Numerical Methods in QCD

Matteo Cacciari

LPTHE, Université Pierre et Marie Curie (Paris 6)

Collège de France, 12 avril 2005

Outline

- Introduction
- Calculations
- Integrations
- Simulations
- Conclusions

Outline

- Introduction
- Calculations
- Integrations
- Simulations
- Conclusions

Outline

- Introduction
- Calculations
- Integrations
- Simulations
- Conclusions

Outline

- Introduction
- Calculations
- Integrations
- Simulations
- Conclusions

Outline

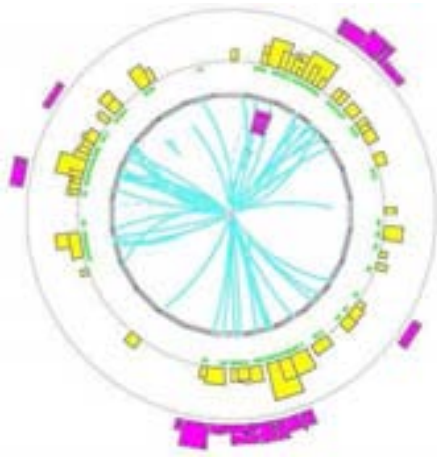
- Introduction
- Calculations
- Integrations
- Simulations
- Conclusions

Theory vs. Experiment

Theory

$$\mathcal{L}_{QCD} = -\frac{1}{4} F^a_{\mu\nu} F^{a,\mu\nu} + \dots$$

Experiment



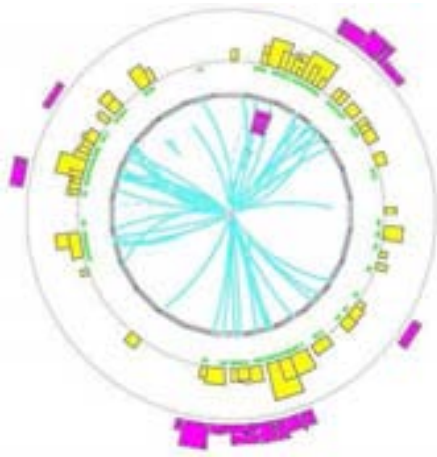
Obviously, a lot of work in between....

Theory vs. Experiment

Theory

$$\mathcal{L}_{QCD} = -\frac{1}{4} F^a_{\mu\nu} F^{a,\mu\nu} + \dots$$

Experiment



Obviously, a lot of work in between....

Lattice

In principle, given a lagrangian, and hence an *action* S, once can always calculate an expectation value of a given operator:

$$\langle \Phi(\phi) \rangle = \frac{1}{Z} \int \mathcal{D}\phi \, \Phi(\phi) \exp(-S(\phi))$$

In practice, of course, this is not easily doable, especially analitically.

One possible way out: discretize the space-time. This transforms the integral above into a finite sum:

$$\langle \Phi(\phi) \rangle \simeq \sum_{\text{field configurations } i} \Phi(\phi^{(i)}) w(\phi^{(i)})$$

Price to pay: large computing power needed.

Lattice

In principle, given a lagrangian, and hence an *action* S, once can always calculate an expectation value of a given operator:

$$\langle \Phi(\phi) \rangle = \frac{1}{Z} \int \mathcal{D}\phi \, \Phi(\phi) \exp(-S(\phi))$$

In practice, of course, this is not easily doable, especially analitically.

One possible way out: discretize the space-time. This transforms the integral above into a finite sum:

$$\langle \Phi(\phi) \rangle \simeq \sum_{\text{field configurations } i} \Phi(\phi^{(i)}) w(\phi^{(i)})$$

Price to pay: large computing power needed.

Lattice

In principle, given a lagrangian, and hence an *action* S, once can always calculate an expectation value of a given operator:

$$\langle \Phi(\phi) \rangle = \frac{1}{Z} \int \mathcal{D}\phi \, \Phi(\phi) \exp(-S(\phi))$$

In practice, of course, this is not easily doable, especially analitically.

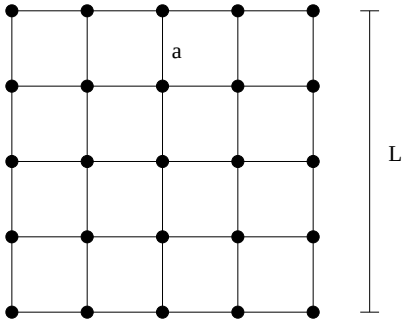
One possible way out: discretize the space-time. This transforms the integral above into a finite sum:

$$\langle \Phi(\phi) \rangle \simeq \sum_{\text{field configurations } i} \Phi(\phi^{(i)}) w(\phi^{(i)})$$

Price to pay: large computing power needed.

Lattice

Two distance scales are important:



a: grid size
L: total lattice size

Of course, L must be larger than the system we are trying to describe, while a must be small enough to 'see' its details.

Recalling that the size of a light hadron is $\sim 1/\Lambda_{QCD}$, we'd like to have

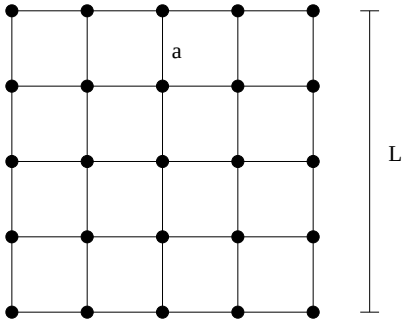
$$a \ll \frac{1}{m_Q} \ll \frac{1}{\Lambda_{QCD}} \ll \frac{1}{m_q} \ll L$$

In practice, the finite amount of computing power restricts us to

$$a \sim \frac{1}{m_Q} \ll \frac{1}{\Lambda_{QCD}} < \frac{1}{m_q} < L$$

Lattice

Two distance scales are important:



a: grid size
L: total lattice size

Of course, L must be larger than the system we are trying to describe, while a must be small enough to 'see' its details.

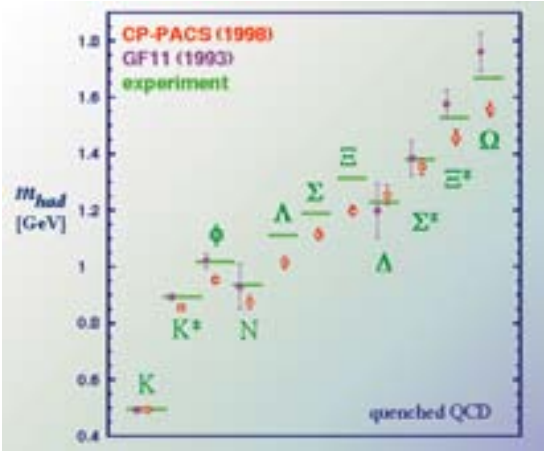
Recalling that the size of a light hadron is $\sim 1/\Lambda_{QCD}$, we'd like to have

$$a \ll \frac{1}{m_Q} \ll \frac{1}{\Lambda_{QCD}} \ll \frac{1}{m_q} \ll L$$

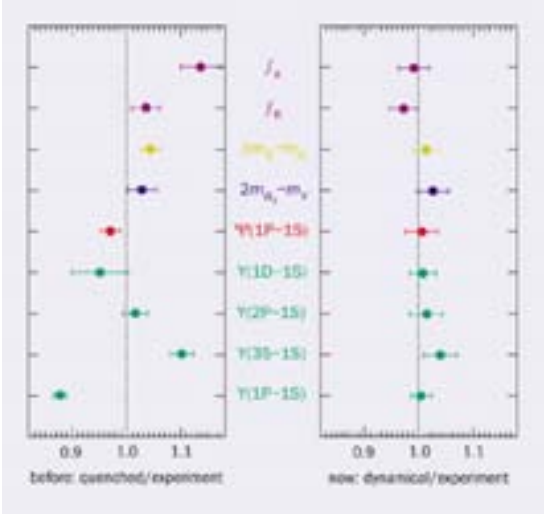
In practice, the finite amount of computing power restricts us to

$$a \sim \frac{1}{m_Q} \ll \frac{1}{\Lambda_{QCD}} < \frac{1}{m_q} < L$$

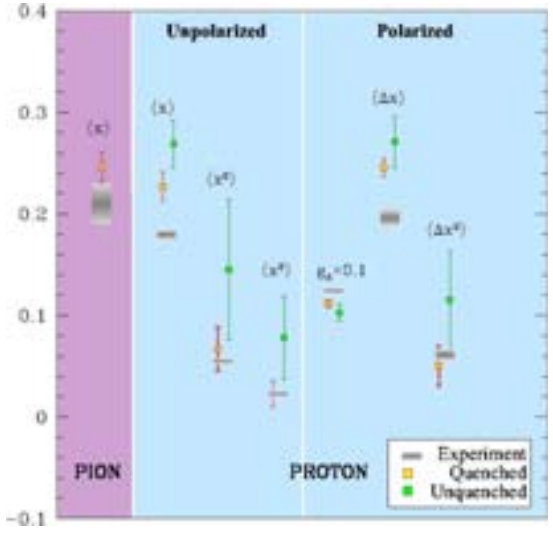
Selected Lattice Results



Light hadron masses



Couplings, masses, splittings

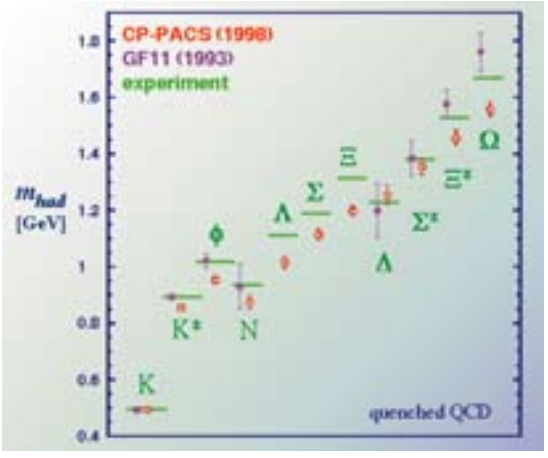


Moments of Structure Functions
(A. Shindler & K. Jansen)

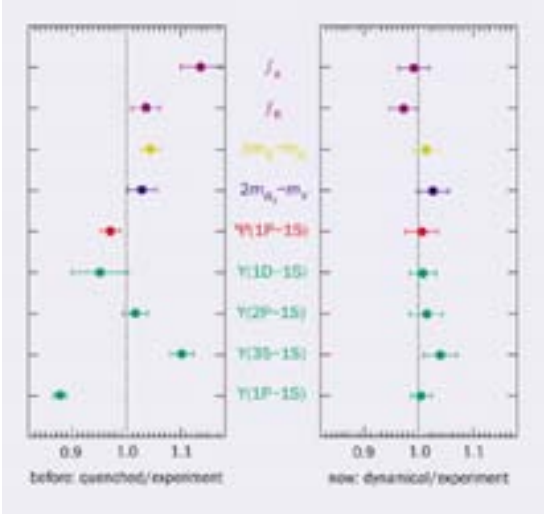
"Decent" results, but mainly for 'static' quantities.
Lattice still cannot do much dynamics.

For this, we move to perturbative QCD.

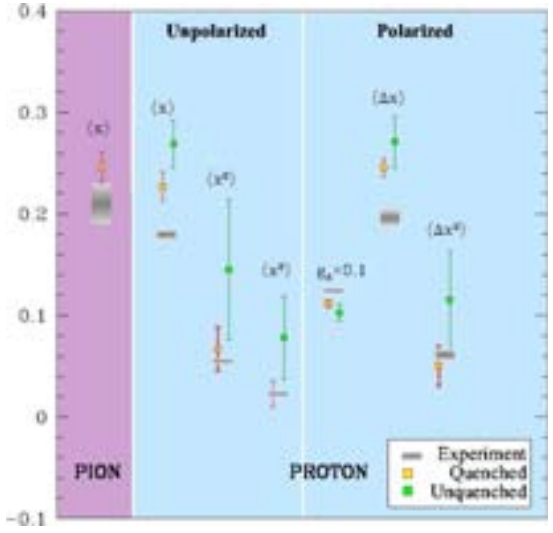
Selected Lattice Results



Light hadron masses



Couplings, masses, splittings

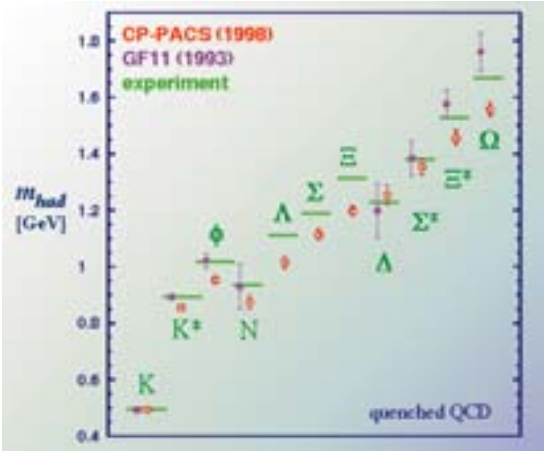


Moments of Structure Functions
(A. Shindler & K. Jansen)

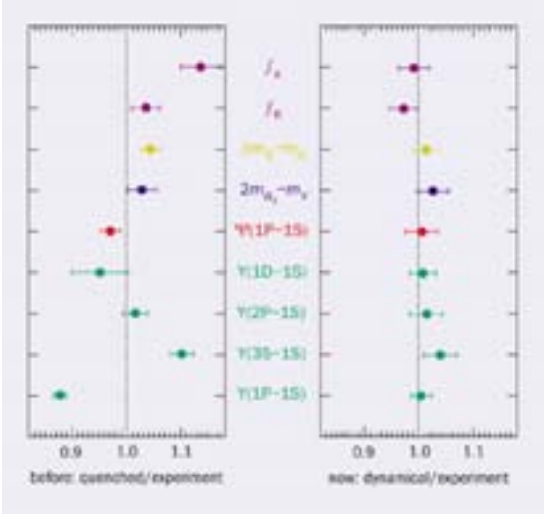
"Decent" results, but mainly for 'static' quantities.
Lattice still cannot do much dynamics.

For this, we move to perturbative QCD.

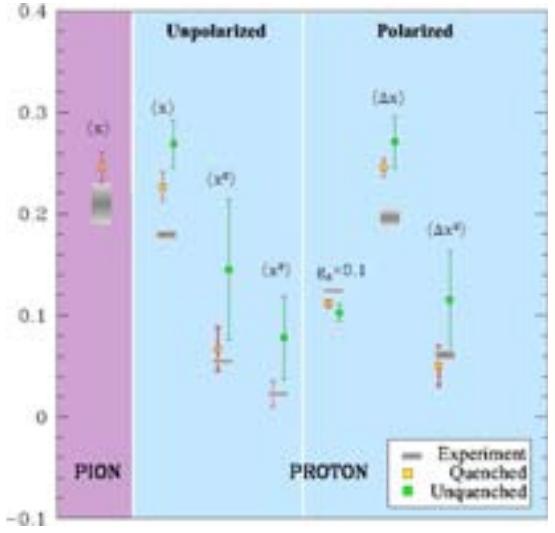
Selected Lattice Results



Light hadron masses



Couplings, masses, splittings



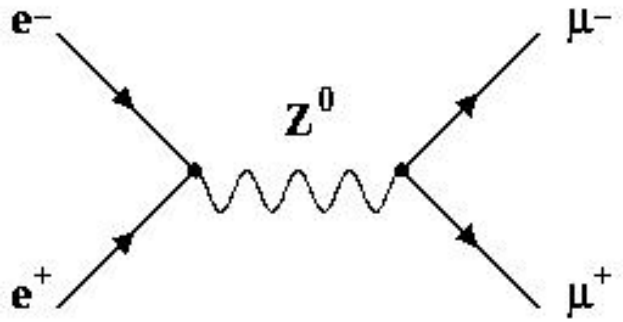
Moments of Structure Functions
(A. Shindler & K. Jansen)

"Decent" results, but mainly for 'static' quantities.
Lattice still cannot do much dynamics.

For this, we move to perturbative QCD.

Feynman Diagrams

The main (or at least oldest) tool of perturbative QCD are the Feynman diagrams, e.g.



Procedure fully algorithmic.
In principle any scattering process can be calculated:

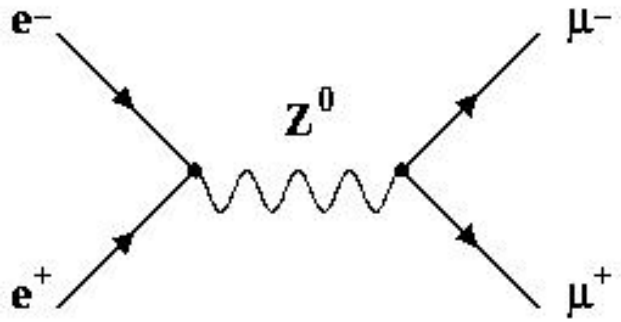
$$d\sigma(ab \to n) = \frac{1}{\text{flux}} \overline{|\mathcal{M}(ab \to n)|^2} d\text{Lips}_n$$

$$|\mathcal{M}(ab \to n)|^2 \sim \sum_{\text{polarizations}} \sum_{\text{colours}} \sum_{\text{spins}} \sum_{\cdots} \mathcal{A}\mathcal{A}^\dagger$$

In practice, the calculation becomes quickly very complicated with increasing number of particles in the final state, as both the number of diagrams and the complexity of the phase space integration grow dramatically.

Feynman Diagrams

The main (or at least oldest) tool of perturbative QCD are the Feynman diagrams, e.g.



Procedure fully algorithmic.
In principle any scattering process can be calculated:

$$d\sigma(ab \to n) = \frac{1}{\text{flux}} \overline{|\mathcal{M}(ab \to n)|^2} d\text{Lips}_n$$

$$|\mathcal{M}(ab \to n)|^2 \sim \sum_{\text{polarizations}} \sum_{\text{colours}} \sum_{\text{spins}} \sum_{\cdots} \mathcal{A}\mathcal{A}^\dagger$$

In practice, the calculation becomes quickly very complicated with increasing number of particles in the final state, as both the number of diagrams and the complexity of the phase space integration grow dramatically.

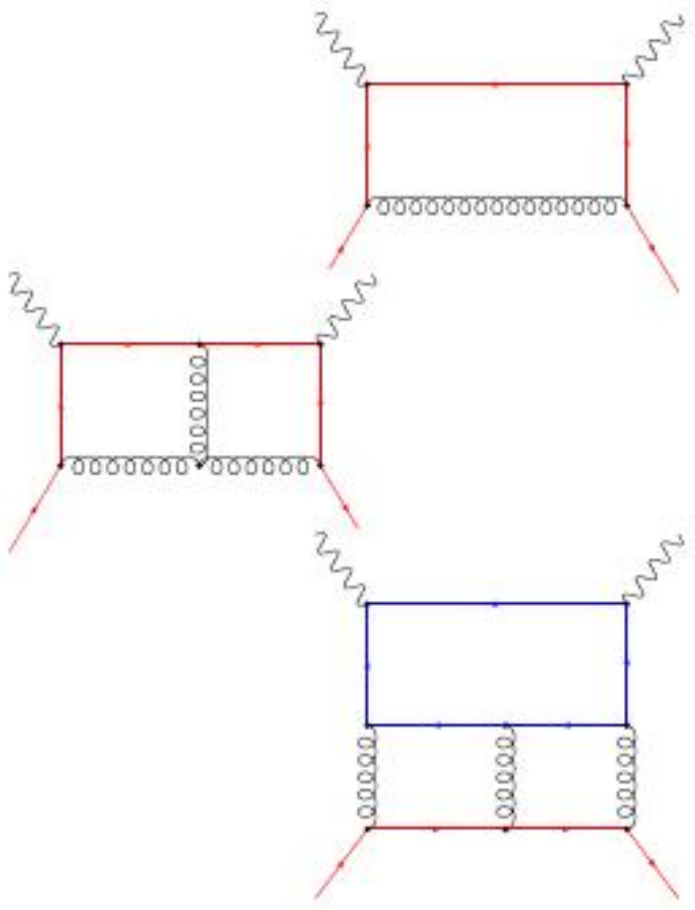
AP Splitting Functions

Example:
increasing complexity of Altarelli-Parisi splitting functions calculation.

- **One-loop** Feynman diagrams
  $\longrightarrow$ in total 18 for $\gamma_{ij}^{(0)}$ / $P_{ij}^{(0)}$
  (pencil + paper)

- **Two-loop** Feynman diagrams
  $\longrightarrow$ in total 350 for $\gamma_{ij}^{(1)}$ / $P_{ij}^{(1)}$
  (simple computer algebra)

- **Three-loop** Feynman diagrams
  $\longrightarrow$ in total 9607 for $\gamma_{ij}^{(2)}$ / $P_{ij}^{(2)}$
  (cutting edge technology $\longrightarrow$ computer algebra system FORM Vermaseren '89-'04)



1 loop

$$P_{qq}^{(0)}(x) = C_F \left(\frac{1+x^2}{1-x} \right)_+$$

$$P_{gq}^{(0)}(x) = C_F \frac{1+(1-x)^2}{x}$$

$$P_{qg}^{(0)}(x) = T_R \left[x^2 + (1-x)^2 \right]$$

$$P_{gg}^{(0)}(x) = 2C_A \left[\frac{x}{(1-x)_+} + \frac{1-x}{x} + x(1-x) \right] + \frac{1}{6}(11C_A - 4n_f T_R)\delta(1-x)$$

Altarelli, Parisi, 1977

2 loops

$$P_{ps}^{(1)}(x) = 4C_F n_f\left(\frac{20}{9}\frac{1}{x} - 2 + 6x - 4\mathrm{H}_0 + x^2\left[\frac{8}{3}\mathrm{H}_0 - \frac{56}{9}\right] + (1+x)\left[5\mathrm{H}_0 - 2\mathrm{H}_{0,0}\right]\right)$$

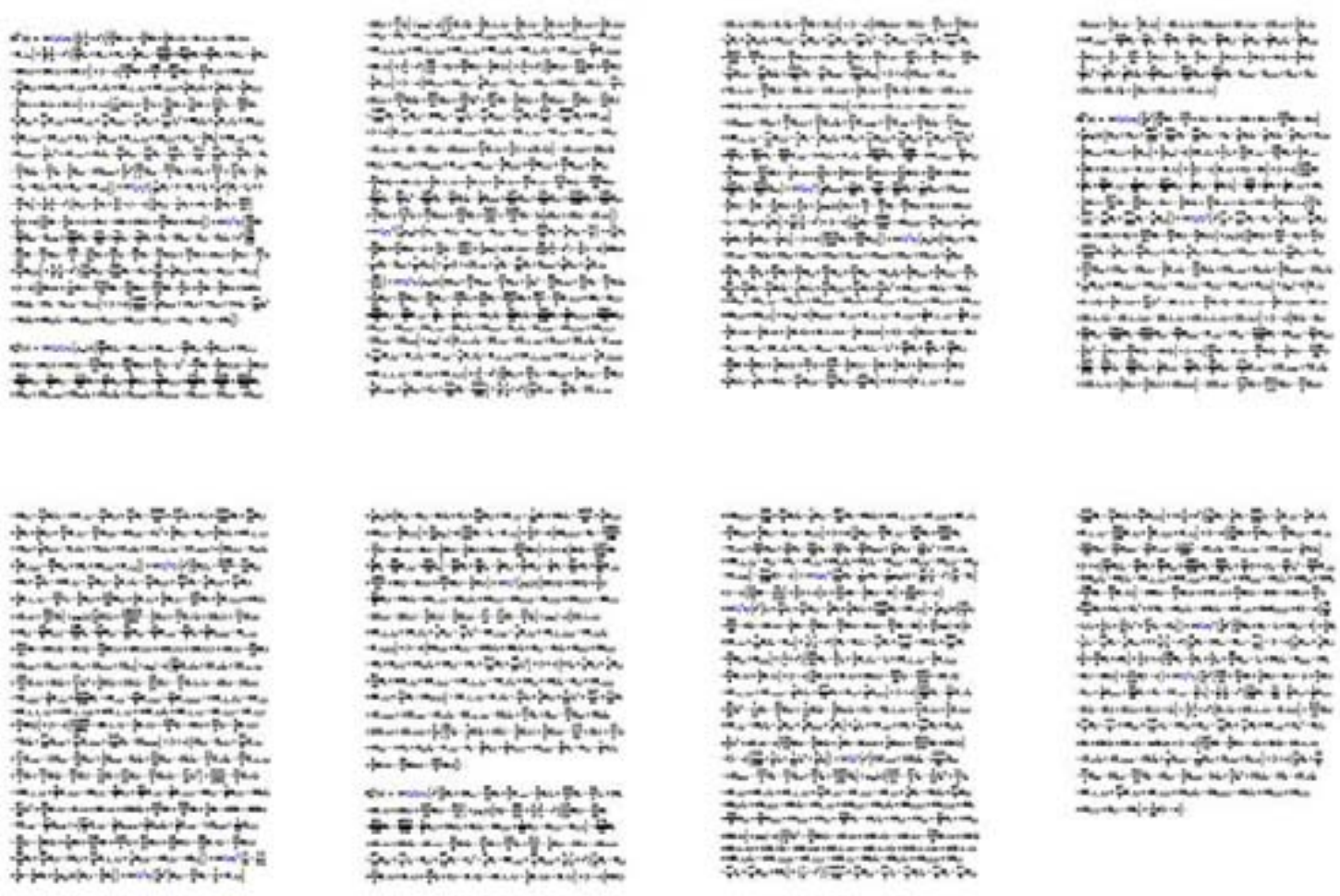
$$P_{qg}^{(1)}(x) = 4C_A n_f\left(\frac{20}{9}\frac{1}{x} - 2 + 25x - 2p_{qg}(-x)\mathrm{H}_{-1,0} - 2p_{qg}(x)\mathrm{H}_{1,1} + x^2\left[\frac{44}{3}\mathrm{H}_0 - \frac{218}{9}\right]\right.$$
$$\left. + 4(1-x)\left[\mathrm{H}_{0,0} - 2\mathrm{H}_0 + x\mathrm{H}_1\right] - 4\zeta_2 x - 6\mathrm{H}_{0,0} + 9\mathrm{H}_0\right) + 4C_F n_f\left(2p_{qg}(x)\left[\mathrm{H}_{1,0} + \mathrm{H}_{1,1} + \mathrm{H}_2\right.\right.$$
$$\left.\left. - \zeta_2\right] + 4x^2\left[\mathrm{H}_0 + \mathrm{H}_{0,0} + \frac{5}{2}\right] + 2(1-x)\left[\mathrm{H}_0 + \mathrm{H}_{0,0} - 2x\mathrm{H}_1 + \frac{29}{4}\right] - \frac{15}{2} - \mathrm{H}_{0,0} - \frac{1}{2}\mathrm{H}_0\right)$$

$$P_{gq}^{(1)}(x) = 4C_A C_F\left(\frac{1}{x} + 2p_{gq}(x)\left[\mathrm{H}_{1,0} + \mathrm{H}_{1,1} + \mathrm{H}_2 - \frac{11}{6}\mathrm{H}_1\right] - x^2\left[\frac{8}{3}\mathrm{H}_0 - \frac{44}{9}\right] + 4\zeta_2 - 2\right.$$
$$\left. - 7\mathrm{H}_0 + 2\mathrm{H}_{0,0} - 2\mathrm{H}_1 x + (1+x)\left[2\mathrm{H}_{0,0} - 5\mathrm{H}_0 + \frac{37}{9}\right] - 2p_{gq}(-x)\mathrm{H}_{-1,0}\right) - 4C_F n_f\left(\frac{2}{3}x\right.$$
$$\left. - p_{gq}(x)\left[\frac{2}{3}\mathrm{H}_1 - \frac{10}{9}\right]\right) + 4{C_F}^2\left(p_{gq}(x)\left[3\mathrm{H}_1 - 2\mathrm{H}_{1,1}\right] + (1+x)\left[\mathrm{H}_{0,0} - \frac{7}{2} + \frac{7}{2}\mathrm{H}_0\right] - 3\mathrm{H}_{0,0}\right.$$
$$\left. + 1 - \frac{3}{2}\mathrm{H}_0 + 2\mathrm{H}_1 x\right)$$

$$P_{gg}^{(1)}(x) = 4C_A n_f\left(1 - x - \frac{10}{9}p_{gg}(x) - \frac{13}{9}\left(\frac{1}{x} - x^2\right) - \frac{2}{3}(1+x)\mathrm{H}_0 - \frac{2}{3}\delta(1-x)\right) + 4{C_A}^2\left(27\right.$$
$$\left. + (1+x)\left[\frac{11}{3}\mathrm{H}_0 + 8\mathrm{H}_{0,0} - \frac{27}{2}\right] + 2p_{gg}(-x)\left[\mathrm{H}_{0,0} - 2\mathrm{H}_{-1,0} - \zeta_2\right] - \frac{67}{9}\left(\frac{1}{x} - x^2\right) - 12\mathrm{H}_0\right.$$
$$\left. - \frac{44}{3}x^2\mathrm{H}_0 + 2p_{gg}(x)\left[\frac{67}{18} - \zeta_2 + \mathrm{H}_{0,0} + 2\mathrm{H}_{1,0} + 2\mathrm{H}_2\right] + \delta(1-x)\left[\frac{8}{3} + 3\zeta_3\right]\right) + 4C_F n_f\left(2\mathrm{H}_0\right.$$
$$\left. + \frac{2}{3}\frac{1}{x} + \frac{10}{3}x^2 - 12 + (1+x)\left[4 - 5\mathrm{H}_0 - 2\mathrm{H}_{0,0}\right] - \frac{1}{2}\delta(1-x)\right).$$

Curci, Furmanski, Petronzio, 1979

3 loops



Moch, Vermaseren, Vogt, 2004

Calculational tools

Of course, one cannot go on calculating traces and convolutions for thousands of diagrams by hand.

Three main classes of tools are at our disposal for multi-particle, tree level calculations:

- Tools for computer-aided analytical manipulations (Schoonschip M. Veltman, FORM [www.nikhef.nl/~form/], Mathematica/FeynCalc [www.feyncalc.org], ...)
- Tools for generating, calculating Feynman diagrams, and automatically integrating over phase space (CompHEP [theory.sinp.msu.ru/comphep], MadGraph/MadEvent [madgraph.hep.uiuc.edu], Sherpa, ...)
- Tools for calculating amplitudes numerically directly from the lagrangian (Alpha/ALPGEN [home.cern.ch/mlm/alpgen], ...)

[A different tack is of course to do analytical calculations without Feynman diagrams techniques. See Kosower's seminar for a very recent approach.]

Tree-level calculations have of course limited accuracy. Fully automated procedures for loop calculations, however, do not yet exist.

Calculational tools

Of course, one cannot go on calculating traces and convolutions for thousands of diagrams by hand.

Three main classes of tools are at our disposal for multi-particle, tree level calculations:

- Tools for computer-aided analytical manipulations (Schoonschip M. Veltman, FORM [www.nikhef.nl/~form/], Mathematica/FeynCalc [www.feyncalc.org], ...)
- Tools for generating, calculating Feynman diagrams, and automatically integrating over phase space (CompHEP [theory.sinp.msu.ru/comphep], MadGraph/MadEvent [madgraph.hep.uiuc.edu], Sherpa, ...)
- Tools for calculating amplitudes numerically directly from the lagrangian (Alpha/ALPGEN [home.cern.ch/mlm/alpgen], ...)

[A different tack is of course to do analytical calculations without Feynman diagrams techniques. See Kosower's seminar for a very recent approach.]

Tree-level calculations have of course limited accuracy. Fully automated procedures for loop calculations, however, do not yet exist.

Calculational tools

Of course, one cannot go on calculating traces and convolutions for thousands of diagrams by hand.

Three main classes of tools are at our disposal for multi-particle, tree level calculations:

- Tools for computer-aided analytical manipulations (Schoonschip M. Veltman, FORM [www.nikhef.nl/~form/], Mathematica/FeynCalc [www.feyncalc.org], ...)
- Tools for generating, calculating Feynman diagrams, and automatically integrating over phase space (CompHEP [theory.sinp.msu.ru/comphep], MadGraph/MadEvent [madgraph.hep.uiuc.edu], Sherpa, ...)
- Tools for calculating amplitudes numerically directly from the lagrangian (Alpha/ALPGEN [home.cern.ch/mlm/alpgen], ...)

[A different tack is of course to do analytical calculations without Feynman diagrams techniques. See Kosower's seminar for a very recent approach.]

Tree-level calculations have of course limited accuracy. Fully automated procedures for loop calculations, however, do not yet exist.

Calculational tools

Of course, one cannot go on calculating traces and convolutions for thousands of diagrams by hand.

Three main classes of tools are at our disposal for multi-particle, tree level calculations:

- Tools for computer-aided analytical manipulations (Schoonschip M. Veltman, FORM [www.nikhef.nl/~form/], Mathematica/FeynCalc [www.feyncalc.org], ...)
- Tools for generating, calculating Feynman diagrams, and automatically integrating over phase space (CompHEP [theory.sinp.msu.ru/comphep], MadGraph/MadEvent [madgraph.hep.uiuc.edu], Sherpa, ...)
- Tools for calculating amplitudes numerically directly from the lagrangian (Alpha/ALPGEN [home.cern.ch/mlm/alpgen], ...)

[A different tack is of course to do analytical calculations without Feynman diagrams techniques. See Kosower's seminar for a very recent approach.]

Tree-level calculations have of course limited accuracy. Fully automated procedures for loop calculations, however, do not yet exist.

Calculational tools

Of course, one cannot go on calculating traces and convolutions for thousands of diagrams by hand.

Three main classes of tools are at our disposal for multi-particle, tree level calculations:

- Tools for computer-aided analytical manipulations (Schoonschip M. Veltman, FORM [www.nikhef.nl/~form/], Mathematica/FeynCalc [www.feyncalc.org], ...)
- Tools for generating, calculating Feynman diagrams, and automatically integrating over phase space (CompHEP [theory.sinp.msu.ru/comphep], MadGraph/MadEvent [madgraph.hep.uiuc.edu], Sherpa, ...)
- Tools for calculating amplitudes numerically directly from the lagrangian (Alpha/ALPGEN [home.cern.ch/mlm/alpgen], ...)

> [A different tack is of course to do analytical calculations without Feynman diagrams techniques. See Kosower's seminar for a very recent approach.]

Tree-level calculations have of course limited accuracy. Fully automated procedures for loop calculations, however, do not yet exist.

Calculational tools

Of course, one cannot go on calculating traces and convolutions for thousands of diagrams by hand.

Three main classes of tools are at our disposal for multi-particle, tree level calculations:

- Tools for computer-aided analytical manipulations (Schoonschip M. Veltman, FORM [www.nikhef.nl/~form/], Mathematica/FeynCalc [www.feyncalc.org], ...)
- Tools for generating, calculating Feynman diagrams, and automatically integrating over phase space (CompHEP [theory.sinp.msu.ru/comphep], MadGraph/MadEvent [madgraph.hep.uiuc.edu], Sherpa, ...)
- Tools for calculating amplitudes numerically directly from the lagrangian (Alpha/ALPGEN [home.cern.ch/mlm/alpgen], ...)

  [A different tack is of course to do analytical calculations without Feynman diagrams techniques. See Kosower's seminar for a very recent approach.]

Tree-level calculations have of course limited accuracy. Fully automated procedures for loop calculations, however, do not yet exist.

Phase Space Integration

Given a *differential* cross section, one must calculate what is really measured, i.e.

$$\sigma_{cuts} = \int d\sigma \Theta(cuts)$$

The integration difficulty can range from non-existent

$$\sigma^{e^+e^-\to\mu^+\mu^-}_{\theta_1<\theta<\theta_2} \sim \int_{\cos\theta_1}^{\cos\theta_2} (1+\cos^2\theta)\, d\cos\theta$$

to extremely elevated:

- many particles in final state
- cuts on momenta, energies, angles, invariant masses, ...
- 'almost singular' behaviour of cross section due to Breit-Wigner peaks
- convolutions with parton distribution and fragmentation functions
- ....

$\Rightarrow$ numerical integration

Phase Space Integration

Given a *differential* cross section, one must calculate what is really measured, i.e.

$$\sigma_{cuts} = \int d\sigma \Theta(cuts)$$

The integration difficulty can range from non-existent

$$\sigma^{e^+e^-\to\mu^+\mu^-}_{\theta_1<\theta<\theta_2} \sim \int_{\cos\theta_1}^{\cos\theta_2} (1+\cos^2\theta)\, d\cos\theta$$

to extremely elevated:

- many particles in final state
- cuts on momenta, energies, angles, invariant masses, ...
- 'almost singular' behaviour of cross section due to Breit-Wigner peaks
- convolutions with parton distribution and fragmentation functions
- ....

$\Rightarrow$ numerical integration

Phase Space Integration

Given a *differential* cross section, one must calculate what is really measured, i.e.

$$\sigma_{cuts} = \int d\sigma \Theta(cuts)$$

The integration difficulty can range from non-existent

$$\sigma^{e^+e^-\to\mu^+\mu^-}_{\theta_1<\theta<\theta_2} \sim \int_{\cos\theta_1}^{\cos\theta_2} (1+\cos^2\theta)\, d\cos\theta$$

to extremely elevated:

- many particles in final state
- cuts on momenta, energies, angles, invariant masses, ...
- 'almost singular' behaviour of cross section due to Breit-Wigner peaks
- convolutions with parton distribution and fragmentation functions
- ....

$\Rightarrow$ numerical integration

Numerical Phase Space Integration

Two large classes of numerical integrators: polynomial and Monte Carlo

- Polynomial integration works best with smooth (well, polynomial) integrands. Weights are constructed (depending on exact method), and the result is given by

$$I = \int_{\mathcal{V}} f(x)dx \simeq \sum_{j=1}^{N} w_j f(x_j)$$

  Convergence (i.e. uncertainty on result) goes like $\sigma_P \sim 1/N^{\frac{1}{p}}$, p being the number of dimensions

- Monte Carlo integration does not care about smoothness. The result is given by

$$I = \mathcal{V}\langle f \rangle \simeq \mathcal{V}\frac{1}{N}\sum_{j=1}^{N} f(x_j)$$

  Convergence goes like $\sigma_{MC} \sim 1/\sqrt{N}$

$\Rightarrow$ For $p > 2, 3$ Monte Carlo integration starts being faster

Monte Carlo Integration

The n-particles phase space

$$d\mathrm{Lips}_n = \delta^{(4)}\left(\sum P_{initial} - \sum P_{final}\right)\prod_{i=1}^{n}\frac{d^3 p_i}{2E_i}$$

has dimension $p = 3n - 4$. Hence, already for 3 particles in the final state Monte Carlo integration is convenient.

Further advantage: while integrating over the whole phase space any differential distribution can be calculated simultaneously, simply by binning over the appropriate variable.

Moreover, one can output unweighted events. This means that the probability of producing an event with a given set of momenta is proportional to its cross section.

$\Rightarrow$ Hence, the output looks (almost) like nature

Monte Carlo Integration

The n-particles phase space

$$d\mathrm{Lips}_n = \delta^{(4)}\left(\sum P_{initial} - \sum P_{final}\right)\prod_{i=1}^{n}\frac{d^3 p_i}{2E_i}$$

has dimension $p = 3n - 4$. Hence, already for 3 particles in the final state Monte Carlo integration is convenient.

Further advantage: while integrating over the whole phase space any differential distribution can be calculated simultaneously, simply by binning over the appropriate variable.

Moreover, one can output unweighted events. This means that the probability of producing an event with a given set of momenta is proportional to its cross section.

$\Rightarrow$ Hence, the output looks (almost) like nature

Monte Carlo Integration

The n-particles phase space

$$d\mathrm{Lips}_n = \delta^{(4)}\left(\sum P_{initial} - \sum P_{final}\right)\prod_{i=1}^{n}\frac{d^3 p_i}{2E_i}$$

has dimension $p = 3n - 4$. Hence, already for 3 particles in the final state Monte Carlo integration is convenient.

Further advantage: while integrating over the whole phase space any differential distribution can be calculated simultaneously, simply by binning over the appropriate variable.

Moreover, one can output unweighted events. This means that the probability of producing an event with a given set of momenta is proportional to its cross section.

$\Rightarrow$ Hence, the output looks (almost) like nature

Monte Carlo Integration

NB. All this looks easy and straightforward. In practice, even the simple exercise of evaluating accurately the average value for a function f can turn into a very lenghty one if the integrand is especially badly behaved (singularities, peaks, ...)

The generic name for approaches aimed at improving the convergence is 'variance reducing techniques'. The goal is to calculate an average with a small enough standard deviation, using a limited number of function evaluations N (and hence of computing time)

Now we know how to

- calculate $ab \longrightarrow n$ tree-level parton matrix elements
- integrate over the phace space and produce total and differential cross sections
- use MonteCarlos to make exclusive event generators: for every event I know what partons are out there and with what momenta. Moreover, the probability of simulating the event is set by its cross section

However

- all this is tree-level. What happens when I try to calculate loops? How many loops must/can I calculate?
- I calculate partons, but I measure hadrons. How do I fill the gap?

Now we know how to

- calculate $ab \longrightarrow n$ tree-level parton matrix elements
- integrate over the phace space and produce total and differential cross sections
- use MonteCarlos to make exclusive event generators: for every event I know what partons are out there and with what momenta. Moreover, the probability of simulating the event is set by its cross section

However

- all this is tree-level. What happens when I try to calculate loops? How many loops must/can I calculate?
- I calculate partons, but I measure hadrons. How do I fill the gap?

Now we know how to

- calculate $ab \longrightarrow n$ tree-level parton matrix elements
- integrate over the phace space and produce total and differential cross sections
- use MonteCarlos to make exclusive event generators: for every event I know what partons are out there and with what momenta. Moreover, the probability of simulating the event is set by its cross section

However

- all this is tree-level. What happens when I try to calculate loops? How many loops must/can I calculate?
- I calculate partons, but I measure hadrons. How do I fill the gap?

Now we know how to

- calculate $ab \longrightarrow n$ tree-level parton matrix elements
- integrate over the phace space and produce total and differential cross sections
- use MonteCarlos to make exclusive event generators: for every event I know what partons are out there and with what momenta. Moreover, the probability of simulating the event is set by its cross section

However

- all this is tree-level. What happens when I try to calculate loops? How many loops must/can I calculate?
- I calculate partons, but I measure hadrons. How do I fill the gap?

Now we know how to

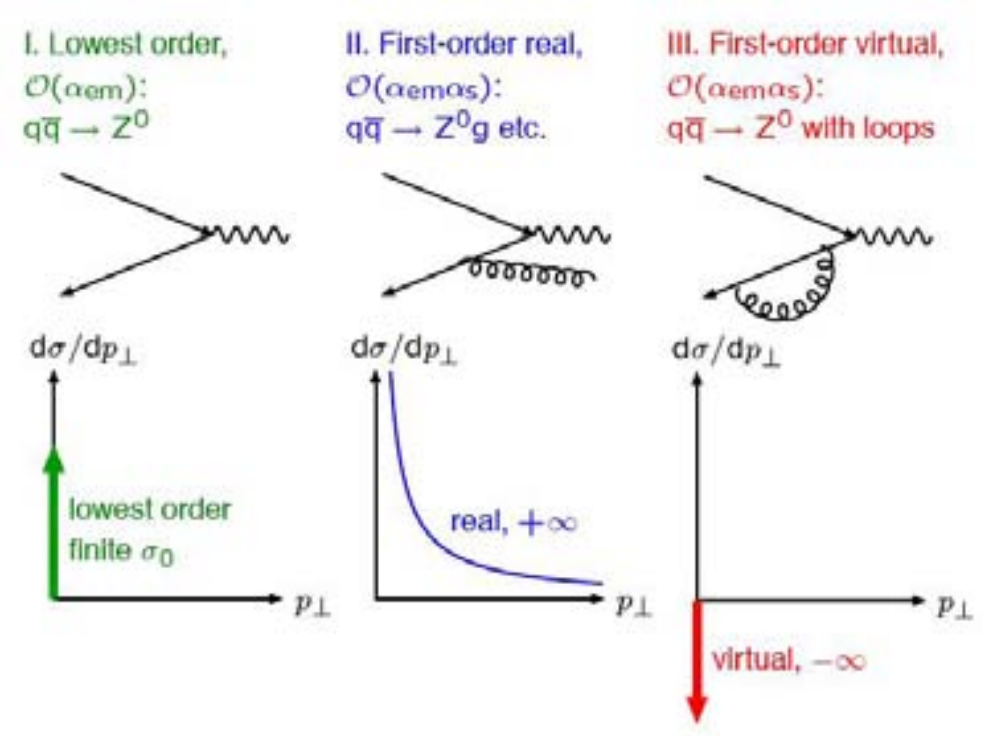
- calculate $ab \longrightarrow n$ tree-level parton matrix elements
- integrate over the phace space and produce total and differential cross sections
- use MonteCarlos to make exclusive event generators: for every event I know what partons are out there and with what momenta. Moreover, the probability of simulating the event is set by its cross section

However

- all this is tree-level. What happens when I try to calculate loops? How many loops must/can I calculate?
- I calculate partons, but I measure hadrons. How do I fill the gap?

Divergences

Calculating (by hand, aided by analytical manipulation tools) a one loop amplitude is not the end of the effort. We must still integrate over the phase space, with the additional complication that real and virtual contributions have a different number of particles in the final states and are separately divergent.

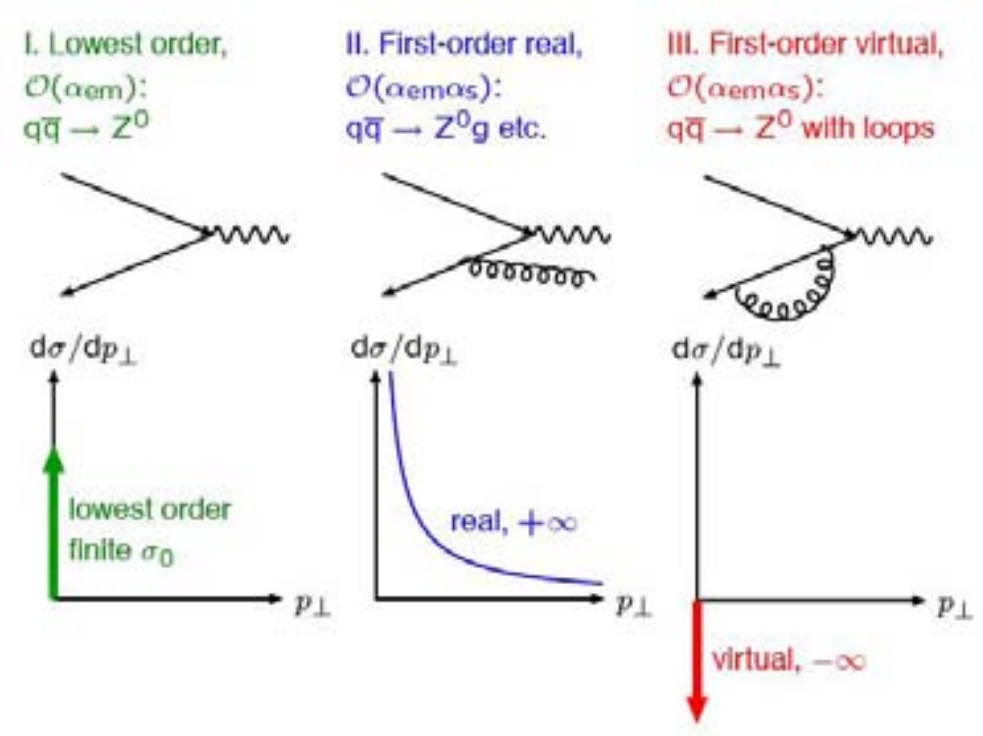


$$\sigma_{NLO} = \int_n d\sigma^{LO} + \int_{n+1} d\sigma_{Real} + \int_n d\sigma_{Virt}$$

How to calculate numerically a divergent quantity?

Divergences

Calculating (by hand, aided by analytical manipulation tools) a one loop amplitude is not the end of the effort. We must still integrate over the phase space, with the additional complication that real and virtual contributions have a different number of particles in the final states and are separately divergent.



$$\sigma_{NLO} = \int_n d\sigma^{LO} + \int_{n+1} d\sigma_{Real} + \int_n d\sigma_{Virt}$$

How to calculate numerically a divergent quantity?

Working around divergences

Consider the emission of a massless particle of 'energy' x.
The general structure of Born, Virtual and Real cross sections are:

$$\left(\frac{d\sigma}{dx}\right)_B = B\,\delta(x) \qquad \left(\frac{d\sigma}{dx}\right)_V = a\left(\frac{B}{2\epsilon} + V\right)\delta(x) \qquad \left(\frac{d\sigma}{dx}\right)_R = a\,\frac{R(x)}{x}$$

The Kinoshita-Lee-Nauenberg cancellation theorem (total cross section finite) requires $\lim_{x\to 0} R(x) = B$.

Calculating the generic infrared-safe observable O to NLO accuracy means to evaluate

$$\langle O \rangle = \lim_{\epsilon \to 0} \int_0^1 dx\, x^{-2\epsilon} O(x) \left[\left(\frac{d\sigma}{dx}\right)_B + \left(\frac{d\sigma}{dx}\right)_V + \left(\frac{d\sigma}{dx}\right)_R \right]$$

Working around divergences

Consider the emission of a massless particle of 'energy' x.
The general structure of Born, Virtual and Real cross sections are:

$$\left(\frac{d\sigma}{dx}\right)_B = B\,\delta(x) \qquad \left(\frac{d\sigma}{dx}\right)_V = a\left(\frac{B}{2\epsilon} + V\right)\delta(x) \qquad \left(\frac{d\sigma}{dx}\right)_R = a\frac{R(x)}{x}$$

The Kinoshita-Lee-Nauenberg cancellation theorem (total cross section finite) requires $\lim_{x\to 0} R(x) = B$.

Calculating the generic infrared-safe observable O to NLO accuracy means to evaluate

$$\langle O \rangle = \lim_{\epsilon \to 0} \int_0^1 dx\, x^{-2\epsilon} O(x) \left[\left(\frac{d\sigma}{dx}\right)_B + \left(\frac{d\sigma}{dx}\right)_V + \left(\frac{d\sigma}{dx}\right)_R \right]$$

Slicing Method

Take a parameter $\delta \ll 1$. The 'real part' of $\langle O \rangle$ becomes

$$\langle O \rangle_R = \int_0^\delta dx\, x^{-2\epsilon} O(x) \left(\frac{d\sigma}{dx}\right)_R + \int_\delta^1 dx\, x^{-2\epsilon} O(x) \left(\frac{d\sigma}{dx}\right)_R$$

Approximating and using the KLN limit ($R(x \to 0) = B$) we find

$$\begin{aligned} \langle O \rangle_R &= aBO(0) \int_0^\delta dx\, \frac{x^{-2\epsilon}}{x} + \int_\delta^1 dx\, O(x) \left(\frac{d\sigma}{dx}\right)_R + \mathcal{O}(\delta) \\ &= a\left(-\frac{1}{2\epsilon} + \log\delta\right) BO(0) + a\int_\delta^1 dx\, \frac{O(x)R(x)}{x} + \mathcal{O}(\delta, \epsilon) \end{aligned}$$

and, finally,

$$\langle O \rangle_{\text{slice}} = BO(0) + a\left[\left(B\log\delta + V\right) O(0) + \int_\delta^1 dx\, \frac{O(x)R(x)}{x}\right] + \mathcal{O}(\delta)$$

NB. One must choose δ so that the final result is sufficiently independent on it.

Slicing Method

Take a parameter $\delta \ll 1$. The 'real part' of $\langle O \rangle$ becomes

$$\langle O \rangle_R = \int_0^\delta dx\, x^{-2\epsilon} O(x) \left(\frac{d\sigma}{dx}\right)_R + \int_\delta^1 dx\, x^{-2\epsilon} O(x) \left(\frac{d\sigma}{dx}\right)_R$$

Approximating and using the KLN limit ($R(x \to 0) = B$) we find

$$\begin{aligned} \langle O \rangle_R &= aBO(0) \int_0^\delta dx\, \frac{x^{-2\epsilon}}{x} + \int_\delta^1 dx\, O(x) \left(\frac{d\sigma}{dx}\right)_R + \mathcal{O}(\delta) \\ &= a\left(-\frac{1}{2\epsilon} + \log\delta\right) BO(0) + a\int_\delta^1 dx\, \frac{O(x)R(x)}{x} + \mathcal{O}(\delta, \epsilon) \end{aligned}$$

and, finally,

$$\langle O \rangle_{\text{slice}} = BO(0) + a\left[\left(B\log\delta + V\right) O(0) + \int_\delta^1 dx\, \frac{O(x)R(x)}{x}\right] + \mathcal{O}(\delta)$$

NB. One must choose δ so that the final result is sufficiently independent on it.

Slicing Method

Take a parameter $\delta \ll 1$. The 'real part' of $\langle O \rangle$ becomes

$$\langle O \rangle_R = \int_0^\delta dx\, x^{-2\epsilon} O(x) \left(\frac{d\sigma}{dx}\right)_R + \int_\delta^1 dx\, x^{-2\epsilon} O(x) \left(\frac{d\sigma}{dx}\right)_R$$

Approximating and using the KLN limit ($R(x \to 0) = B$) we find

$$\begin{aligned} \langle O \rangle_R &= aBO(0) \int_0^\delta dx\, \frac{x^{-2\epsilon}}{x} + \int_\delta^1 dx\, O(x) \left(\frac{d\sigma}{dx}\right)_R + \mathcal{O}(\delta) \\ &= a\left(-\frac{1}{2\epsilon} + \log\delta\right) BO(0) + a\int_\delta^1 dx\, \frac{O(x)R(x)}{x} + \mathcal{O}(\delta, \epsilon) \end{aligned}$$

and, finally,

$$\langle O \rangle_{\text{slice}} = BO(0) + a\left[\left(B\log\delta + V\right) O(0) + \int_\delta^1 dx\, \frac{O(x)R(x)}{x}\right] + \mathcal{O}(\delta)$$

NB. One must choose δ so that the final result is sufficiently independent on it.

Subtraction Method

Rewrite the 'real part' of $\langle O \rangle$ as follows:

$$\begin{aligned}
\langle O \rangle_R &= a \int_0^1 \frac{dx}{x^{1+2\epsilon}} R(x) O(x) \\
&= a \int_0^1 dx \frac{BO(0)}{x^{1+2\epsilon}} + a \int_0^1 \frac{O(x)R(x) - BO(0)}{x^{1+2\epsilon}} \\
&= -a \frac{B}{2\epsilon} O(0) + a \int_0^1 \frac{O(x)R(x) - BO(0)}{x}
\end{aligned}$$

Hence

$$\langle O \rangle_{\text{sub}} = BO(0) + a \left[VO(0) + \int_0^1 \frac{O(x)R(x) - BO(0)}{x} \right]$$

Exact method. Usually preferred in modern implementations.

Subtraction Method

Rewrite the 'real part' of $\langle O \rangle$ as follows:

$$\begin{aligned}
\langle O \rangle_R &= a \int_0^1 \frac{dx}{x^{1+2\epsilon}} R(x) O(x) \\
&= a \int_0^1 dx \frac{BO(0)}{x^{1+2\epsilon}} + a \int_0^1 \frac{O(x)R(x) - BO(0)}{x^{1+2\epsilon}} \\
&= -a \frac{B}{2\epsilon} O(0) + a \int_0^1 \frac{O(x)R(x) - BO(0)}{x}
\end{aligned}$$

Hence

$$\langle O \rangle_{\text{sub}} = BO(0) + a \left[VO(0) + \int_0^1 \frac{O(x)R(x) - BO(0)}{x} \right]$$

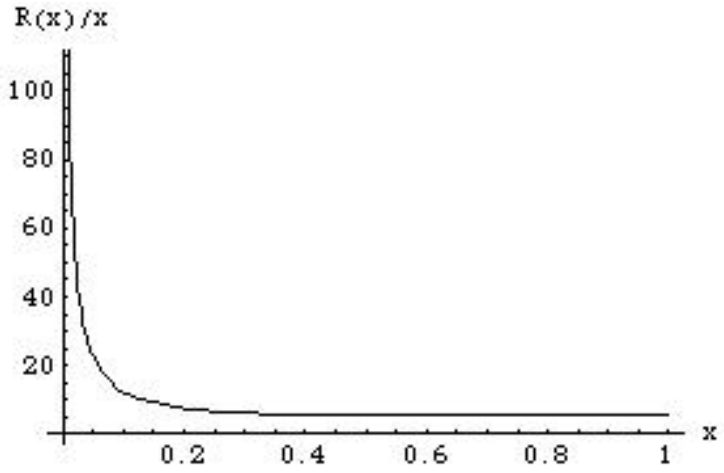
Exact method. Usually preferred in modern implementations.

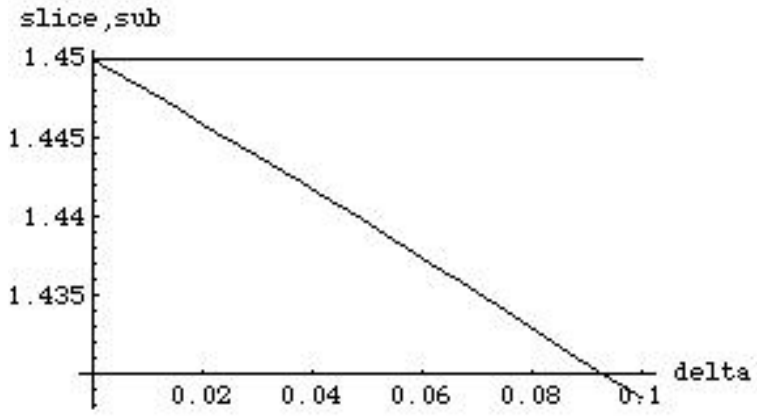
Slicing vs. Subtraction

Take

$$B = 1 \qquad V = 1 \qquad R(x) = 1 + 2x + 3x^2 \qquad a = 0.1$$

Let's use $O(x) = 1$. This will give 'total cross section'





$R(x)/x$ obviously not integrable by itself, but combination with virtual cross section gives a finite result (K-factor = 1.45)

[NB. The smaller δ, the more difficult the numerical integration of course!!]

Event generators vs. Integrators

An 'integrator' allows in principle the calculation of any well-defined differential or total cross section in pQCD. There are however a number of limitations:

1. the cancellation of singularities must be carefully studied analytically. The numerical implementation can be cumbersome
2. in pQCD we deal with partons. The experimentalists measure hadrons
3. even NLO calculations only deal with a finite number of particles (i.e. 'fixed order' calculations). However, the number of emitted particles is of course unlimited (and actually large in soft/collinear regions. i.e. need for 'all-orders' calculations)

We can do something about points [2] and [3] (of course, paying a price for it).

We trade a full quantum mechanical calculation for a "classical-like" approximation. We can then generate exclusive events on a probabilistic base, iterate - and therefore resum - basic interactions, and simulate, via more or less refined models, also the non-perturbative transition from partons to hadrons.

Event generators vs. Integrators

An 'integrator' allows in principle the calculation of any well-defined differential or total cross section in pQCD. There are however a number of limitations:

1. the cancellation of singularities must be carefully studied analytically. The numerical implementation can be cumbersome
2. in pQCD we deal with partons. The experimentalists measure hadrons
3. even NLO calculations only deal with a finite number of particles (i.e. 'fixed order' calculations). However, the number of emitted particles is of course unlimited (and actually large in soft/collinear regions. i.e. need for 'all-orders' calculations)

We can do something about points [2] and [3] (of course, paying a price for it).

We trade a full quantum mechanical calculation for a "classical-like" approximation. We can then generate exclusive events on a probabilistic base, iterate - and therefore resum - basic interactions, and simulate, via more or less refined models, also the non-perturbative transition from partons to hadrons.

Event generators vs. Integrators

An 'integrator' allows in principle the calculation of any well-defined differential or total cross section in pQCD. There are however a number of limitations:

1. the cancellation of singularities must be carefully studied analytically. The numerical implementation can be cumbersome
2. in pQCD we deal with partons. The experimentalists measure hadrons
3. even NLO calculations only deal with a finite number of particles (i.e. 'fixed order' calculations). However, the number of emitted particles is of course unlimited (and actually large in soft/collinear regions. i.e. need for 'all-orders' calculations)

We can do something about points [2] and [3] (of course, paying a price for it).

We trade a full quantum mechanical calculation for a "classical-like" approximation. We can then generate exclusive events on a probabilistic base, iterate - and therefore resum - basic interactions, and simulate, via more or less refined models, also the non-perturbative transition from partons to hadrons.

Event generators vs. Integrators

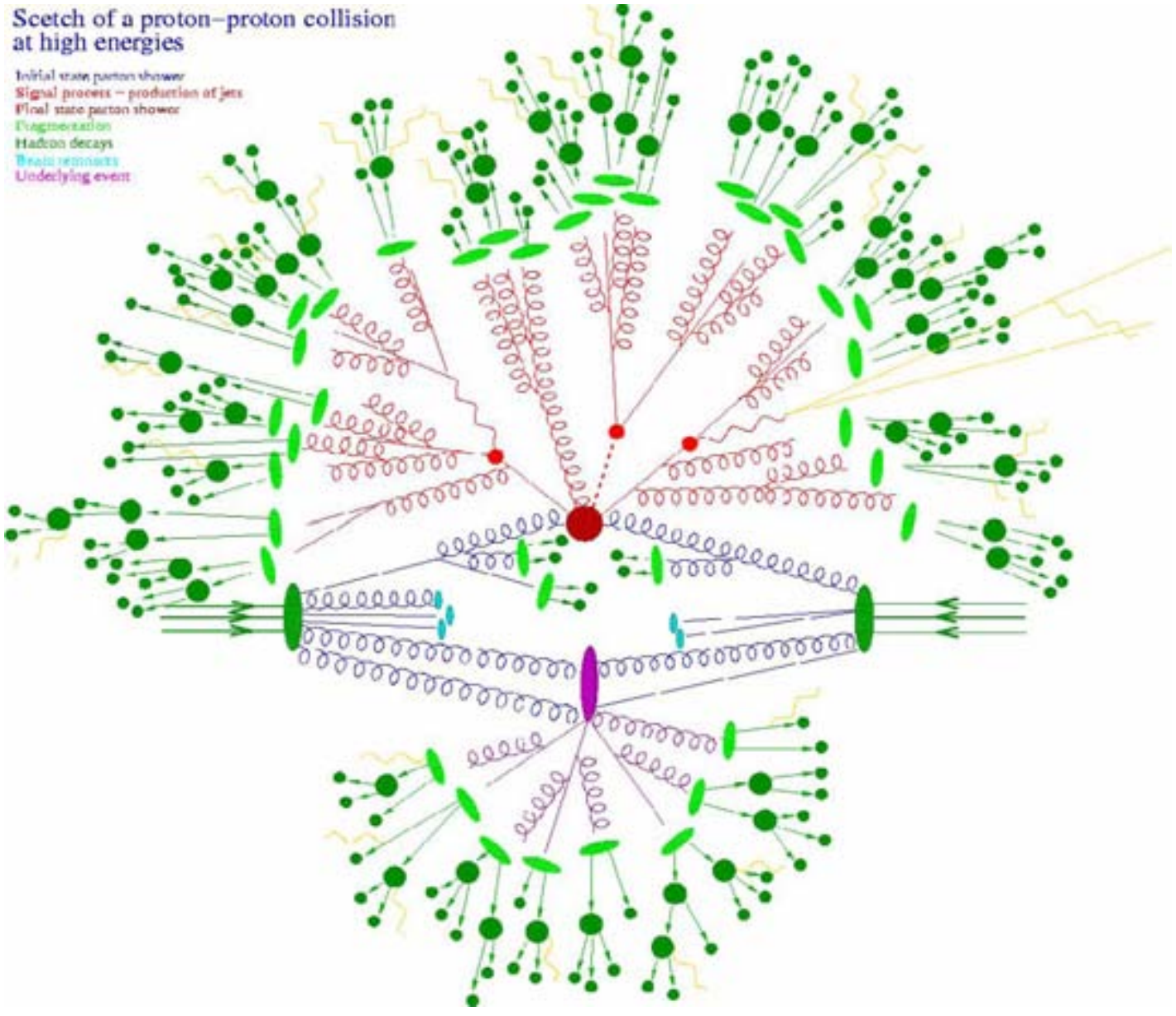
An 'integrator' allows in principle the calculation of any well-defined differential or total cross section in pQCD. There are however a number of limitations:

1. the cancellation of singularities must be carefully studied analytically. The numerical implementation can be cumbersome
2. in pQCD we deal with partons. The experimentalists measure hadrons
3. even NLO calculations only deal with a finite number of particles (i.e. 'fixed order' calculations). However, the number of emitted particles is of course unlimited (and actually large in soft/collinear regions. i.e. need for 'all-orders' calculations)

We can do something about points [2] and [3] (of course, paying a price for it).

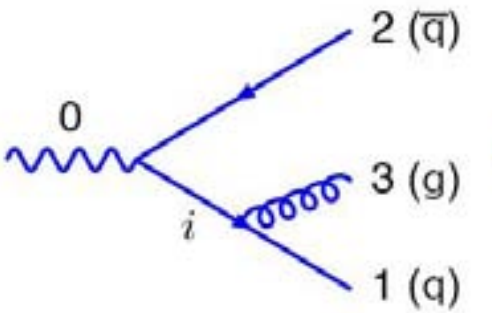
We trade a full quantum mechanical calculation for a "classical-like" approximation. We can then generate exclusive events on a probabilistic base, iterate - and therefore resum - basic interactions, and simulate, via more or less refined models, also the non-perturbative transition from partons to hadrons.

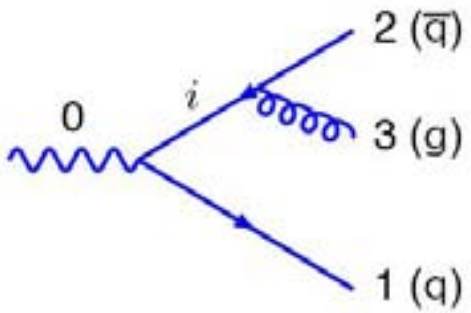
A HEP event


Sceetch of a proton–proton collision
at high energies
Initial state parton shower
Signal process – production of jets
Final state parton shower
Fragmentation
Hadron decays
Beam remnants
Underlying event

Event generators

From Matrix Elements to Parton Showers





$e^+e^- \to q\bar{q}g$

$x_j = 2E_j/E_{\mathrm{cm}} \Rightarrow$
$x_1 + x_2 + x_3 = 2$

$$m_q = 0: \quad \frac{d\sigma_{ME}}{\sigma_0} = \frac{\alpha_s}{2\pi}\frac{4}{3}\frac{x_1^2 + x_2^2}{(1-x_1)(1-x_2)}\,dx_1\,dx_2$$

Rewrite for $x_2 \to 1$, i.e. q–g collinear limit:

$$1 - x_2 = \frac{m_{13}^2}{E_{\mathrm{cm}}^2} = \frac{Q^2}{E_{\mathrm{cm}}^2} \quad \Rightarrow \quad dx_2 = \frac{dQ^2}{E_{\mathrm{cm}}^2}$$

$$x_1 \approx z \quad \Rightarrow \quad dx_1 \approx dz$$

$$x_3 \approx 1 - z$$

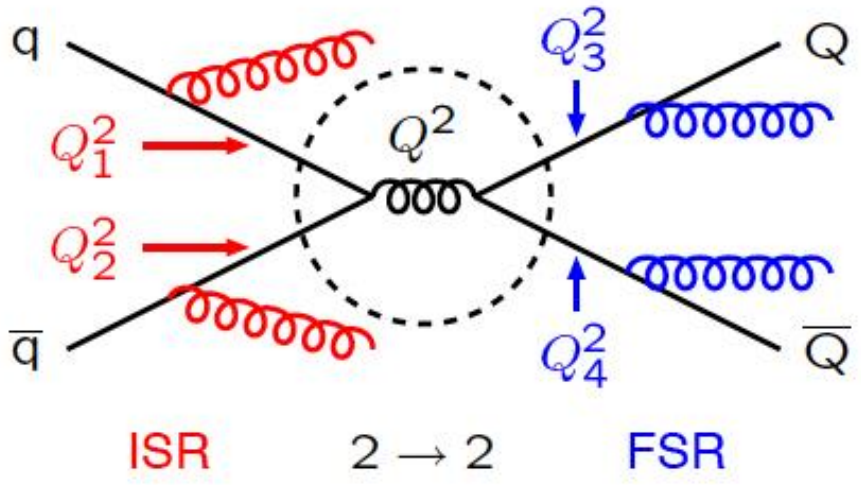
$$\Rightarrow \; d\mathcal{P} = \frac{d\sigma}{\sigma_0} = \frac{\alpha_s}{2\pi}\frac{dx_2}{(1-x_2)}\frac{4}{3}\frac{x_2^2 + x_1^2}{(1-x_1)}\,dx_1 \approx \frac{\alpha_s}{2\pi}\frac{dQ^2}{Q^2}\frac{4}{3}\frac{1+z^2}{1-z}\,dz$$

T. Sjostrand, http://agenda.cern.ch/fullAgenda.php?ida=a042790

Event generators

The combination of the probabilistic parton shower with a hard scattering process gives the (simulation of) a full partonic event:

$$2 \to n \;=\; (2 \to 2) \oplus \text{ISR} \oplus \text{FSR}$$



FSR = Final-State Rad.;
timelike shower
$Q_i^2 \sim m^2 > 0$ decreasing

ISR = Initial-State Rad.;
spacelike shower
$Q_i^2 \sim -m^2 > 0$ increasing

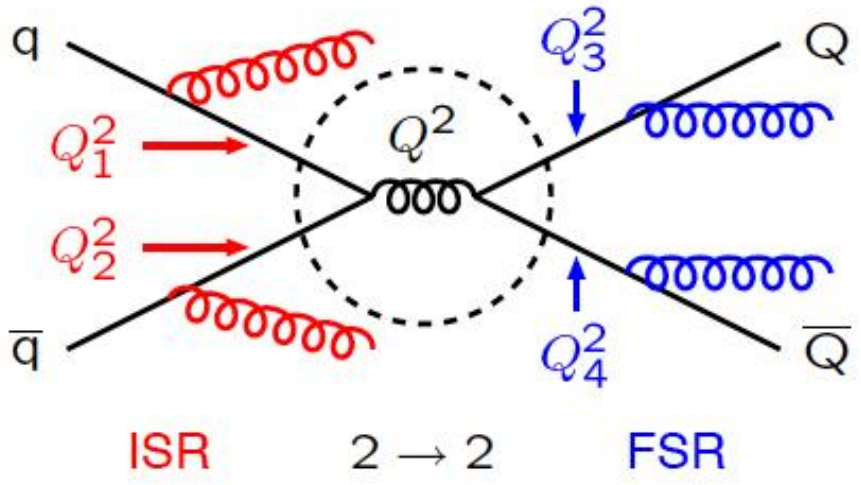
$2 \to 2$ = hard scattering (on-shell):

A hadron level event generator like PYTHIA or HERWIG will then include Parton Distribution Functions in the initial state, and hadronization and decays in the final state, hence fully simulating a high energy event

Event generators

The combination of the probabilistic parton shower with a hard scattering process gives the (simulation of) a full partonic event:

$$2 \to n \;=\; (2 \to 2) \oplus \text{ISR} \oplus \text{FSR}$$



FSR = Final-State Rad.;
timelike shower
$Q_i^2 \sim m^2 > 0$ decreasing

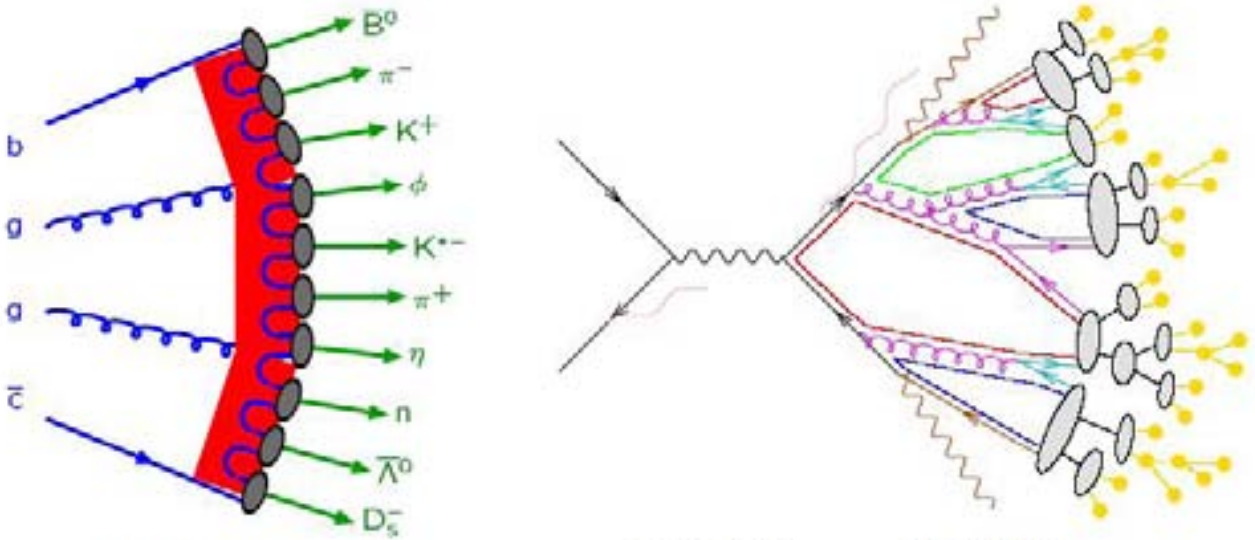
ISR = Initial-State Rad.;
spacelike shower
$Q_i^2 \sim -m^2 > 0$ increasing

$2 \to 2$ = hard scattering (on-shell):

A hadron level event generator like PYTHIA or HERWIG will then include Parton Distribution Functions in the initial state, and hadronization and decays in the final state, hence fully simulating a high energy event

Hadronization

String vs. Cluster



program	PYTHIA	HERWIG
model	string	cluster
energy–momentum picture	powerful	simple
	predictive	unpredictive
parameters	few	many
flavour composition	messy	simple
	unpredictive	in-between
parameters	many	few

"There ain't no such thing as a parameter-free *good* description"

Conclusions

- Lattice calculations are in principle "exact". However, they are still limited by techniques and/or available computing power
- The techniques for automating multi-particle tree-level calculations have greately improved in recent years. $ab \to \sim 8$ partons is feasible. But is it also reliable? What about loops?
- Two-loop calculations are becoming more common in QCD. However, they are still technically demanding (no working phase space integration yet) and no really automated/numerical approaches are available
- Parton shower generators can replace to some extent fixed order calculations in regions where resummation is important. Moreover, they can easily be interfaced to hadronization models. Their theoretical accuracy is however limited (usually LO + LL + some NLL). Proper 'matching' to NLO calculations is only now becoming more common.

Certainly enough, no single tool can satisfy all the needs. The advances in QCD testing/understanding are most certainly due to the development of a broad range of numerical tools that have allowed extensive comparisons to all sorts of experimental data

Conclusions

- Lattice calculations are in principle "exact". However, they are still limited by techniques and/or available computing power
- The techniques for automating multi-particle tree-level calculations have greately improved in recent years. $ab \to \sim 8$ partons is feasible. But is it also reliable? What about loops?
- Two-loop calculations are becoming more common in QCD. However, they are still technically demanding (no working phase space integration yet) and no really automated/numerical approaches are available
- Parton shower generators can replace to some extent fixed order calculations in regions where resummation is important. Moreover, they can easily be interfaced to hadronization models. Their theoretical accuracy is however limited (usually LO + LL + some NLL). Proper 'matching' to NLO calculations is only now becoming more common.

Certainly enough, no single tool can satisfy all the needs. The advances in QCD testing/understanding are most certainly due to the development of a broad range of numerical tools that have allowed extensive comparisons to all sorts of experimental data

Conclusions

- Lattice calculations are in principle "exact". However, they are still limited by techniques and/or available computing power
- The techniques for automating multi-particle tree-level calculations have greately improved in recent years. $ab \rightarrow \sim 8$ partons is feasible. But is it also reliable? What about loops?
- Two-loop calculations are becoming more common in QCD. However, they are still technically demanding (no working phase space integration yet) and no really automated/numerical approaches are available
- Parton shower generators can replace to some extent fixed order calculations in regions where resummation is important. Moreover, they can easily be interfaced to hadronization models. Their theoretical accuracy is however limited (usually LO + LL + some NLL). Proper 'matching' to NLO calculations is only now becoming more common.

Certainly enough, no single tool can satisfy all the needs. The advances in QCD testing/understanding are most certainly due to the development of a broad range of numerical tools that have allowed extensive comparisons to all sorts of experimental data

Conclusions

- Lattice calculations are in principle "exact". However, they are still limited by techniques and/or available computing power
- The techniques for automating multi-particle tree-level calculations have greately improved in recent years. $ab \to \sim 8$ partons is feasible. But is it also reliable? What about loops?
- Two-loop calculations are becoming more common in QCD. However, they are still technically demanding (no working phase space integration yet) and no really automated/numerical approaches are available
- Parton shower generators can replace to some extent fixed order calculations in regions where resummation is important. Moreover, they can easily be interfaced to hadronization models. Their theoretical accuracy is however limited (usually LO + LL + some NLL). Proper 'matching' to NLO calculations is only now becoming more common.

Certainly enough, no single tool can satisfy all the needs. The advances in QCD testing/understanding are most certainly due to the development of a broad range of numerical tools that have allowed extensive comparisons to all sorts of experimental data

Conclusions

- Lattice calculations are in principle "exact". However, they are still limited by techniques and/or available computing power
- The techniques for automating multi-particle tree-level calculations have greately improved in recent years. $ab \rightarrow \sim 8$ partons is feasible. But is it also reliable? What about loops?
- Two-loop calculations are becoming more common in QCD. However, they are still technically demanding (no working phase space integration yet) and no really automated/numerical approaches are available
- Parton shower generators can replace to some extent fixed order calculations in regions where resummation is important. Moreover, they can easily be interfaced to hadronization models. Their theoretical accuracy is however limited (usually LO + LL + some NLL). Proper 'matching' to NLO calculations is only now becoming more common.

Certainly enough, no single tool can satisfy all the needs. The advances in QCD testing/understanding are most certainly due to the development of a broad range of numerical tools that have allowed extensive comparisons to all sorts of experimental data

Final Words of Warning

[. . .] The Monte Carlo simulation has become the major means of visualization of not only detector performance but also of physics phenomena. So far so good. But it often happens that the physics simulations provided by the Monte Carlo generators carry the authority of data itself. They look like data and feel like data, and if one is not careful they are accepted as if they were data.

[. . .] I am prepared to believe that the computer-literate generation (of which I am a little too old to be a member) is in principle no less competent and in fact benefits relative to us in the older generation by having these marvelous tools. They do allow one to look at, indeed visualize, the problems in new ways. But I also fear a kind of "terminal illness", perhaps traceable to the influence of television at an early age. There the way one learns is simply to passively stare into a screen and wait for the truth to be delivered. A number of physicists nowadays seem to do just this.

J.D. Bjorken

from a talk given at the 75th anniversary celebration of the Max-Planck Institute of Physics, Munich, Germany, December 10th, 1992. As quoted in: Beam Line, Winter 1992, Vol. 22, No. 4