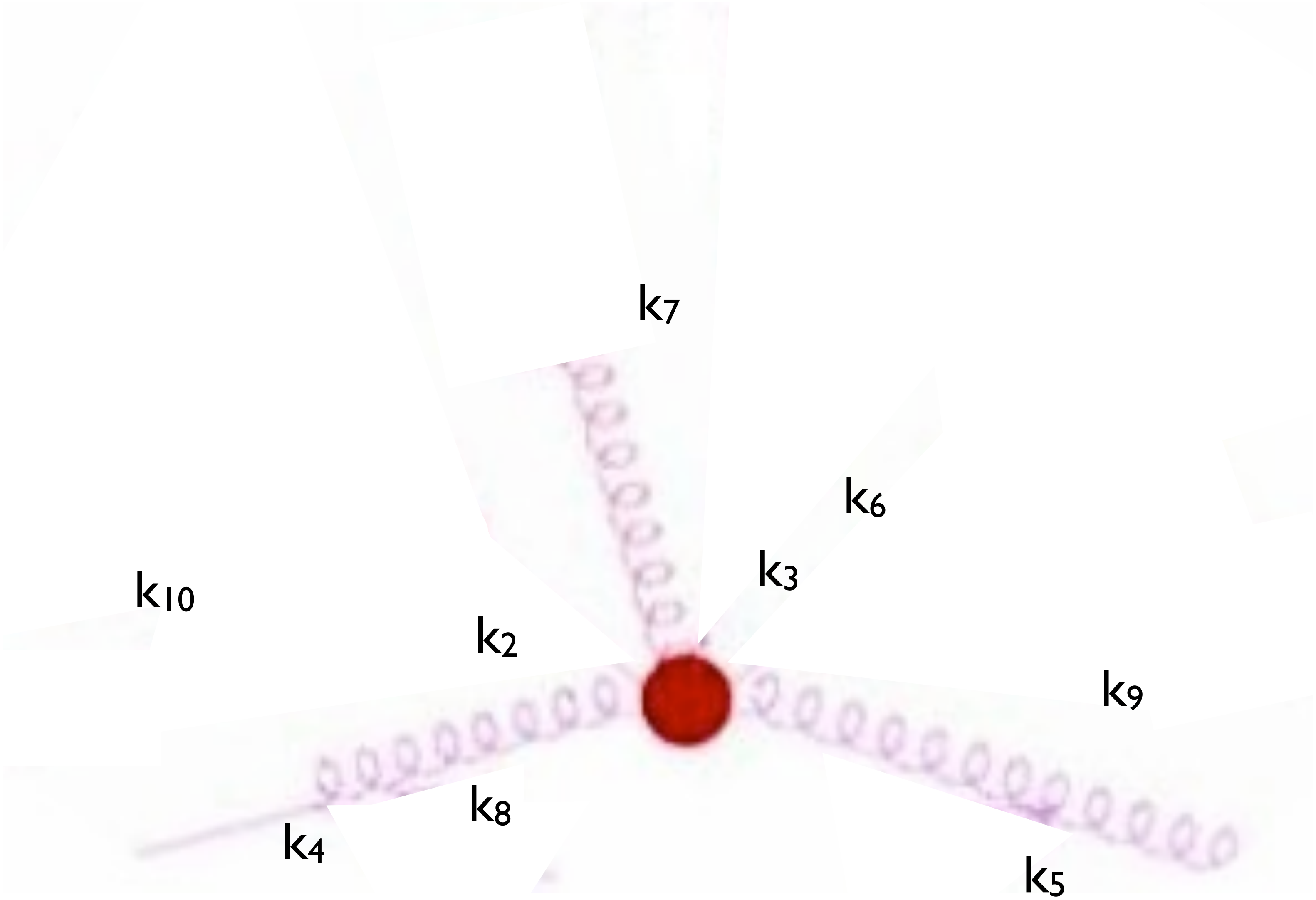
Clustering example



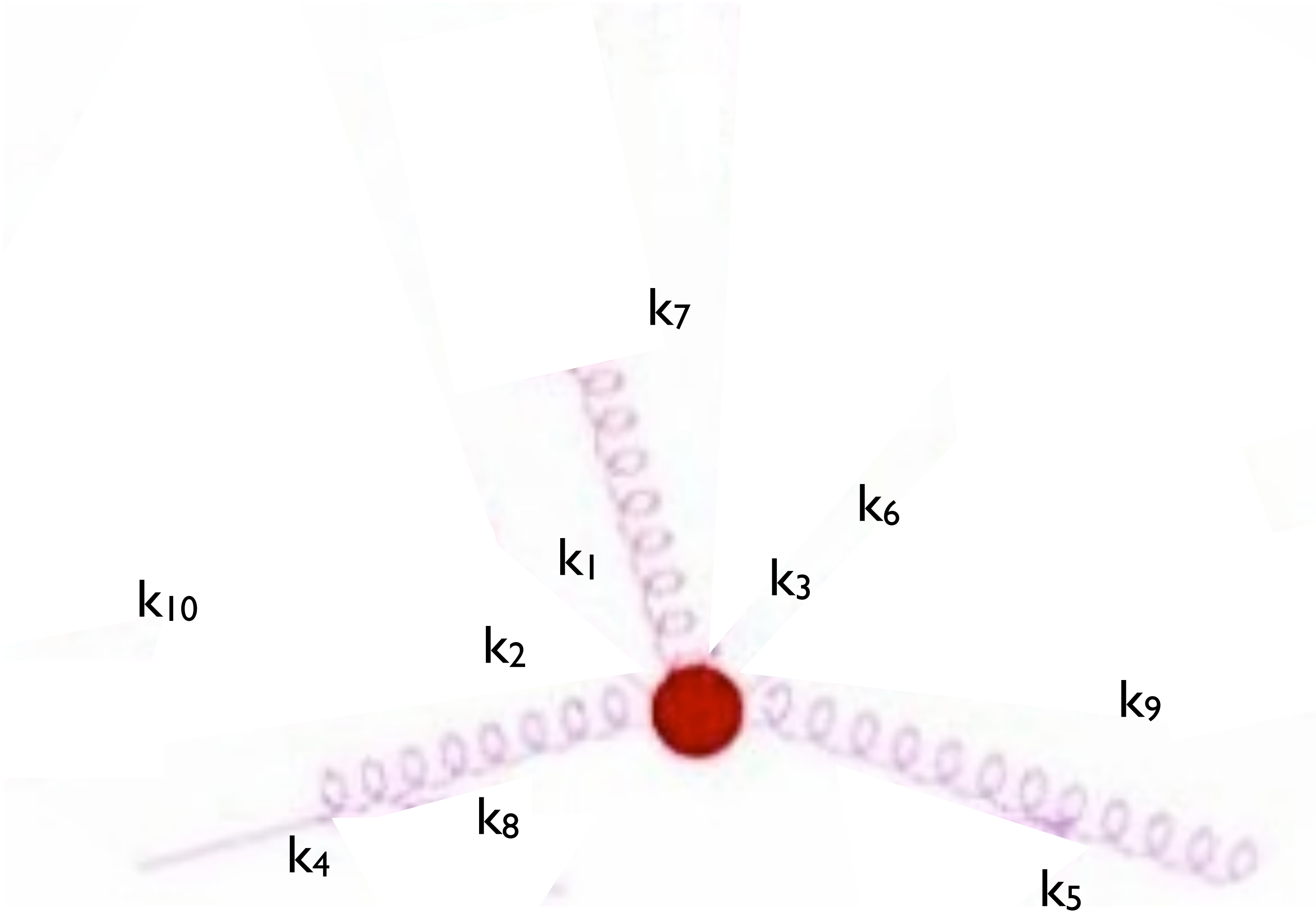
Clustering example



Clustering example



Clustering example



Matching schemes

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

CKKW matching

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

CKKW matching

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

- Apply the required Sudakov suppression

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

analytically, using the best available (NLL) Sudakovs.

CKKW matching

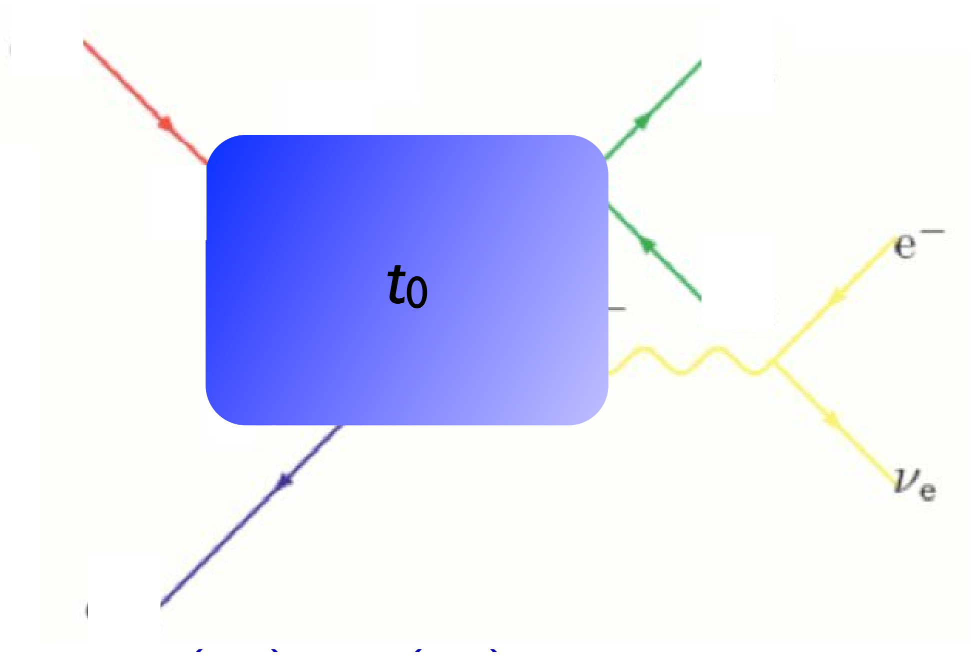
[Catani, Krauss, Kuhn, Webber 2001]
[Krauss 2002]

- Apply the required Sudakov suppression

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

analytically, using the best available (NLL) Sudakovs.

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



CKKW matching

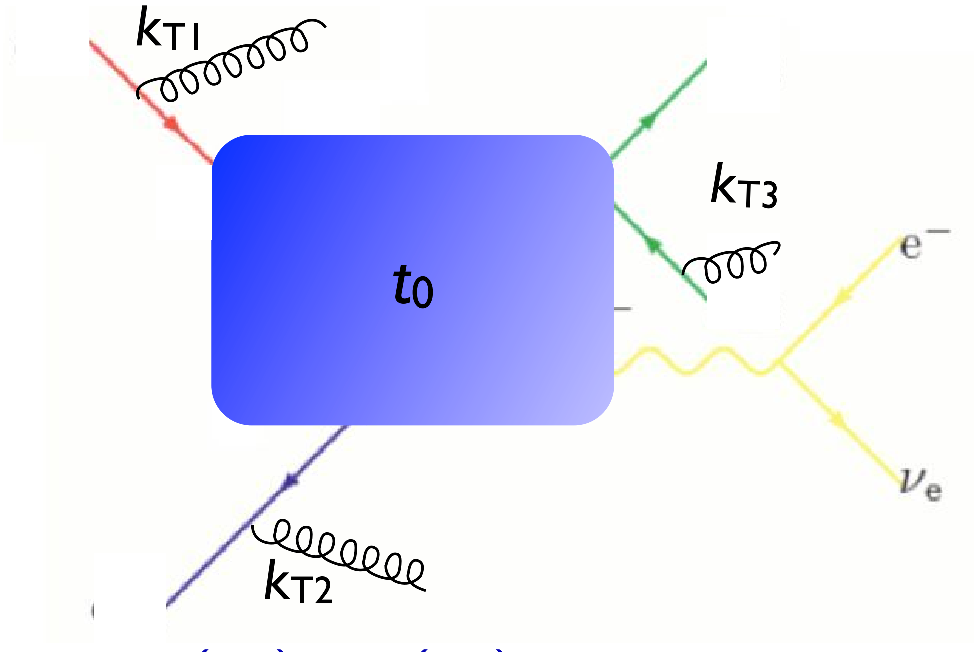
[Catani, Krauss, Kuhn, Webber 2001]
[Krauss 2002]

- Apply the required Sudakov suppression

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

analytically, using the best available (NLL) Sudakovs.

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



CKKW matching

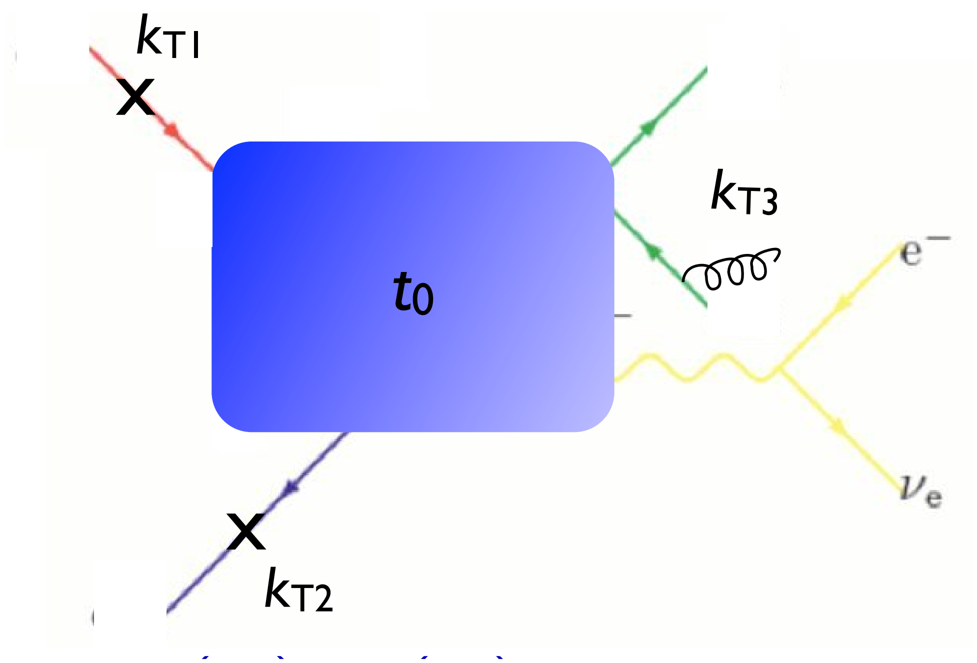
[Catani, Krauss, Kuhn, Webber 2001]
[Krauss 2002]

- Apply the required Sudakov suppression

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

analytically, using the best available (NLL) Sudakovs.

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



CKKW matching

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

- Apply the required Sudakov suppression

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

analytically, using the best available (NLL) Sudakovs.

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

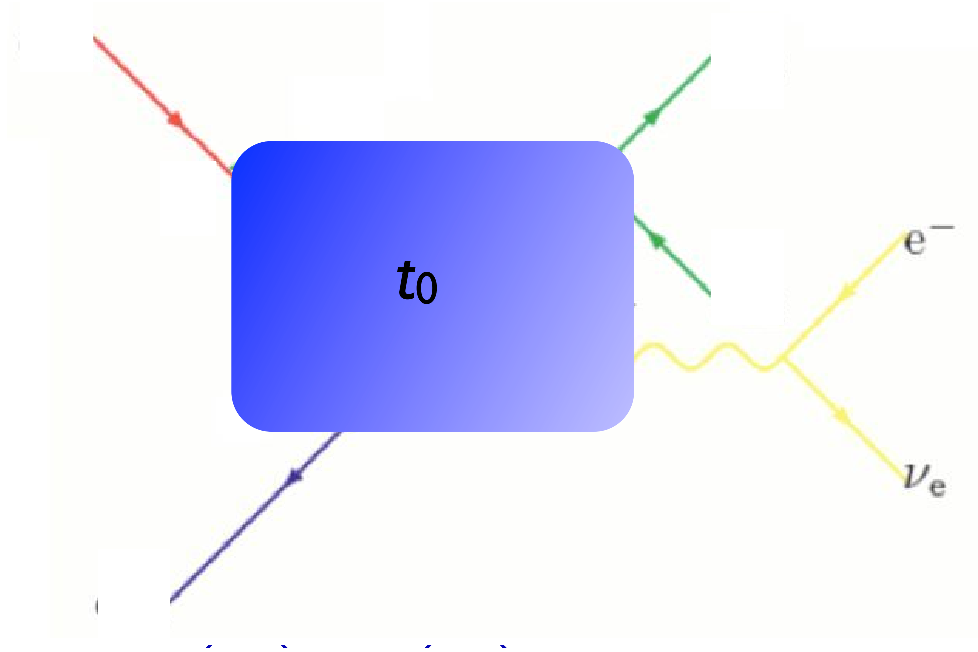
✓ Best theoretical treatment of matrix element

- Requires dedicated PS implementation
- Mismatch between analytical Sudakov and (non-NLL) shower

- Implemented in Sherpa (v. 1.1)

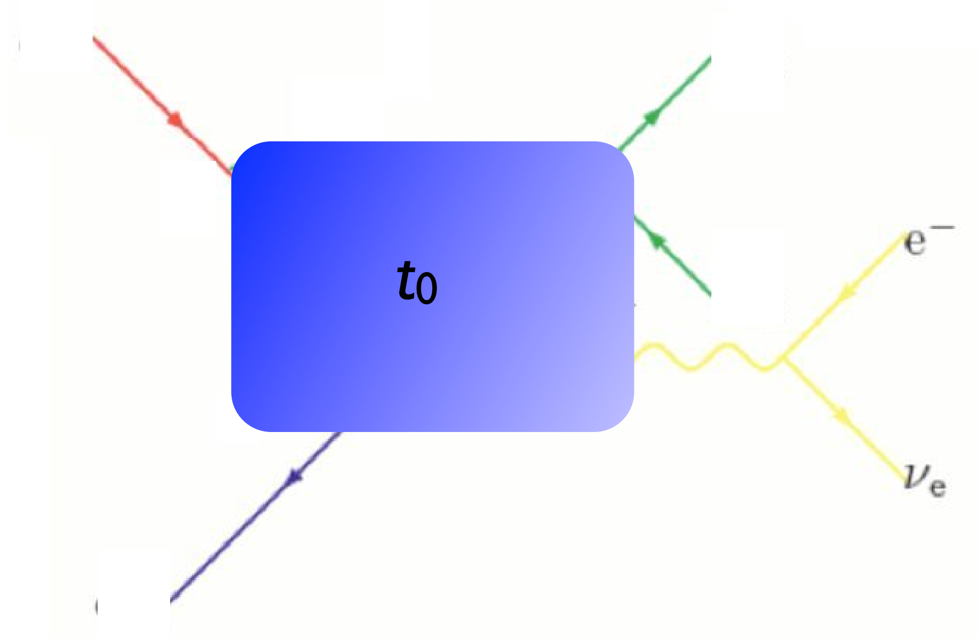
CKKW-L matching

[Lönnblad 2002]
[Hoeche et al. 2009]



CKKW-L matching

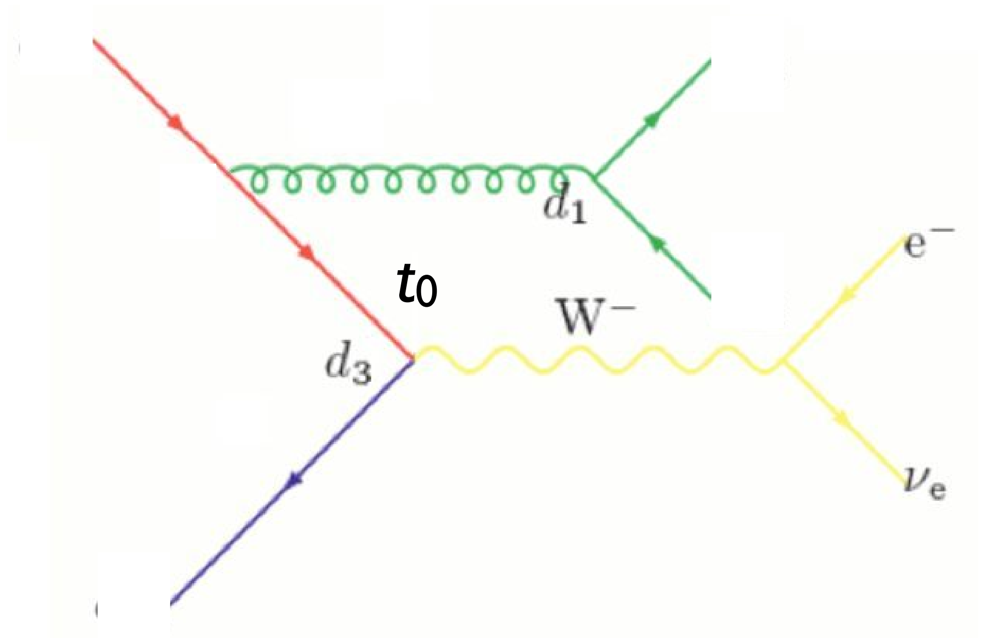
[Lönnblad 2002]
[Hoeche et al. 2009]



- Cluster back to "parton shower history"

CKKW-L matching

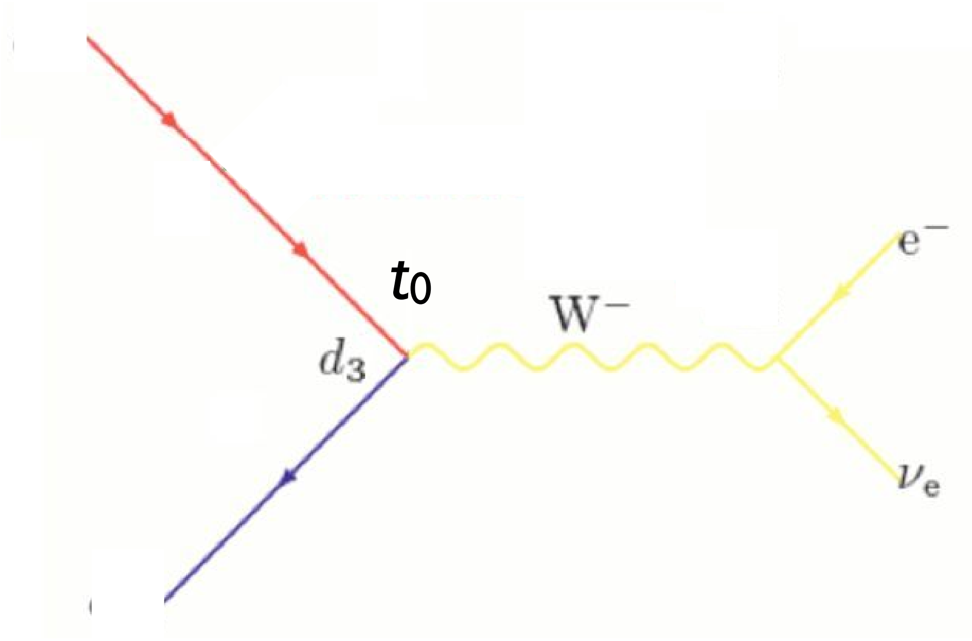
[Lönnblad 2002]
[Hoeche et al. 2009]



- Cluster back to "parton shower history"

CKKW-L matching

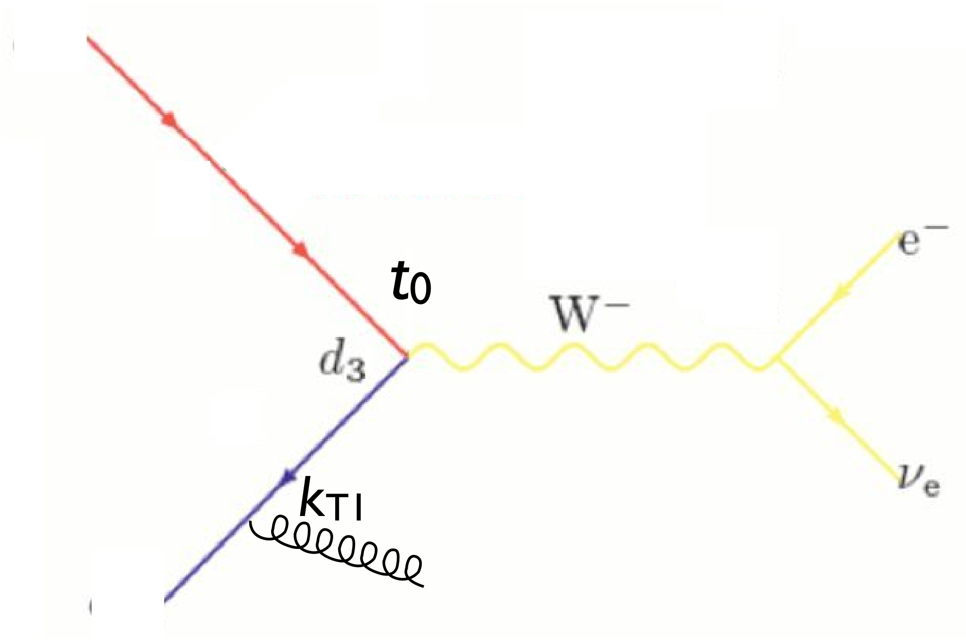
[Lönnblad 2002]
[Hoeche et al. 2009]



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

CKKW-L matching

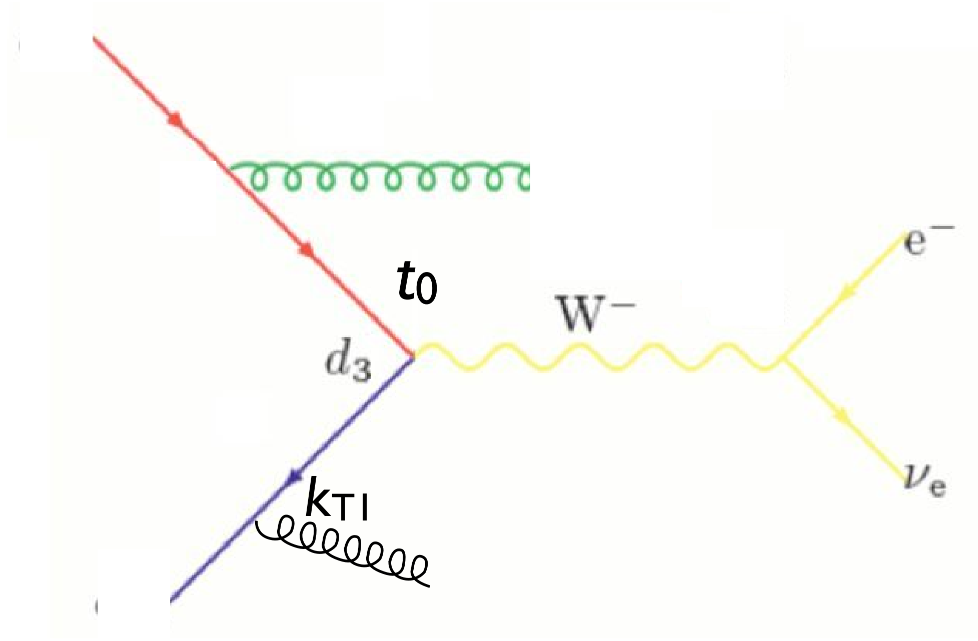
[Lönnblad 2002]
[Hoeche et al. 2009]



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

CKKW-L matching

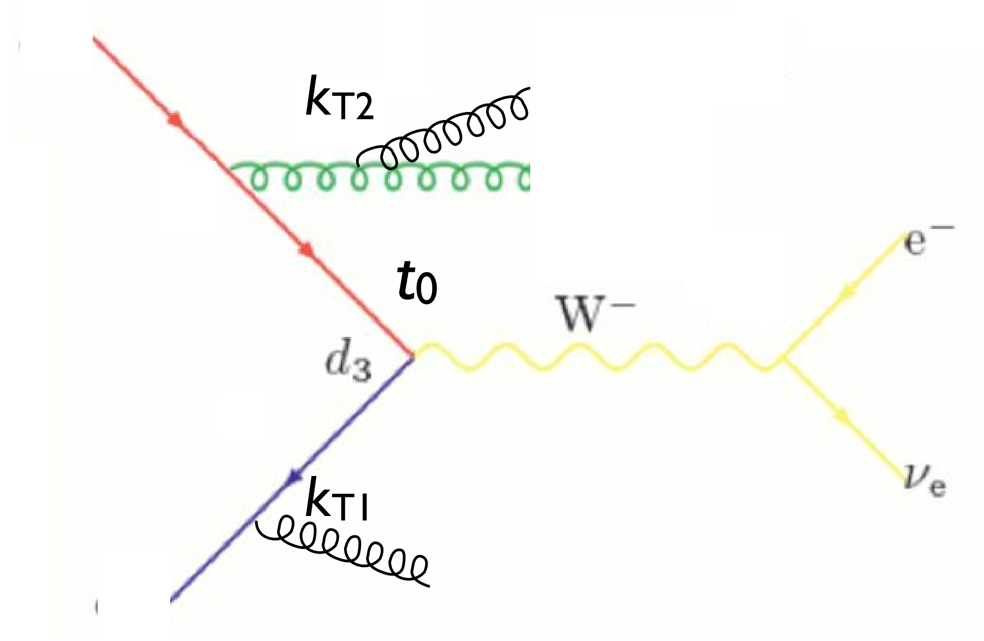
[Lönnblad 2002]
[Hoeche et al. 2009]



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

CKKW-L matching

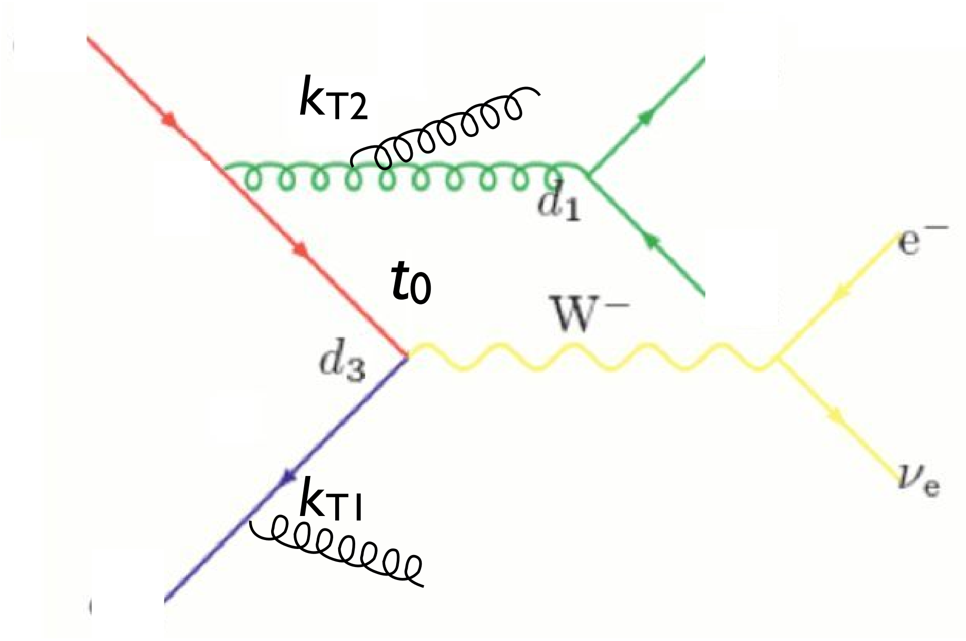
[Lönnblad 2002]
[Hoeche et al. 2009]



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

CKKW-L matching

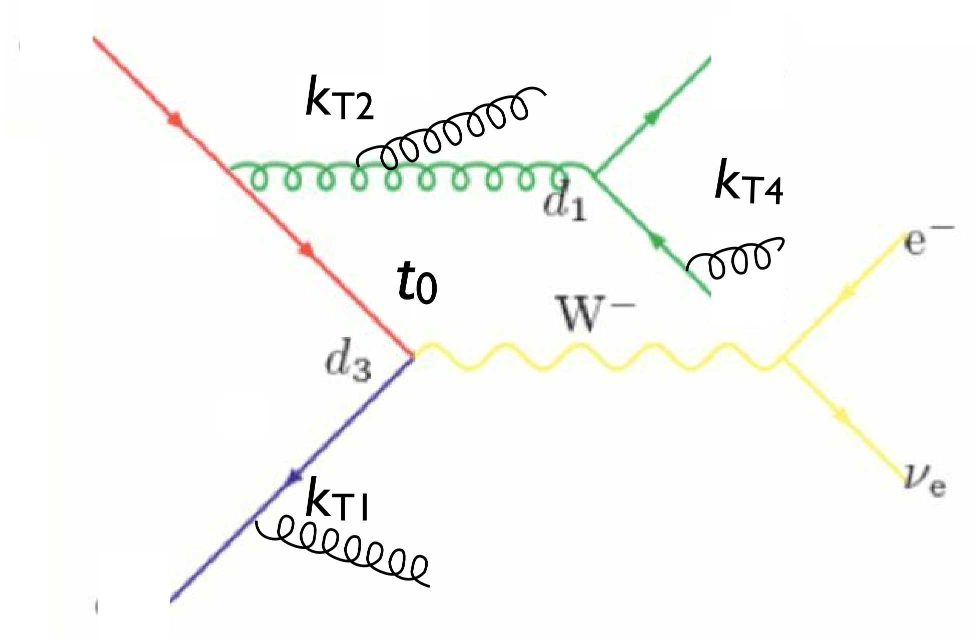
[Lönnblad 2002]
[Hoeche et al. 2009]



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

CKKW-L matching

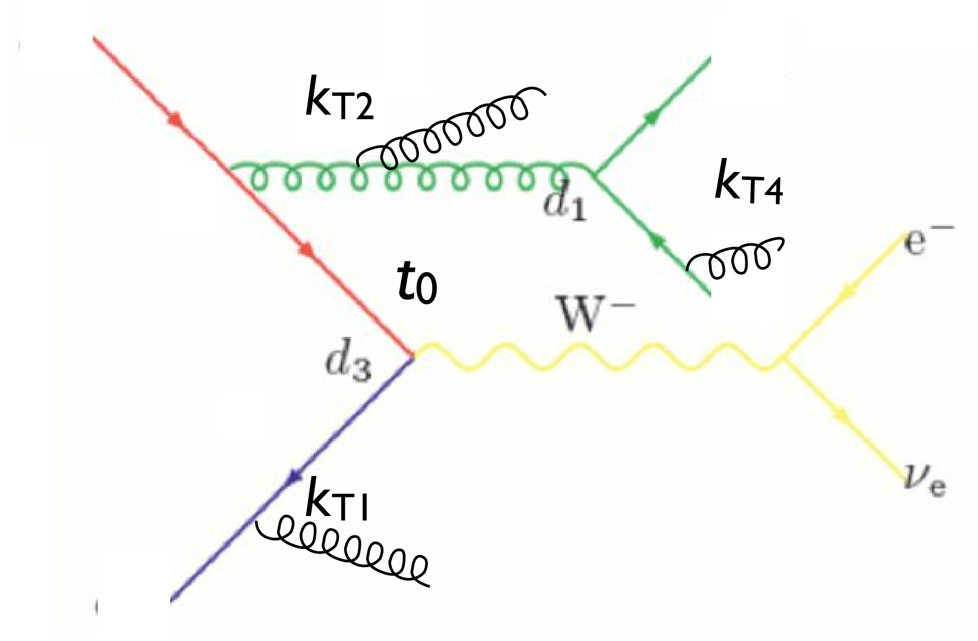
[Lönnblad 2002]
[Hoeche et al. 2009]



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

CKKW-L matching

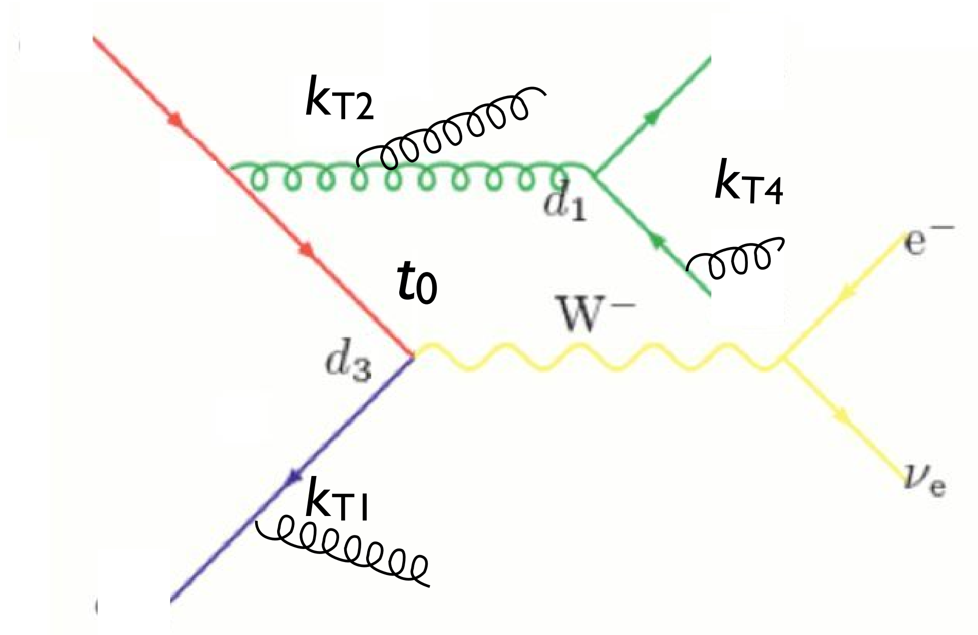
[Lönnblad 2002]
[Hoeche et al. 2009]



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

CKKW-L matching

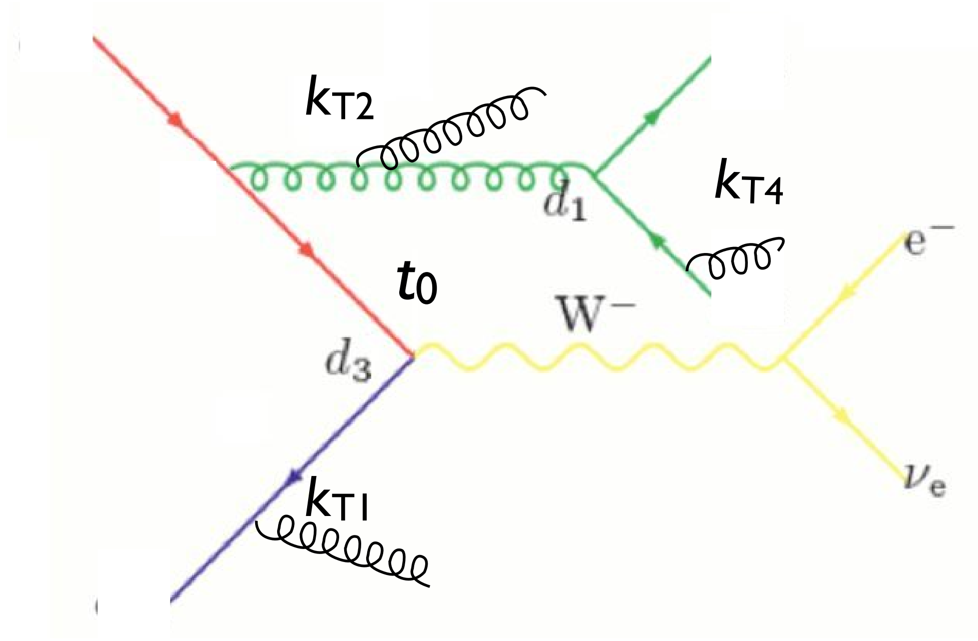
[Lönnblad 2002]
[Hoeche et al. 2009]



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

CKKW-L matching

[Lönnblad 2002]
[Hoeche et al. 2009]



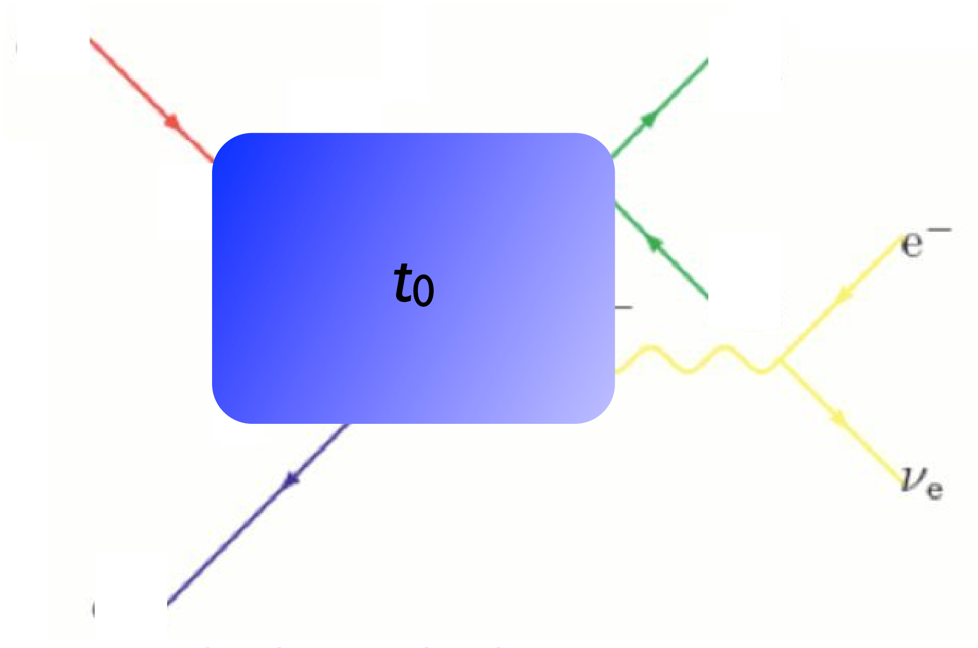
- Cluster back to "parton shower history"
- Perform showering step-by-step for each step in the parton shower history, starting from the clustering scale for that step

✓ Automatic agreement between Sudakov and shower

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

MLM matching

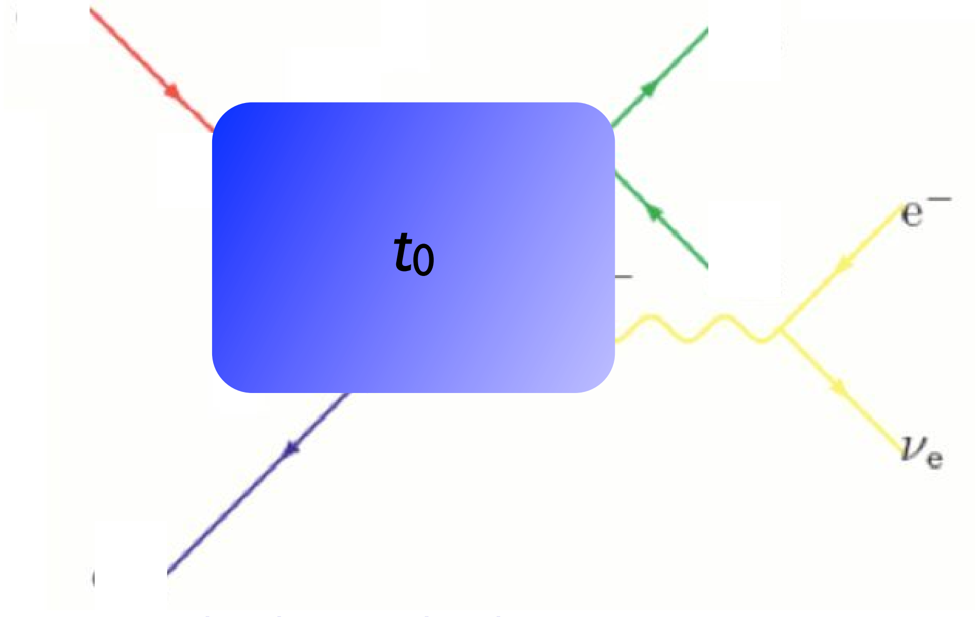
[M.L. Mangano, ~2002, 2007]
[J.A. et al 2007, 2008]



MLM matching

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

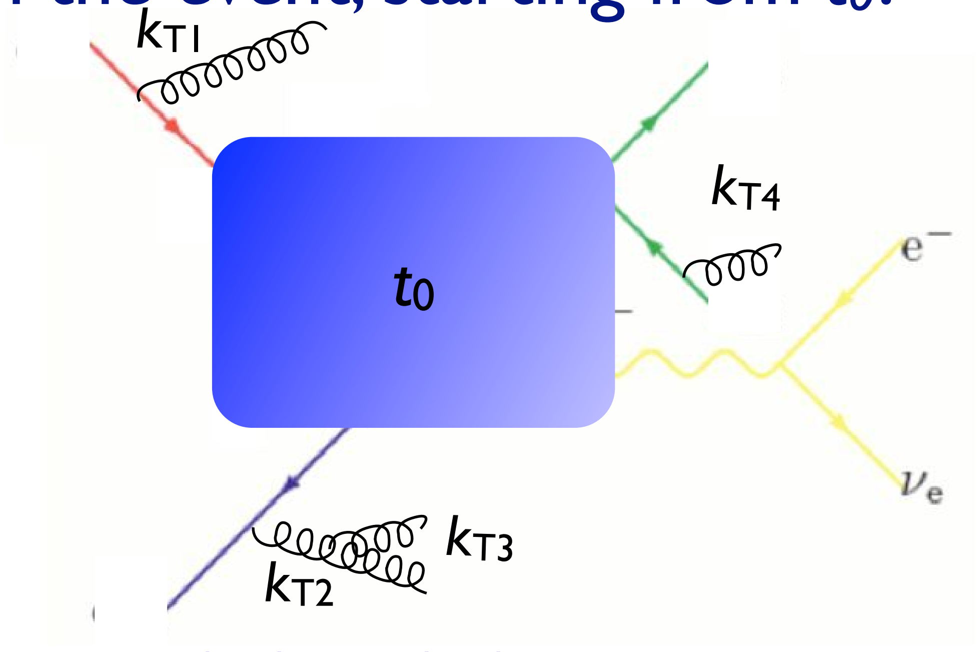
- The simplest way to do the Sudakov suppression is to run the shower on the event, starting from t_0!



MLM matching

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

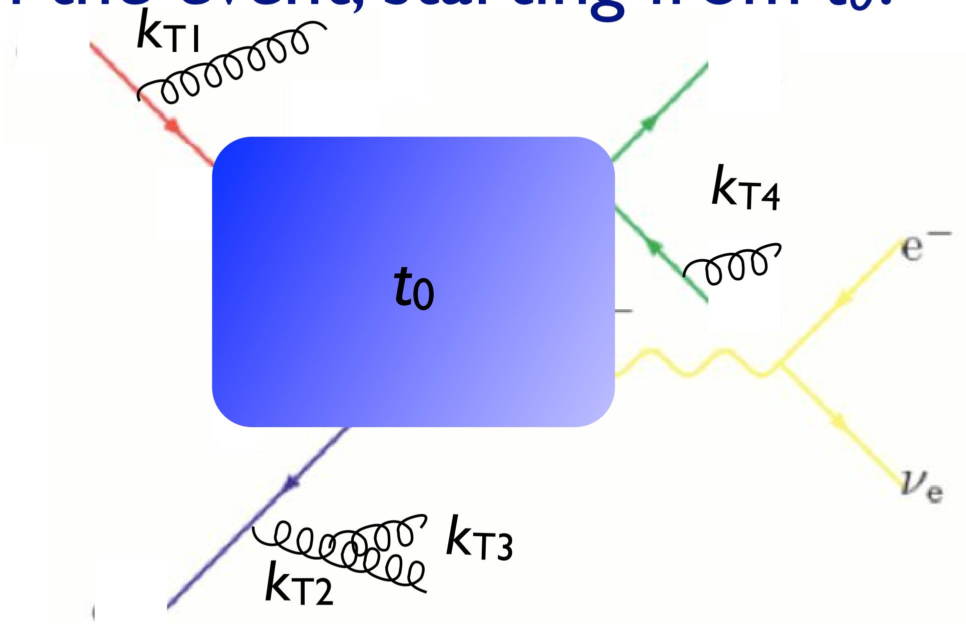
- The simplest way to do the Sudakov suppression is to run the shower on the event, starting from t_0!



MLM matching

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

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

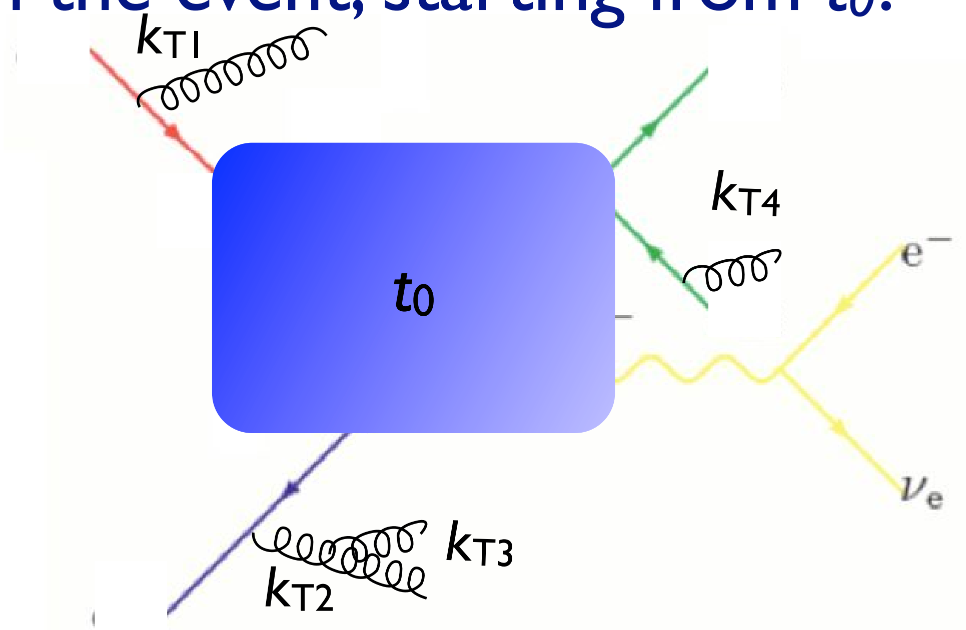


- Perform jet clustering after PS - if hardest jet $k_{T1} > t_{cut}$ or there are jets not matched to partons, reject the event

MLM matching

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

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

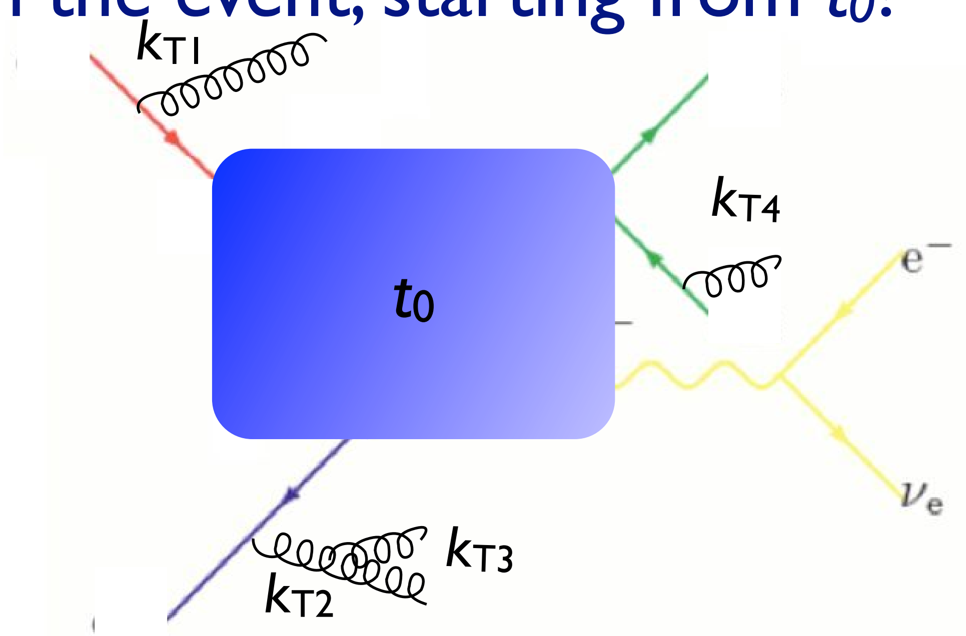


- Perform jet clustering after PS - if hardest jet $k_{T1} > t_{cut}$ or there are jets not matched to partons, reject the event
- The resulting Sudakov suppression from the procedure is
$$(\Delta_{Iq}(t_{\rm cut}, t_0))^2(\Delta_q(t_{\rm cut}, t_0))^2$$
which turns out to be a good enough approximation of the correct expression $(\Delta_{Iq}(t_{\rm cut}, t_0))^2\Delta_g(t_2, t_1)(\Delta_q(t_{\rm cut}, t_2))^2$

MLM matching

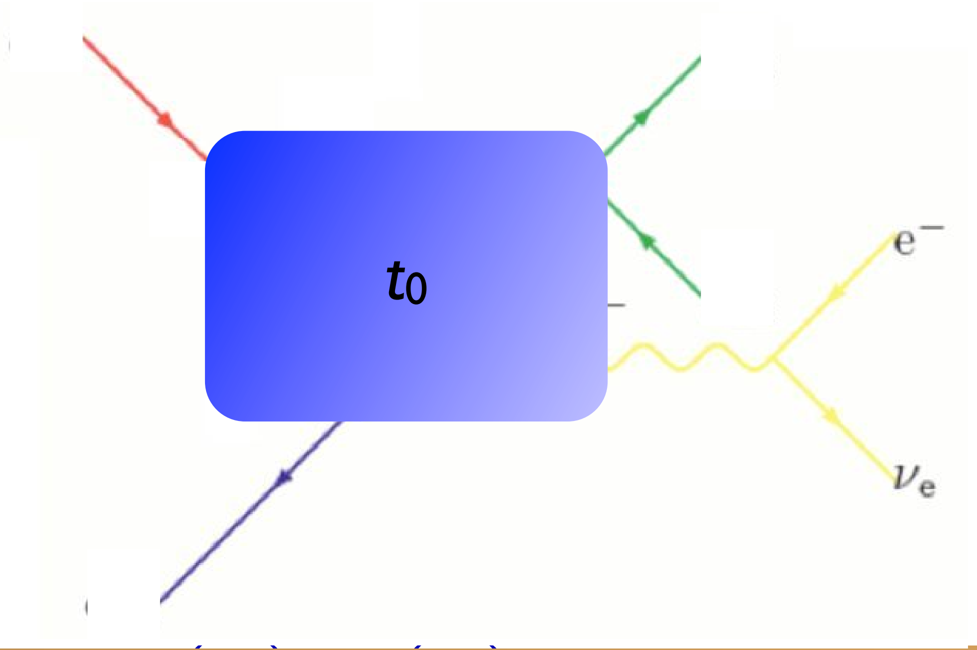
[M.L. Mangano, ~2002, 2007]
[J.A. et al 2007, 2008]

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



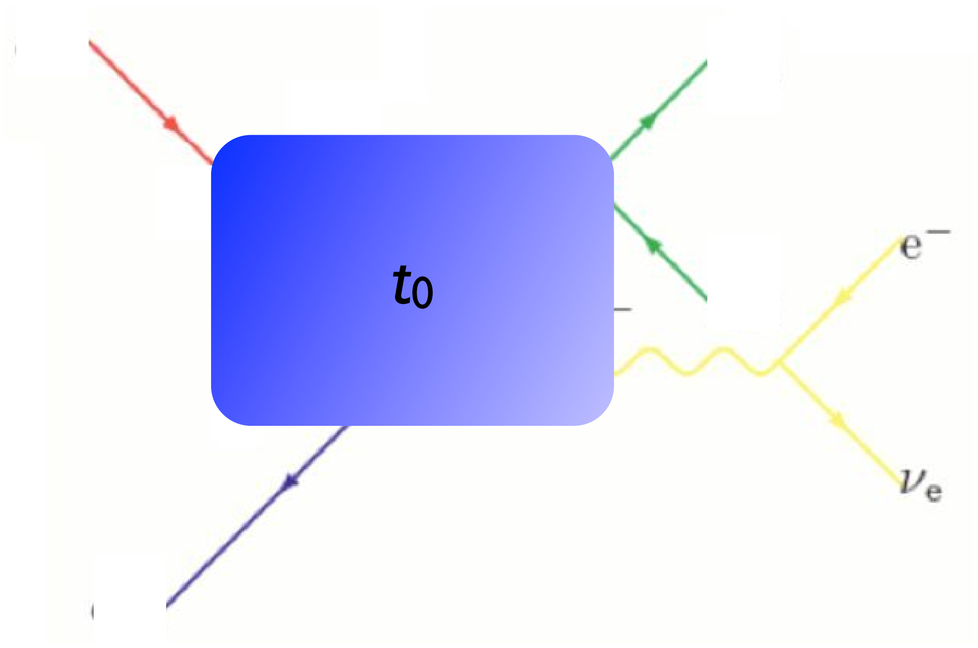
- Perform jet clustering after PS - if hardest jet $k_{T1} > t_{cut}$ or
- ✓ Simplest available scheme
- ✓ Allows matching with any shower, without modification
- ➡ Sudakov suppression not exact, minor mismatch with shower
- Implemented in AlpGen, HELAC, MadGraph+Pythia 6

MLM matching



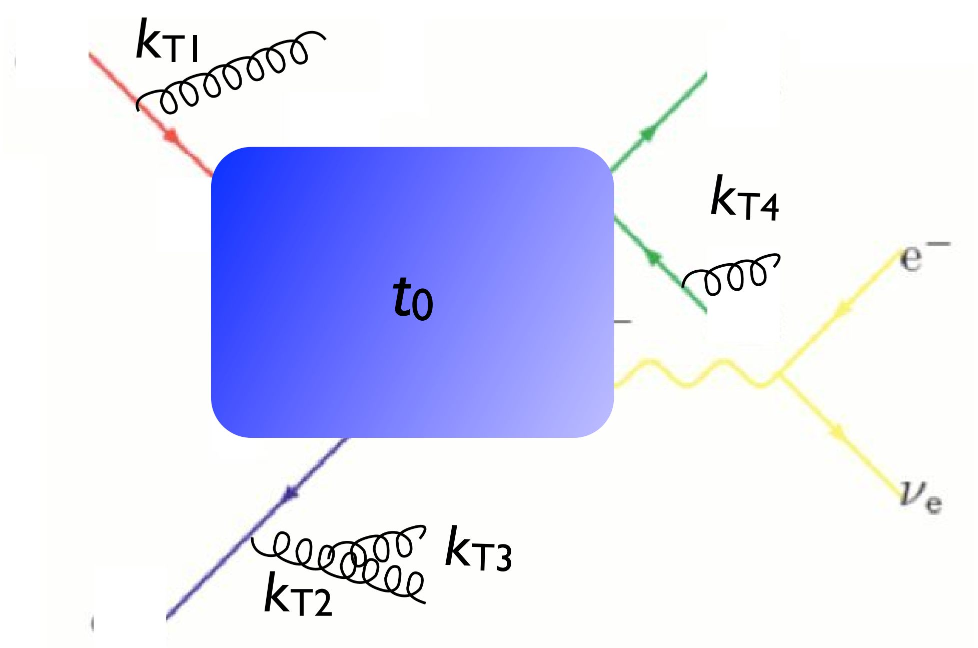
MLM matching

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



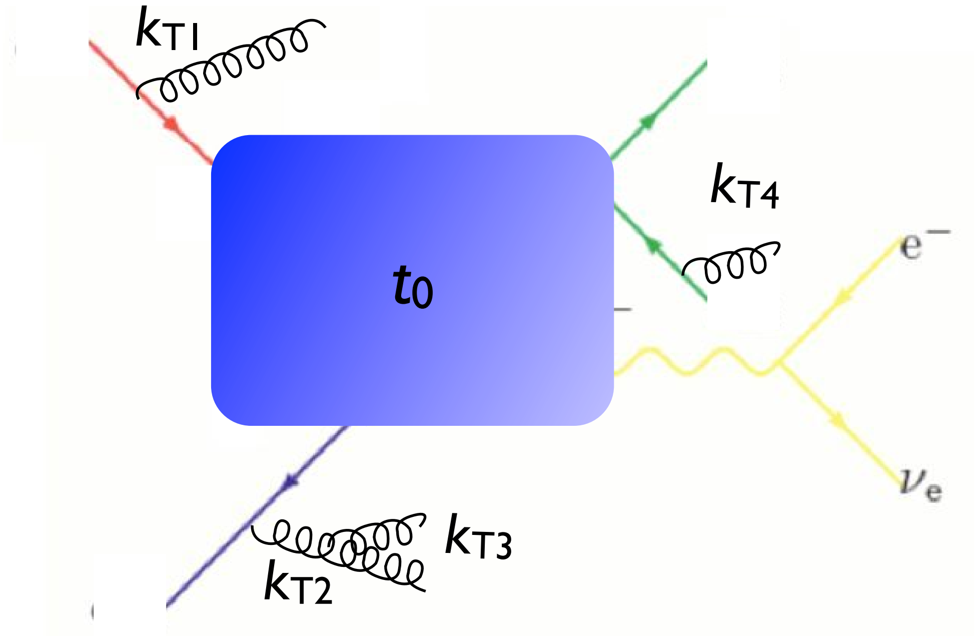
MLM matching

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



MLM matching

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



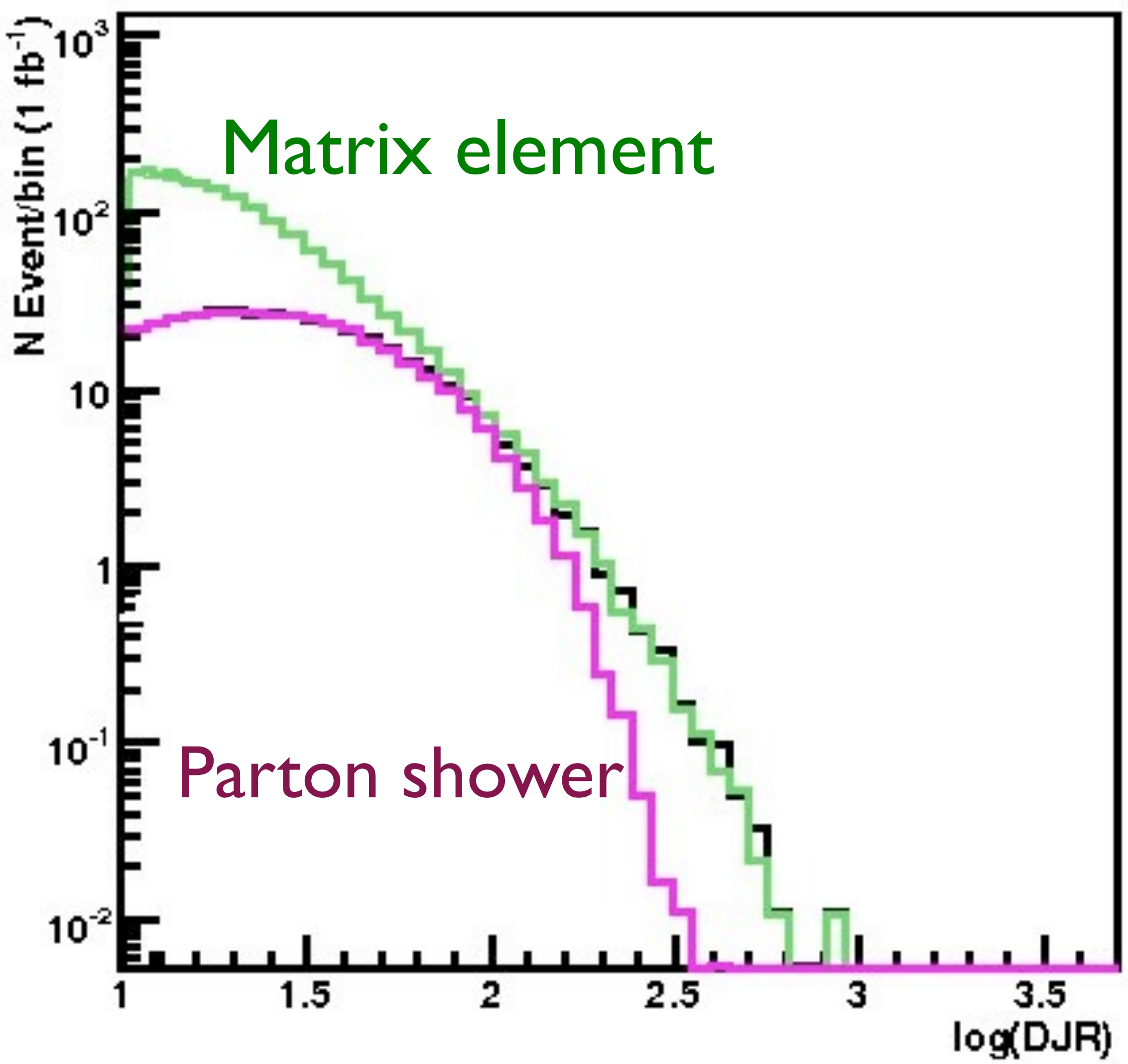
Highest multiplicity sample

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

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

Back to the “matching goal”

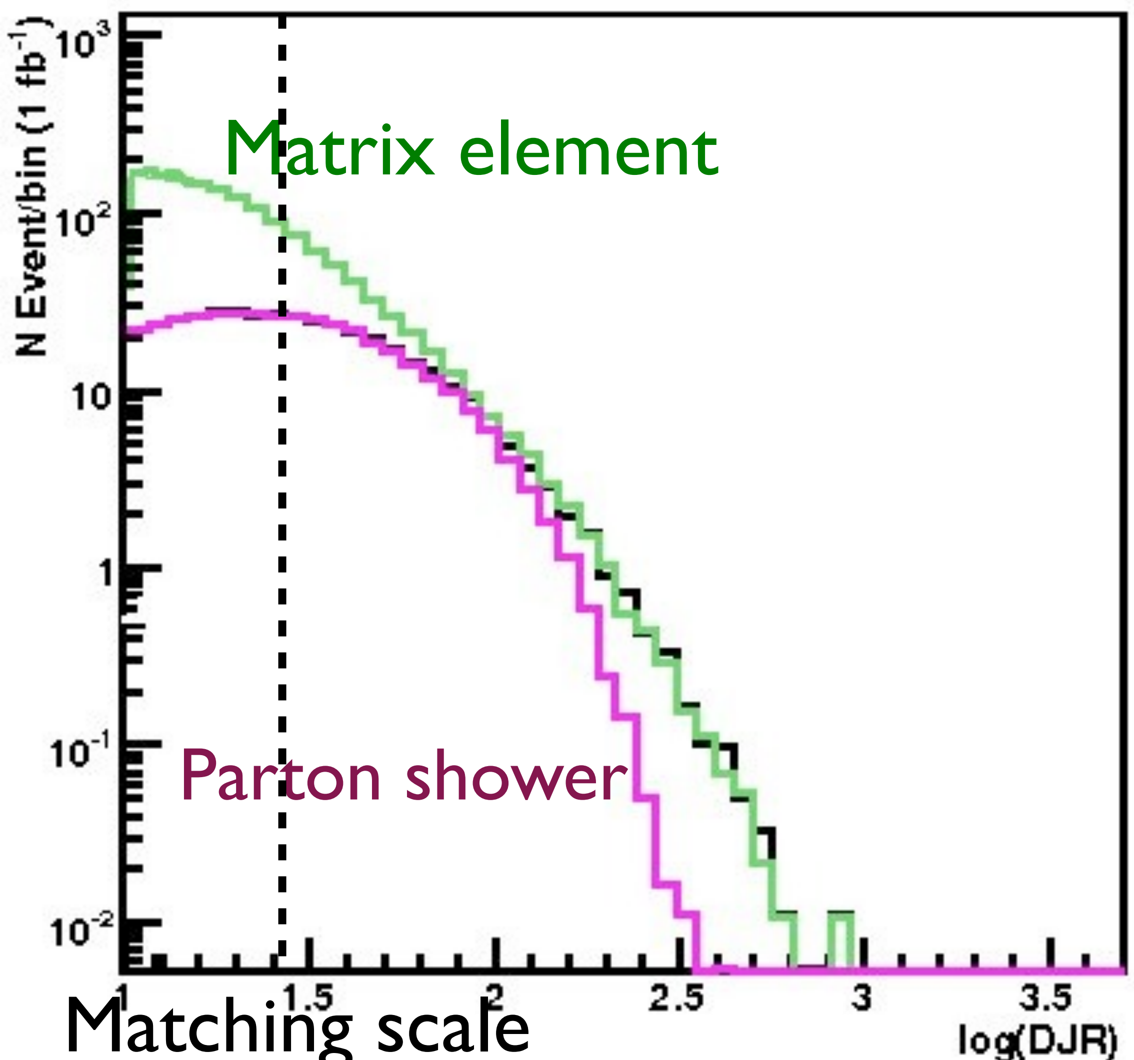
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



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

Back to the "matching goal"

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

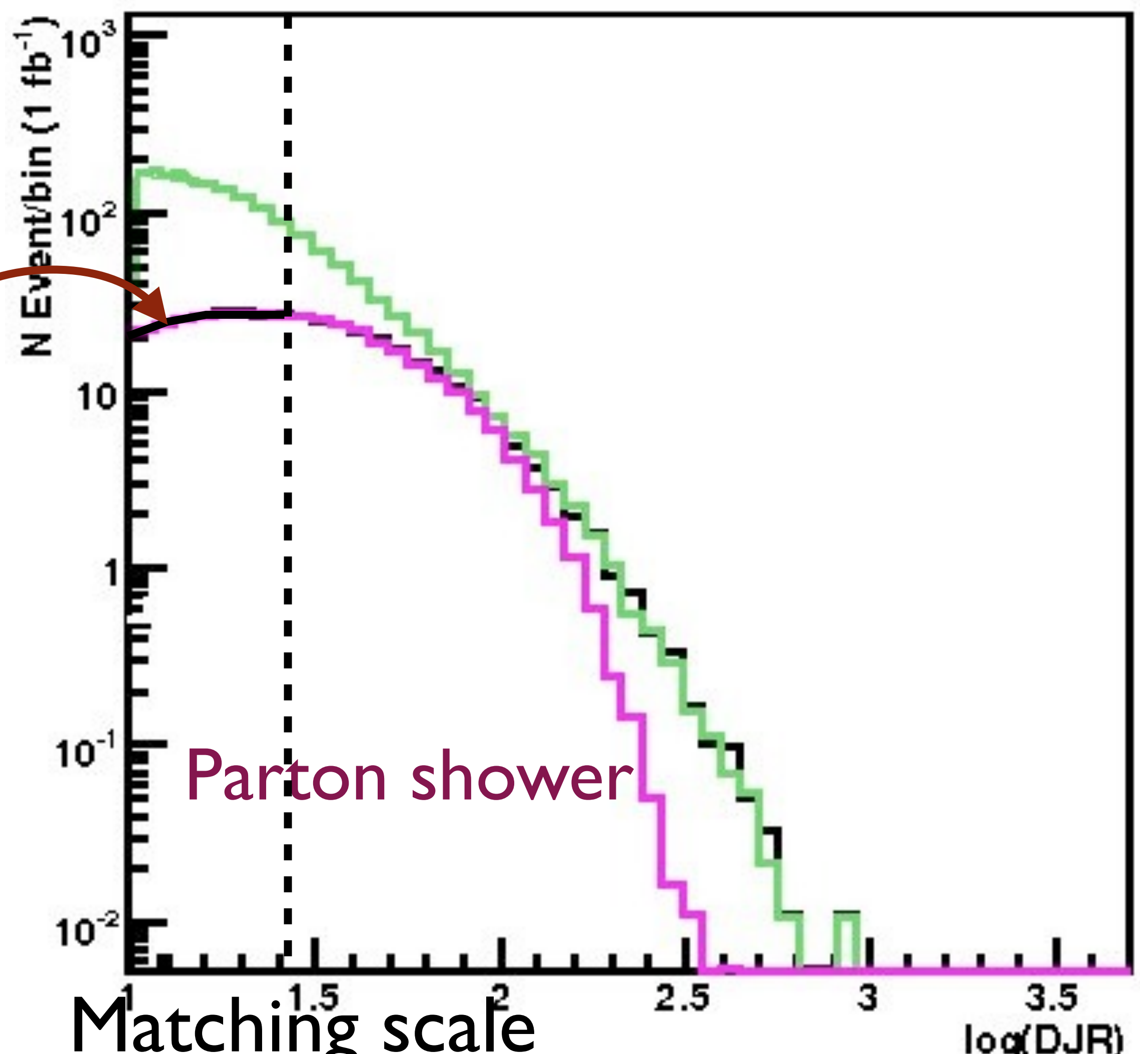


2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Back to the "matching goal"

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

Emissions from PS

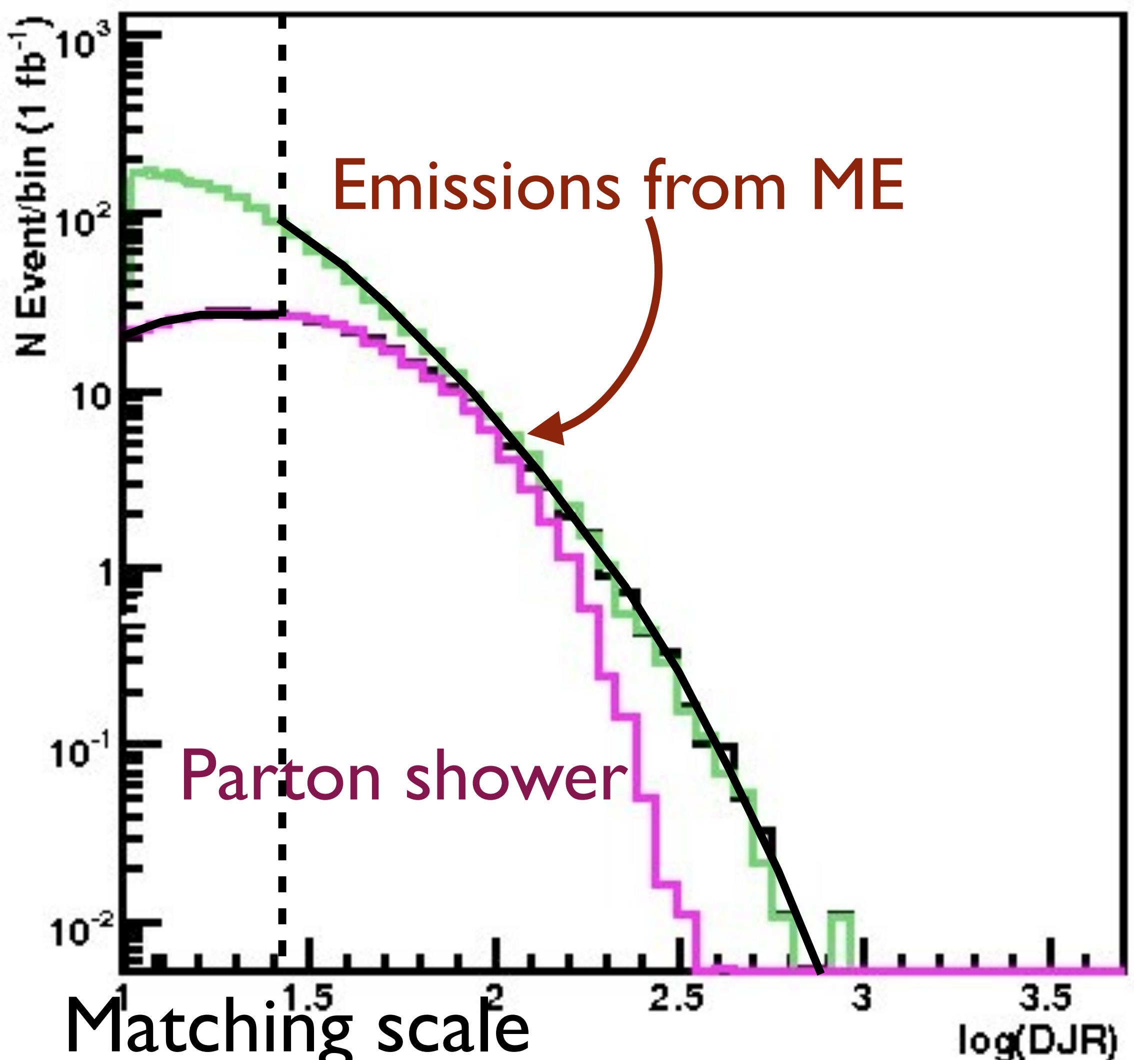


Matching scale

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

Back to the "matching goal"

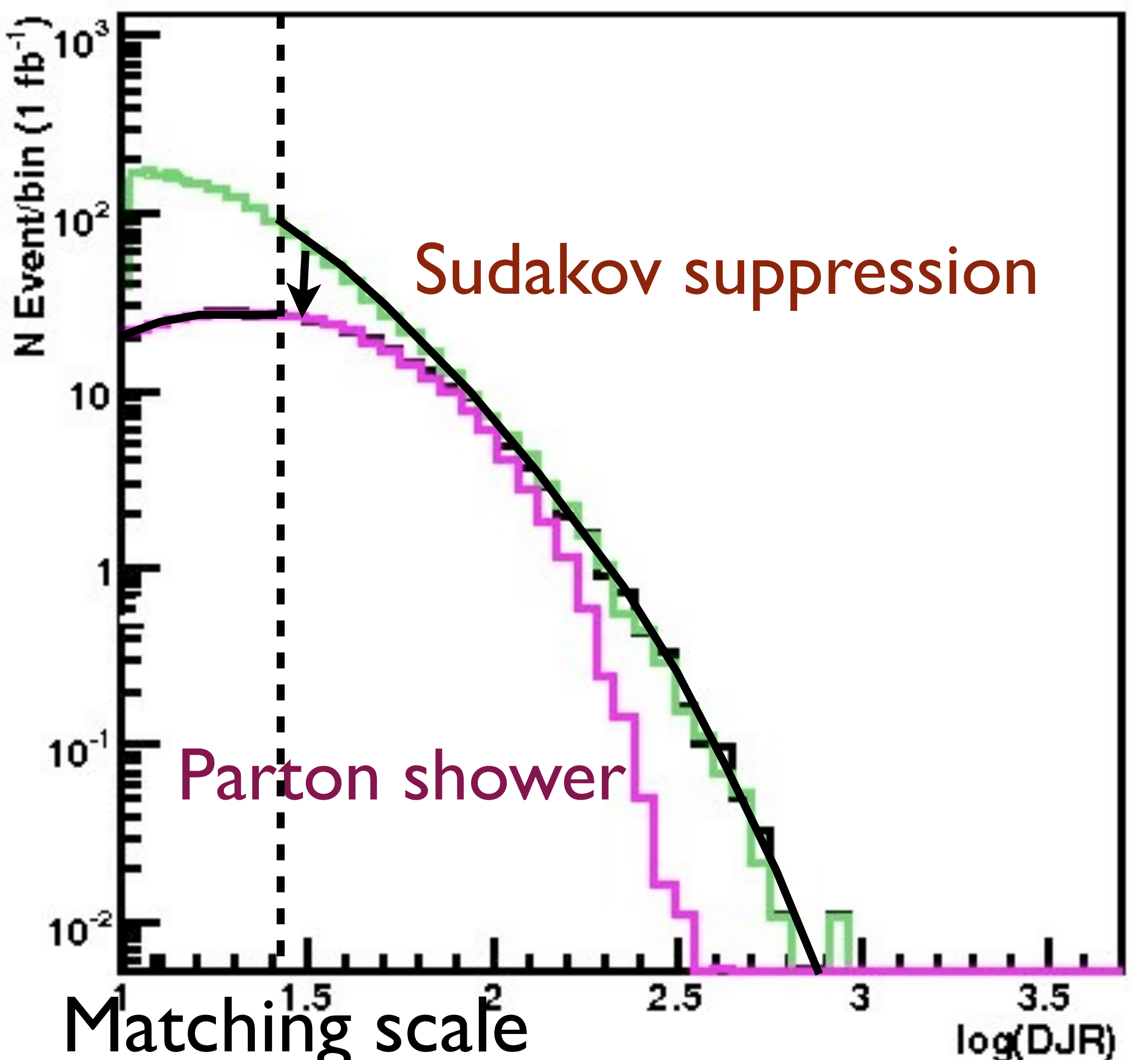
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



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

Back to the "matching goal"

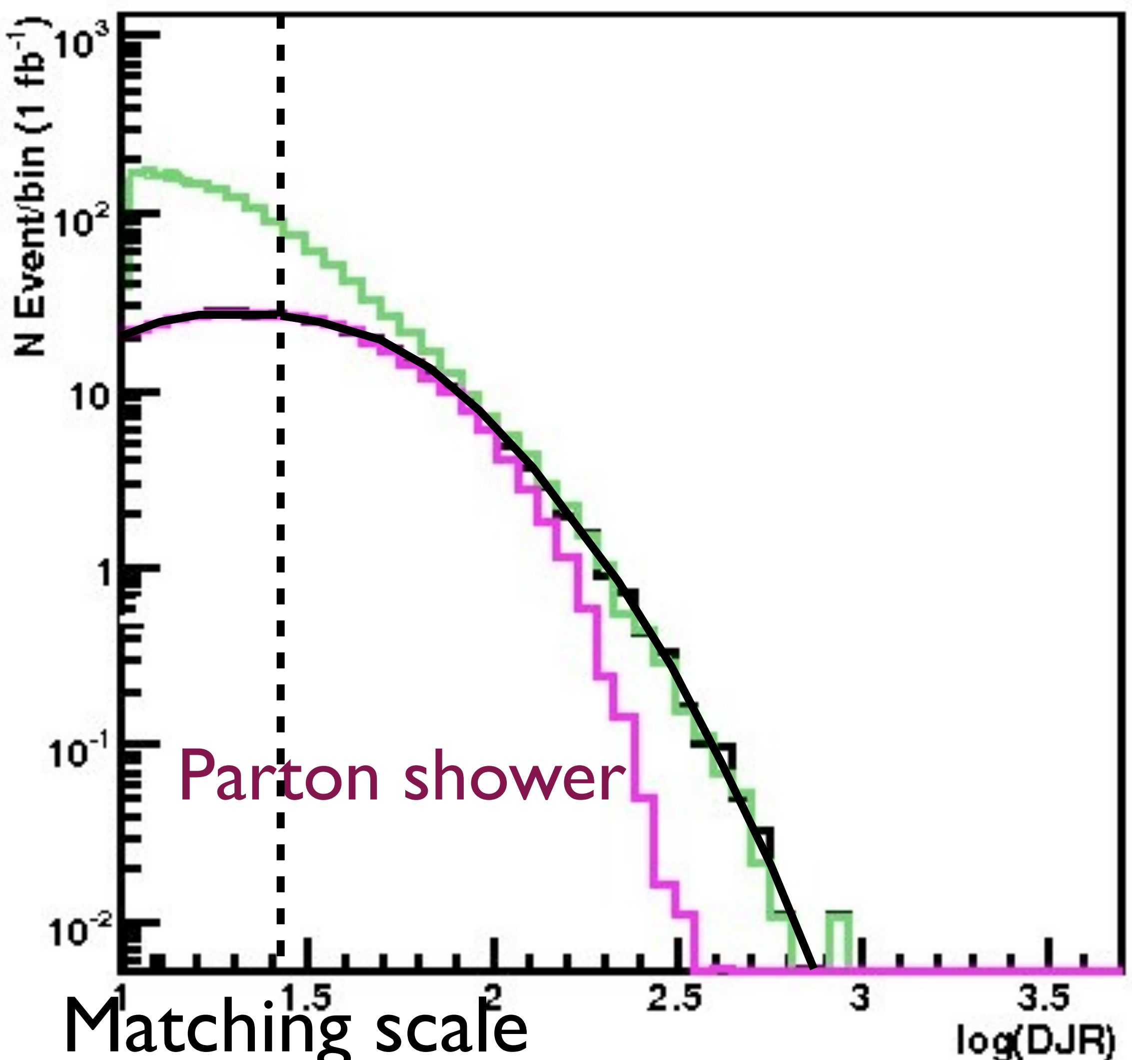
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



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

Back to the "matching goal"

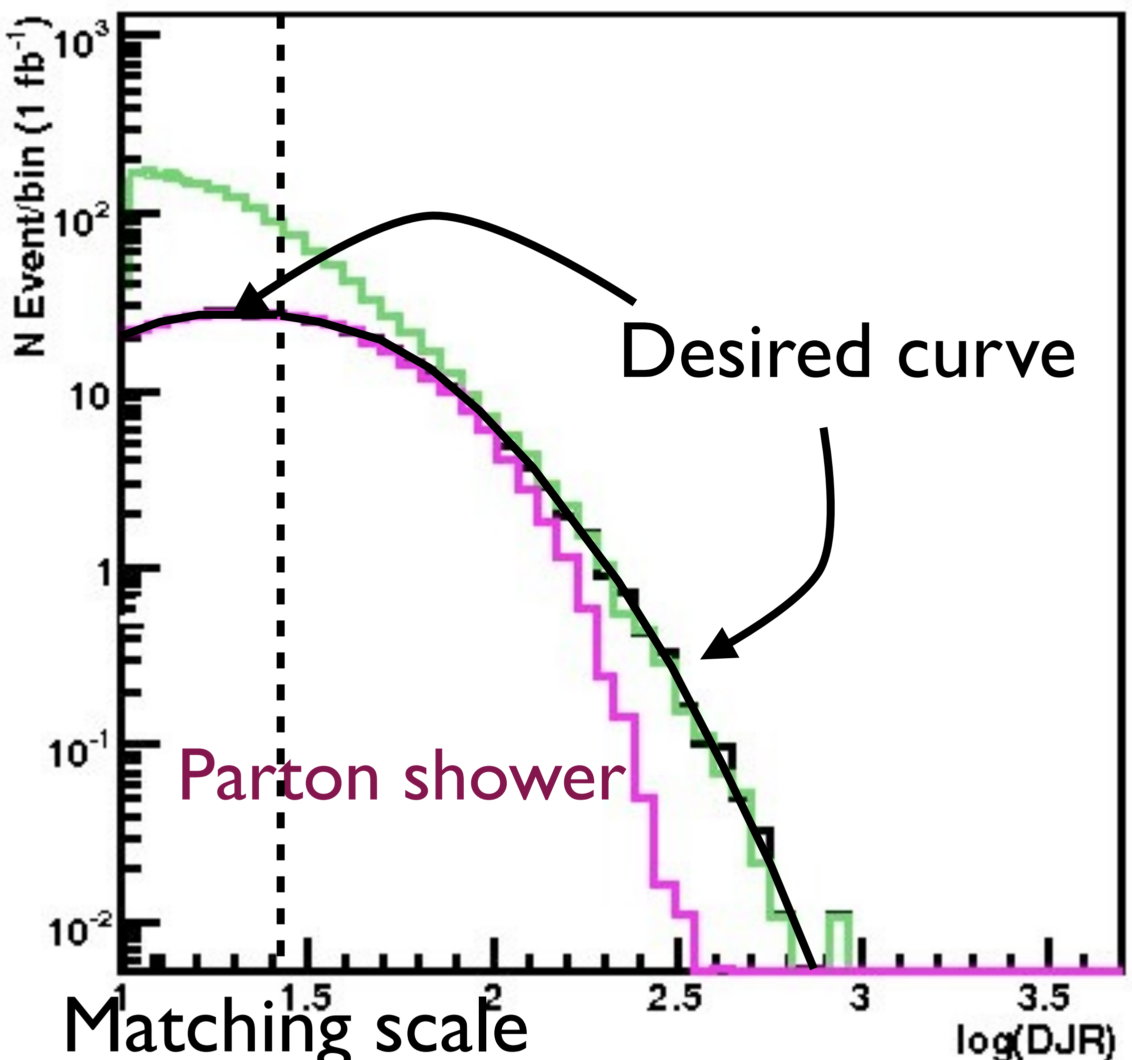
- Regularization of matrix element divergence
- Correction of the parton shower for large momenta
- Smooth jet distributions



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

Back to the “matching goal”

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

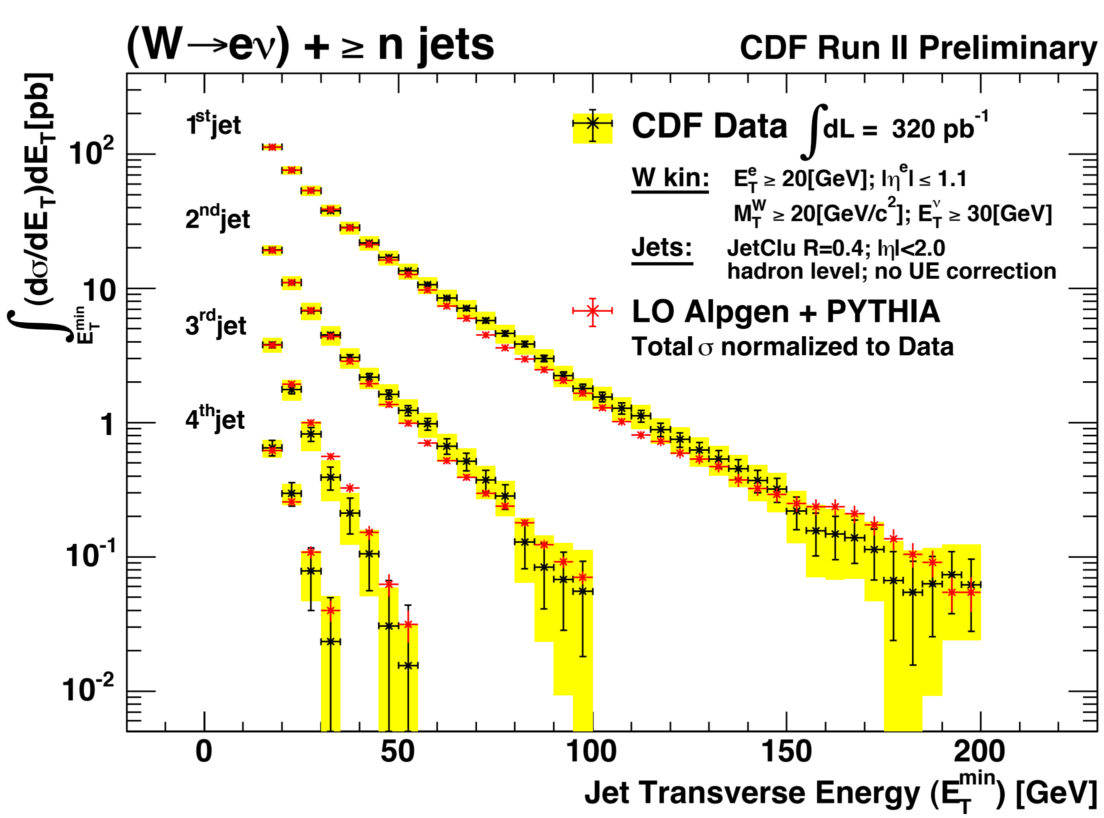


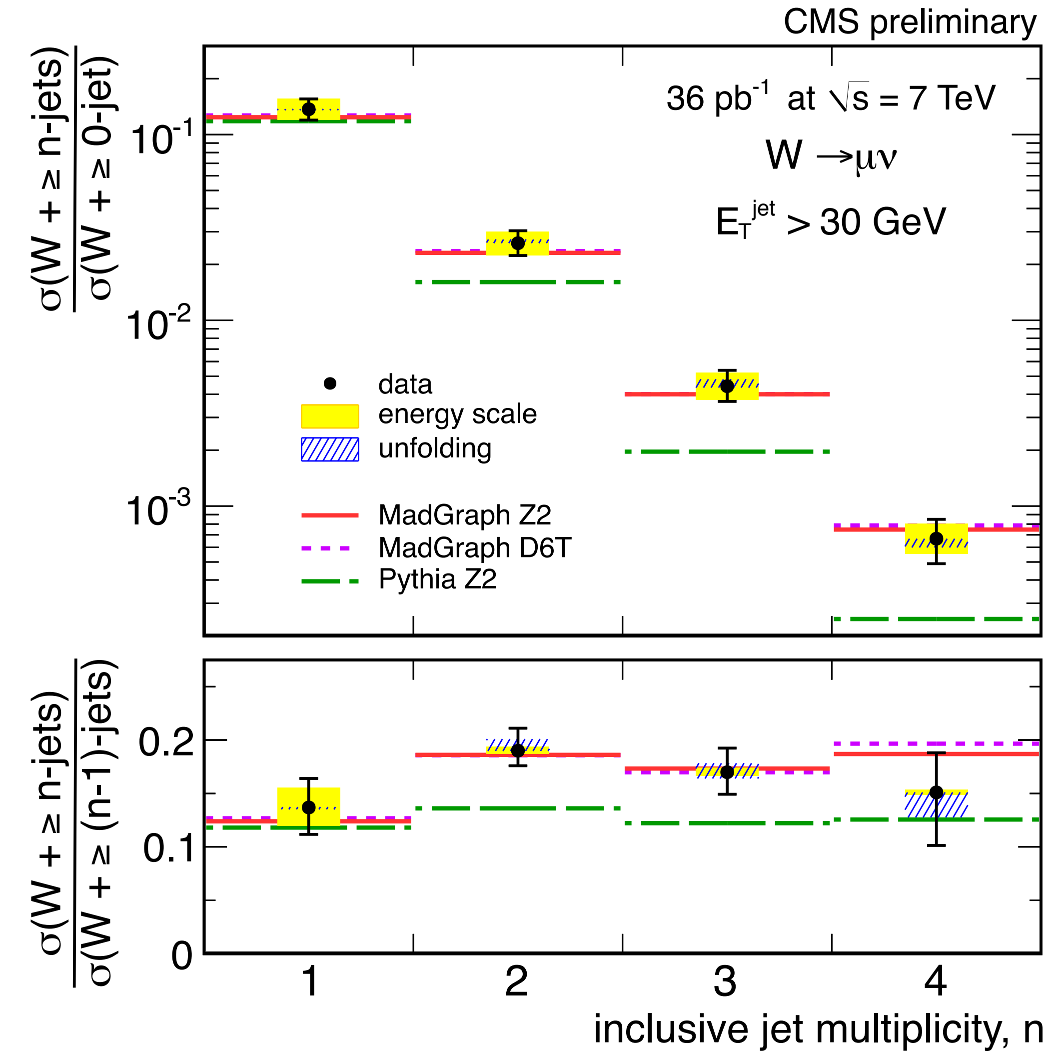
2nd QCD radiation jet in top pair production at the LHC, using MadGraph + Pythia

Summary of Matching Procedure

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

Comparing to experiment: W+jets





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

How to do matching in MadGraph+Pythia

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

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

```
In run_card.dat:
…
  1 = ickkw        <- Matching on
…
  0 = ptj          <- No cone matching
…
 15 = xqcut        <- kT matching scale
```

Matching on

No cone matching

k_T matching scale

Matching automatically done when run through MadEvent and Pythia!

How to do matching in MadGraph+Pythia

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

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

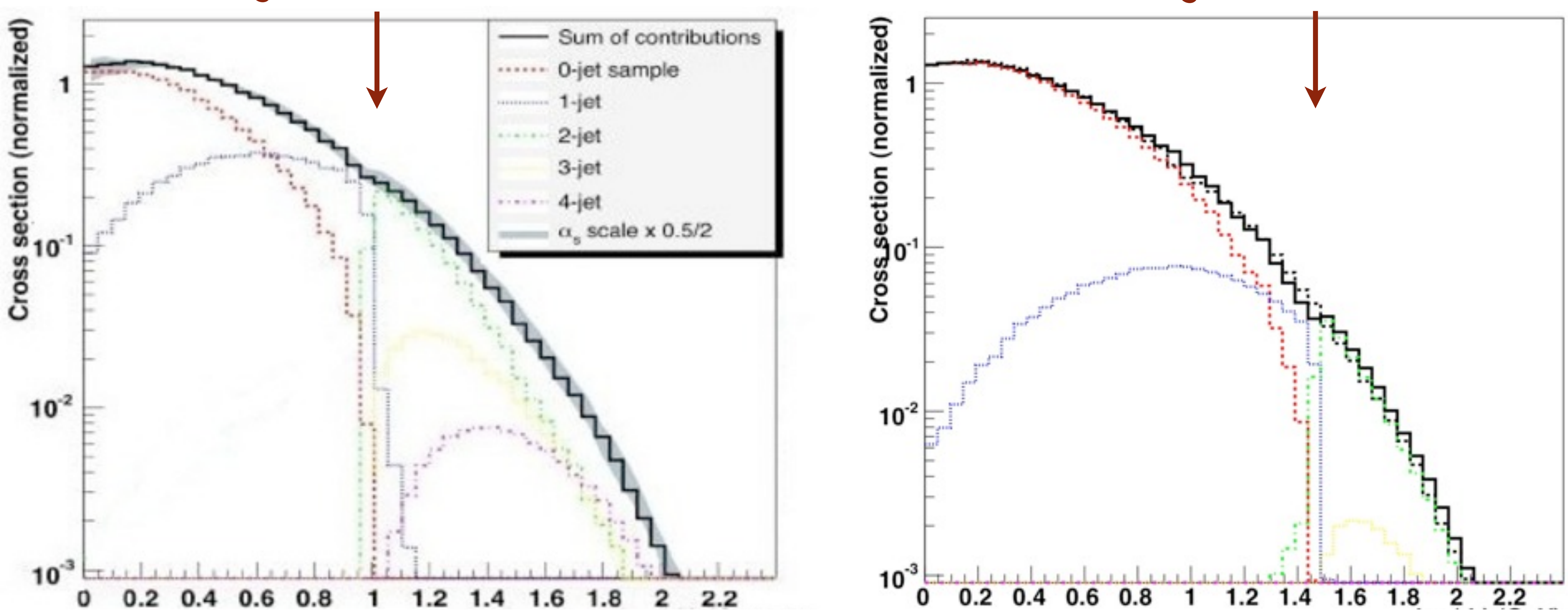
How to do validate the matching

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

Matching validation

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

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



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

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

Summary

Summary

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

Summary

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

Summary

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

Summary

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

Backup slides

MLM matching schemes in MadGraph

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

MLM matching schemes in MadGraph

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

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

MLM matching schemes in MadGraph

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

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

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

MLM matching schemes in MadGraph

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

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

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

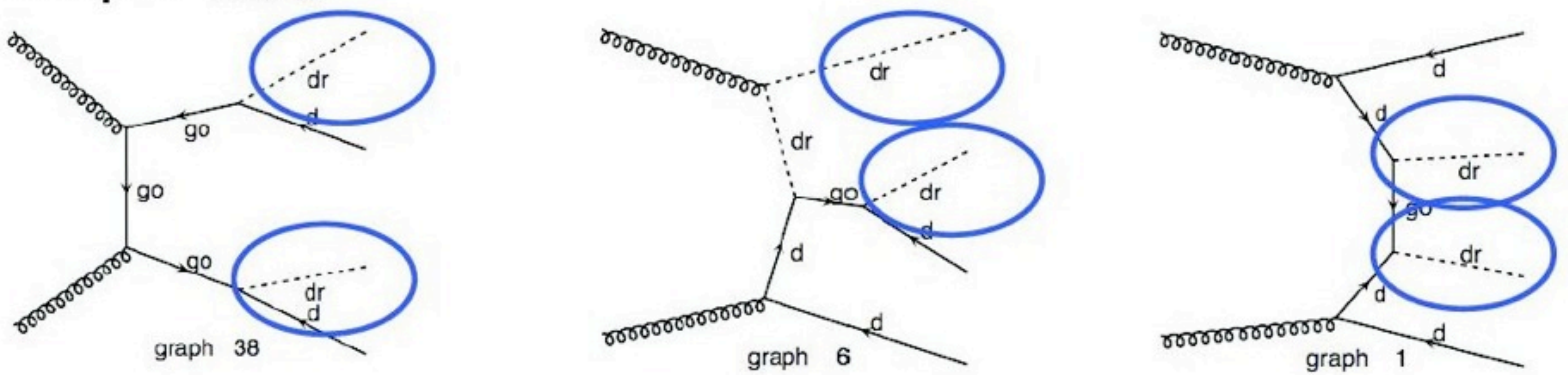
MLM matching schemes in MadGraph

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

Double counting of decays

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

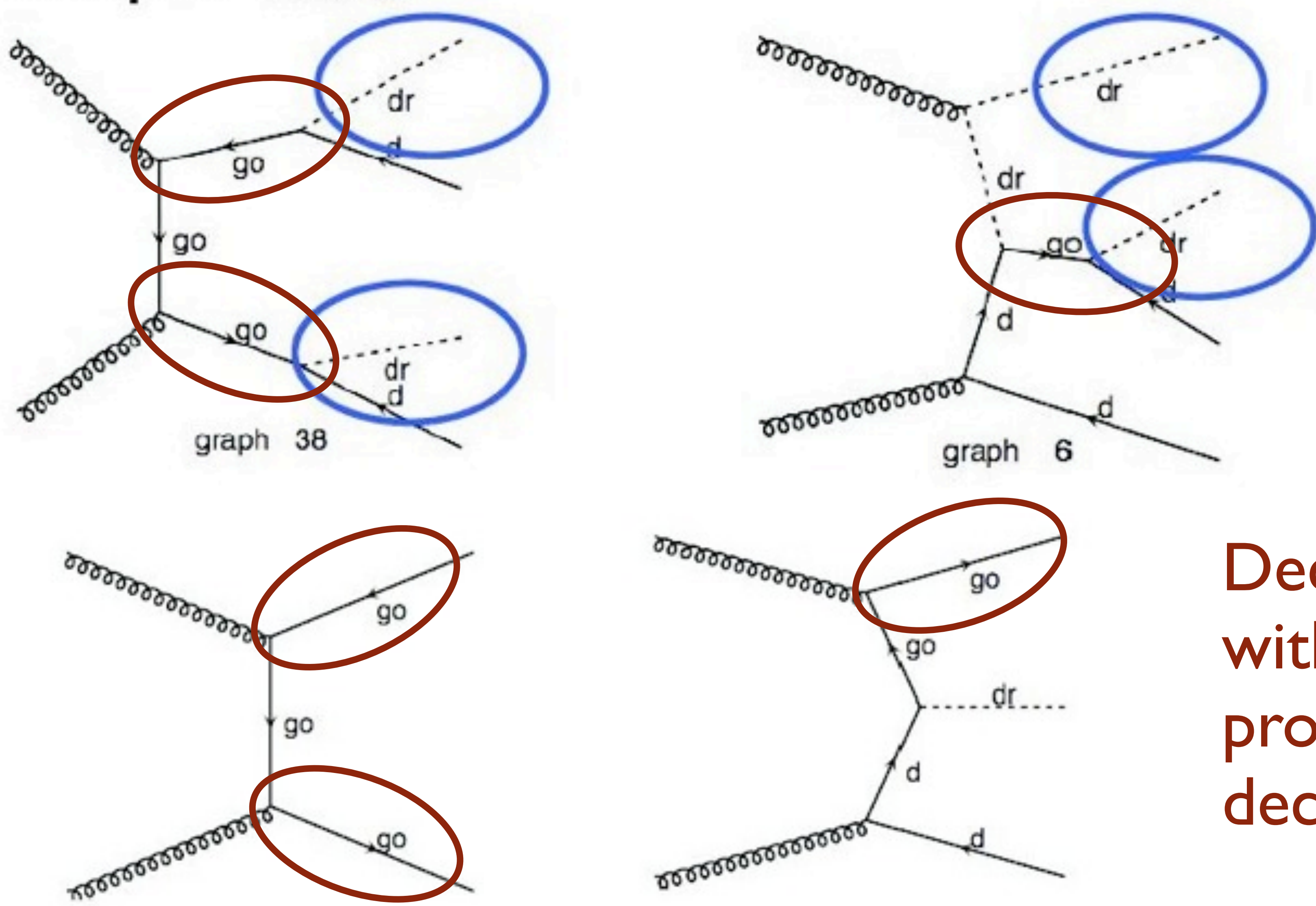
Example: $\tilde{q}\tilde{q}jj$



Double counting of decays

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

Example: $\tilde{q}\tilde{q}jj$



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

Double counting of decays

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

```
mg5> import model_v4 mssm
mg5> generate p p > dr dr~ j j $ go
```

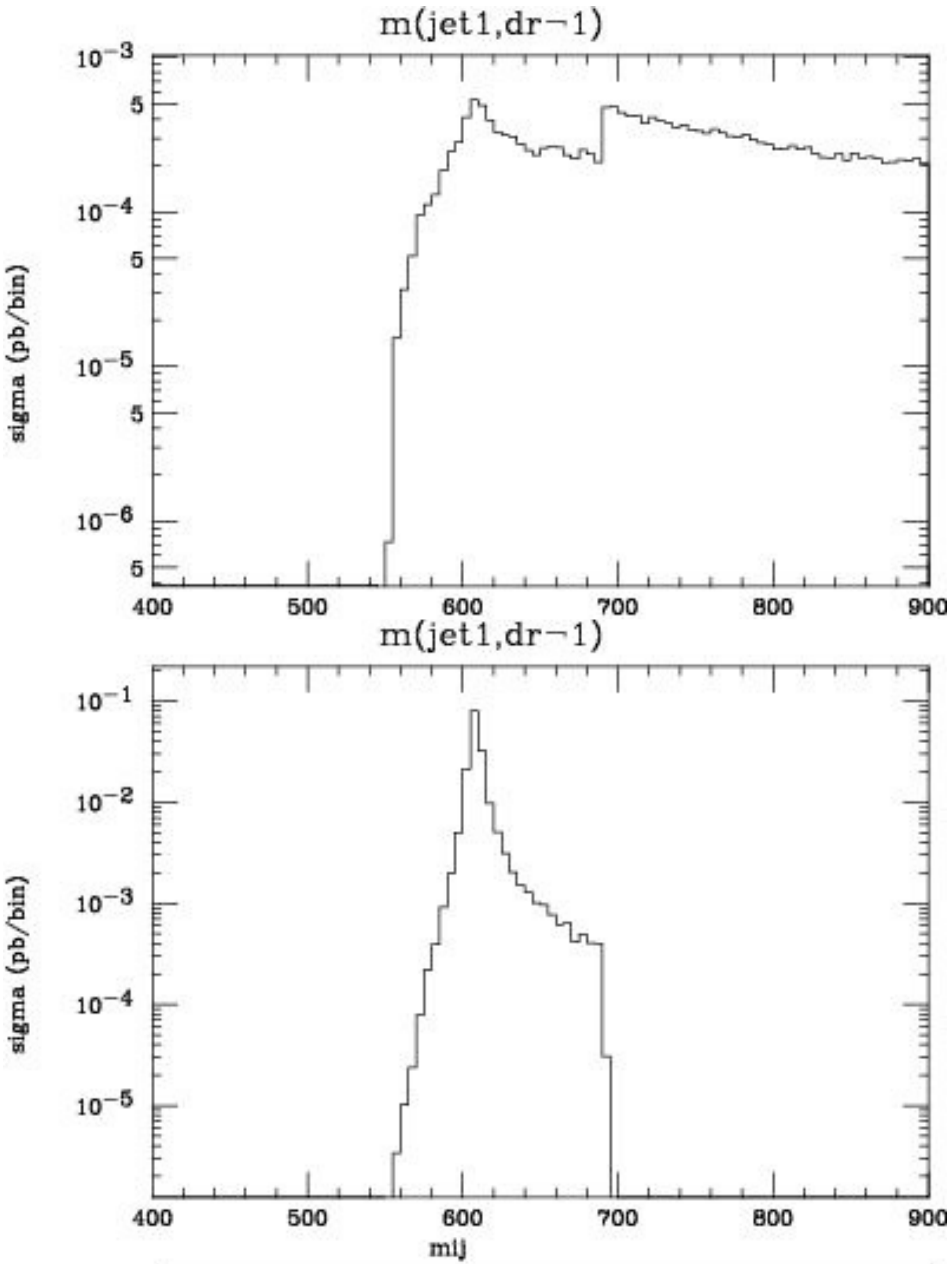
- This removes any on-shell gluinos from the event generation (where on-shell is defined as $m \pm n \cdot \Gamma$ with n set by `bwcutoff` in the run_card.dat)
- The corresponding region is exactly filled if you run gluino production with gluinos decaying to dr j (using the same `bwcutoff`).

Double counting of decays

Invariant mass distributions of d_r squark and d quark

p p > dr dr~ d $ go

p p > dr go, go > dr~ d



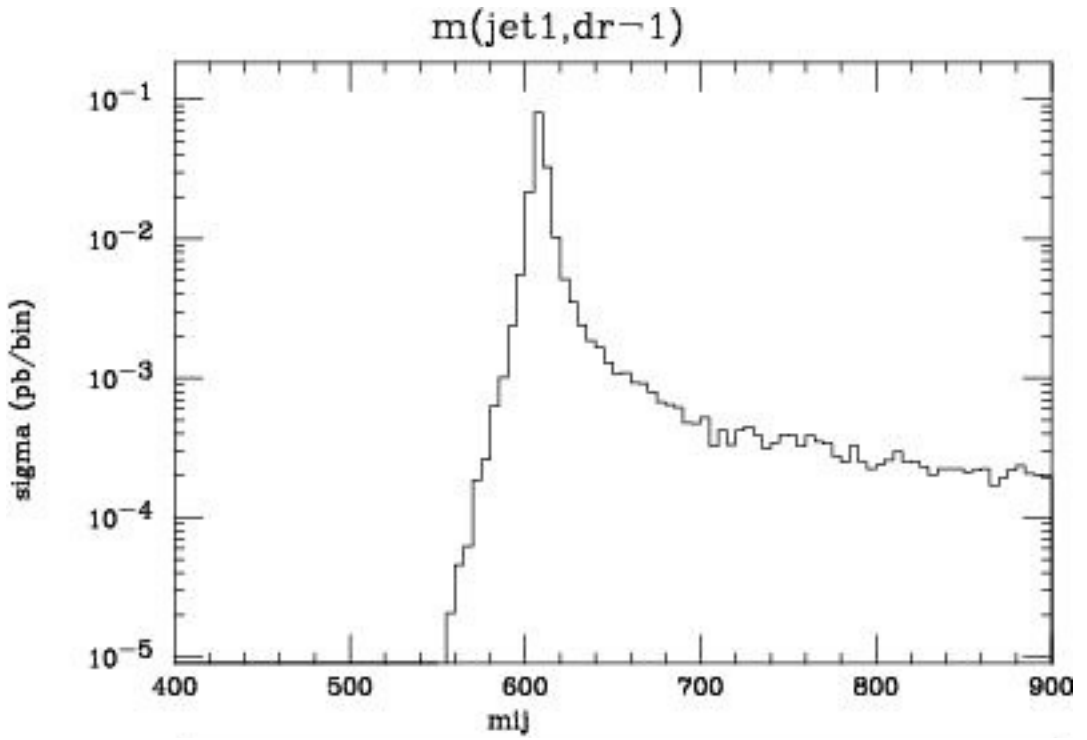
Double counting of decays

Invariant mass distributions
of d_r squark and d quark

p p > dr dr~ d $ go

+

p p > dr go, go > dr~ d



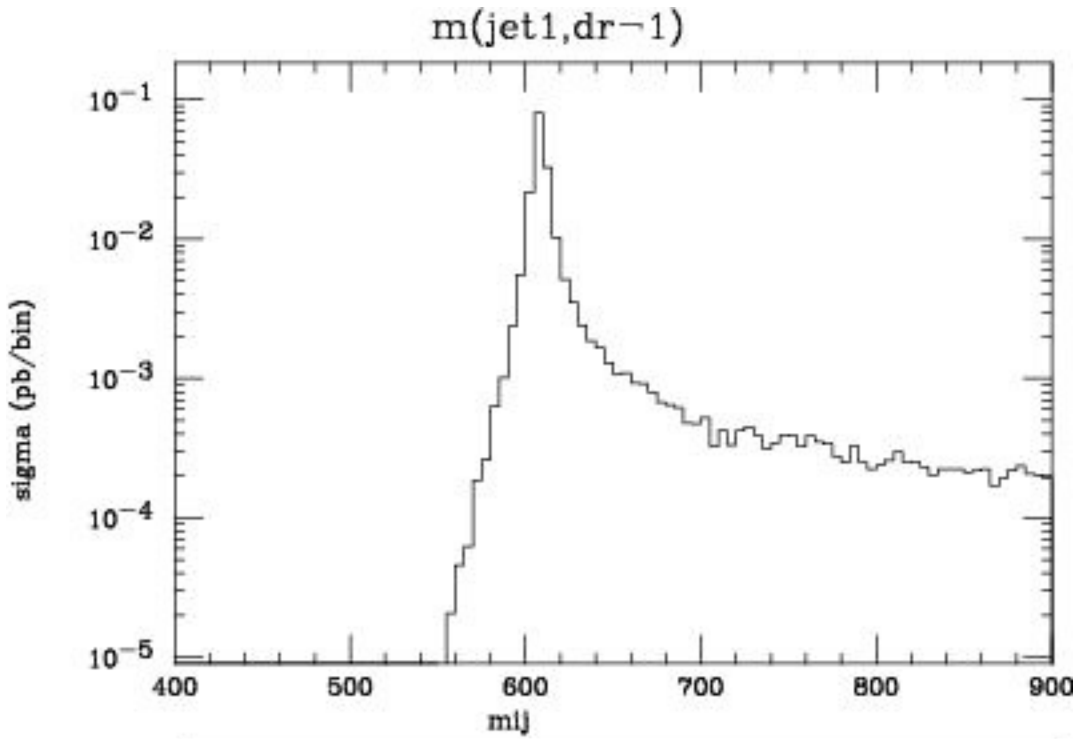
Double counting of decays

Invariant mass distributions of d_r squark and d quark

p p > dr dr~ d $ go

+

p p > dr go, go > dr~ d



Double counting between samples completely removed!