Monte Carlo event generation: Introduction

Marek Schönherr

Institute for Particle Physics Phenomenology

Beijing, 14/07/2014

Organisation

1. Monte Carle event generation: Introduction I
   - Fixed-order calculations
   - Monte Carlo integration
2. Matching and merging I
   - Parton showers
   - Matrix element corrections
3. Matching and merging II
   - Next-to-leading order matching
   - Multijet merging
4. Monte Carle event generation: Introduction II
   - Multiple parton interactions
   - Hadronisation, hadron decays, QED corrections

Organisation

1. Tutorial I
   - Getting to know the generators
   - Fixed order event generation
2. Tutorial II
   - Matching and merging

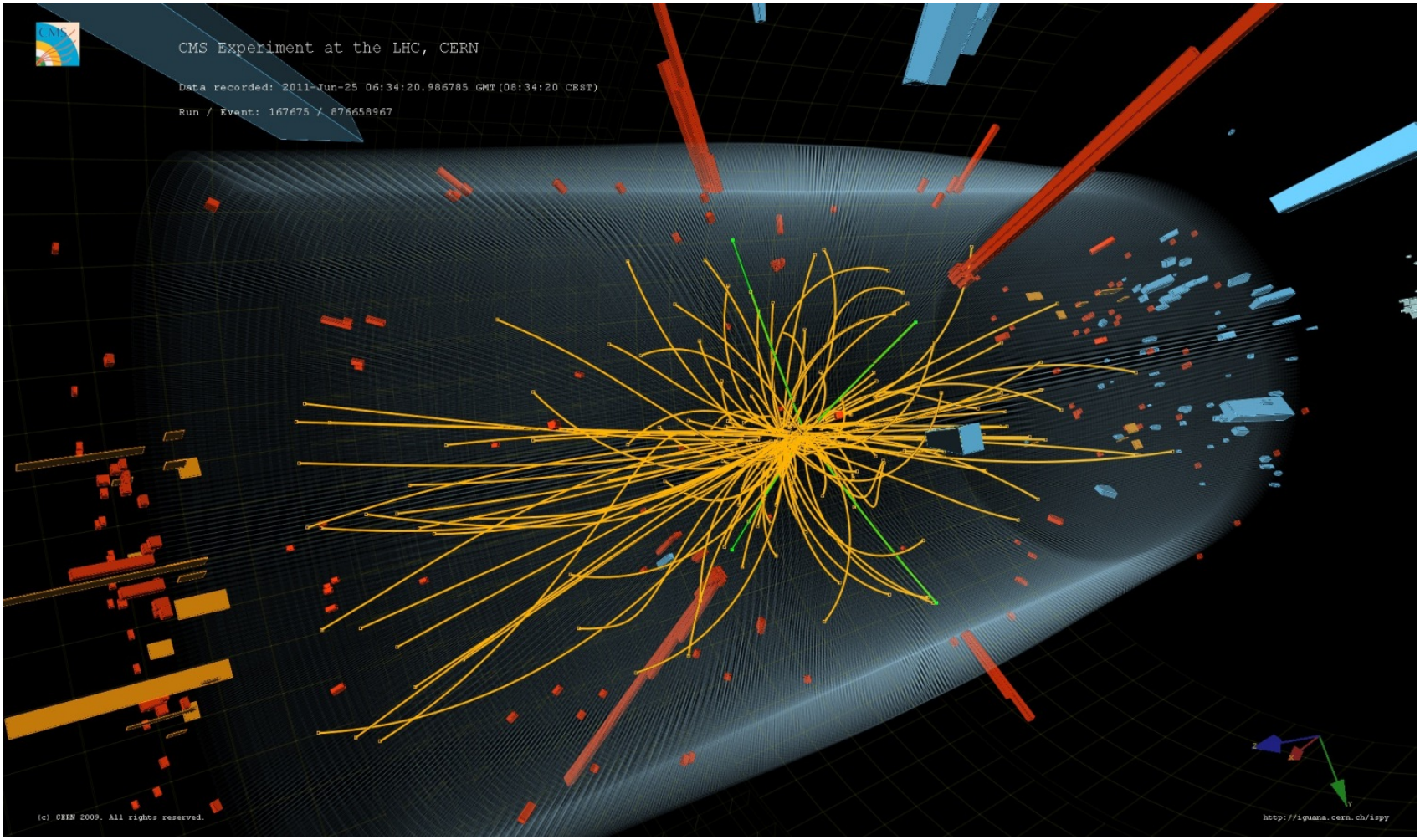
Literature

- R. K. Ellis, W. J. Stirling, B. R. Webber
  QCD and Collider Physics
  Cambridge University Press, 2003
- T. Sjöstrand, S. Mrenna, P. Z. Skands
  PYTHIA 6.4 Physics and Manual
  JHEP 05 (2006) 026
- A. Buckley et al.
  General-Purpose Event Generators for LHC Physics
  Phys. Rept. 504 (2011) 145
- R. D. Field
  Applications of Perturbative QCD
  Addison-Wesley, 1995

Monte Carle event generation: Introduction I

Monte Carle event generation: Introduction I

Task: Compute the probability for this



Task: Compute the probability for this

A typical Drell-Yan event:

$$pp \to \ell^+\ell^- + \mathcal{O}(100)\ \gamma,\ e^\pm,\ \mu^\pm,\ \pi^\pm,\ K^\pm,\ K_L,\ p^\pm,\ n,\ (\nu)$$

A typical 4ℓ event:

$$pp \to 4\ell + \mathcal{O}(150)\ \gamma,\ e^\pm,\ \mu^\pm,\ \pi^\pm,\ K^\pm,\ K_L,\ p^\pm,\ n,\ (\nu)$$

A typical hadronic $t\bar{t}$ event:

$$pp \to \mathcal{O}(700)\ \gamma,\ e^\pm,\ \mu^\pm,\ \pi^\pm,\ K^\pm,\ K_L,\ p^\pm,\ n,\ (\nu)$$

A typical semi-leptonic $t\bar{t}h$ event:

$$pp \to \ell + \mathcal{O}(1200)\ \gamma,\ e^\pm,\ \mu^\pm,\ \pi^\pm,\ K^\pm,\ K_L,\ p^\pm,\ n,\ (\nu)$$

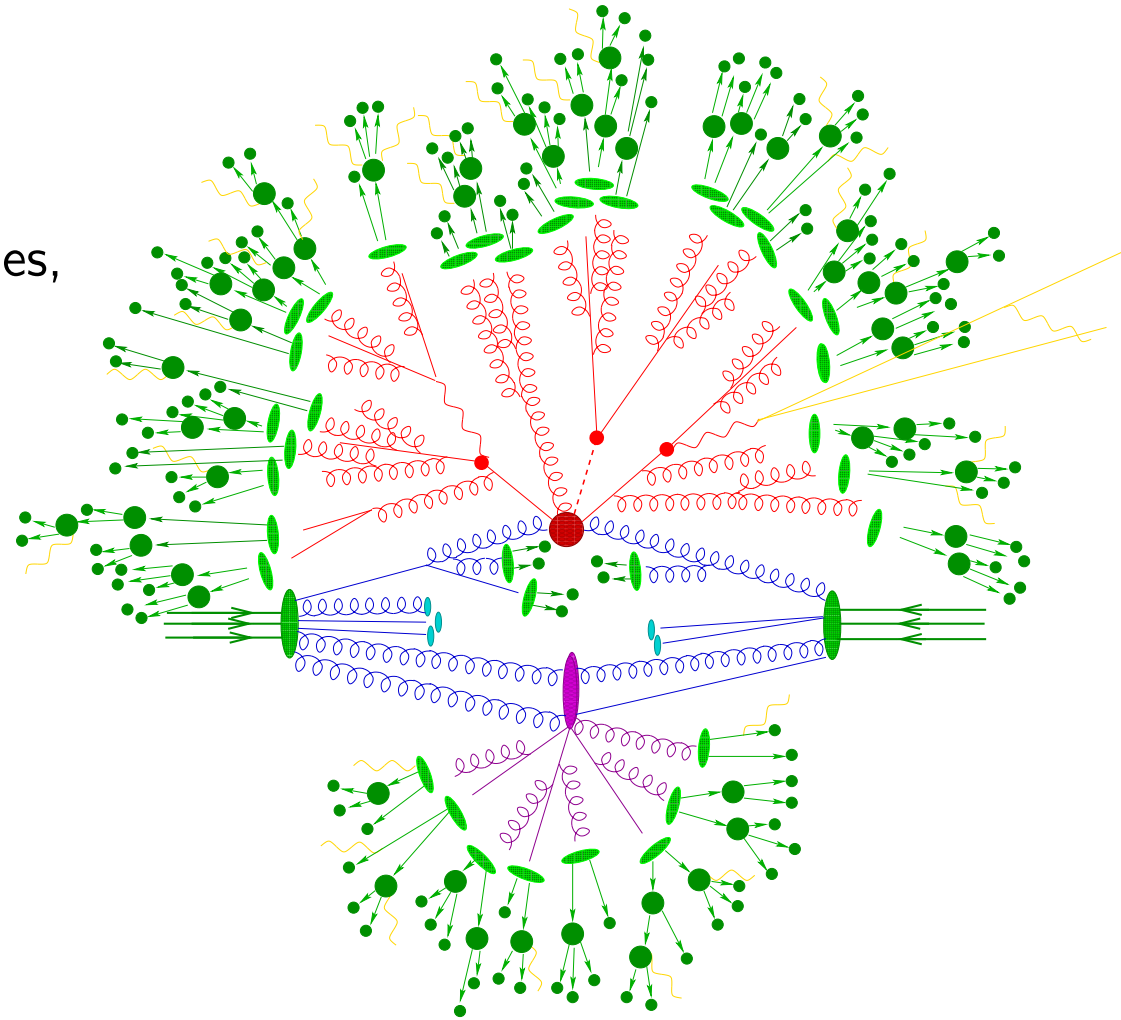
We want to make predictions that can be compared to experimental data, thus we have to describe the final state at the stage where it enters the detector.

Divide et impera

Event structure

Factorise into event stages according to characteristic scales, use relevant approximation in each regime

- Hard scattering
- Parton evolution
- Multiple interactions
- Hadronisation
- Hadron decays
- QED corrections



Event generator for LHC physics

Buckley et.al. arXiv:1101.2599

HERWIG

- originated in coherent shower studies → angular order PS
- first MC@NLO and POWHEG developments
- simple in-house ME generator & spin-correlated decay chains
- original cluster fragmentation

PYTHIA

- originated in hadronisation studies → Lund string
- leading in development of multiple interaction models
- pragmatic attitude to ME generation → external tools
- extensive PS development and earliest MEPS matching

SHERPA

- started with PS generator APACIC++ & ME generator AMEGIC++
- current MPI & hadronisation models pragmatic add-ons
- leading in development of automated MEPS merging
- automated framework for NLO calculations and MC@NLO

Monte Carle event generation: Introduction I

Fixed-order calculations

$$\langle O \rangle = \int \mathrm{d}\Phi \int_0^1 \mathrm{d}x_a \int_0^1 \mathrm{d}x_b \; f_a(x_a, \mu_F^2)\, f_b(x_b, \mu_F^2) \; \left|\mathcal{M}(ab \to X; \mu_R^2, \mu_F^2)\right|^2 O(\Phi)$$

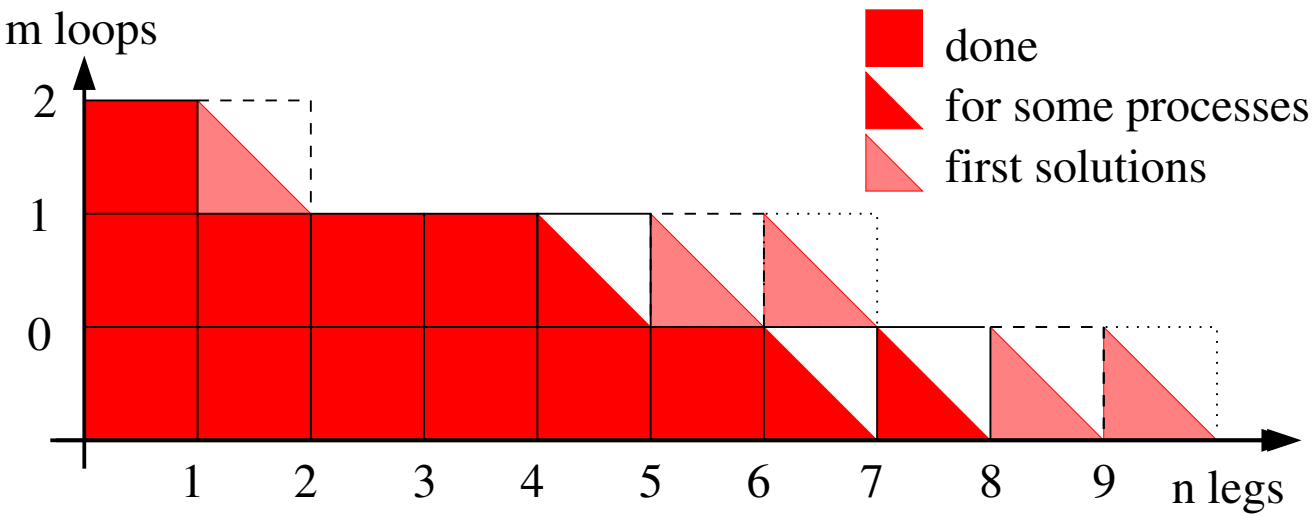
First task:

- calculate matrix element
  - 1st approximation: leading order calculation
  - 2nd approximation: next-to-leading order calculation
  - ⋮

Second task:

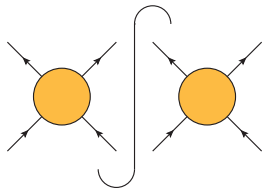
- do integral → 3rd part of this lecture

Availablity of parton-level calculations



- exact *n*-parton perturbative calculation often best, but typically hard to do for "many" external legs
- tree-level case is highly automated
- one-loop case getting there

Leading order matrix elements

LO calculation Born term: $\mathrm{B} =$ 

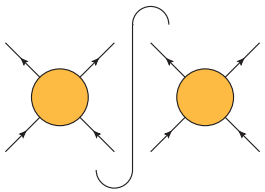
$$\langle O \rangle^{\mathrm{LO}} = \int \mathrm{d}\Phi_B \, \mathrm{B}(\Phi_B) \, O(\Phi_B)$$

Textbook:

- calculate amplitudes using Feynman diagrams
- use completeness relations to square them
- sum/average over external states (helicity and colour)

$\Rightarrow$ computational effort grows quadratically with number of diagrams

Leading order matrix elements – helicity amplitudes

LO calculation Born term: $\mathrm{B} =$ 

Improvements:

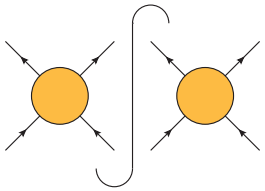
- amplitudes are complex numbers
  first compute amplitudes using Feynman diagram, then add, square

$\Rightarrow$ computational effort grows linearly with number of diagrams

- matrix multiplication is expensive
  write everything as spinor products, "gauge" vector fields
  e.g. $\bar{u}(p_1, h_1)u(p_2, h_2) \in \mathbb{C}$,
  $\bar{u}(p_1, h_1)\gamma^\mu u(p_2, h_2) \in \mathbb{C}^{3+1}$,
  $\bar{u}(p_1, h_1)\gamma^\mu u(p_2, h_2)\,\bar{u}(p_3, h_3)\gamma_\mu u(p_3, h_3) \in \mathbb{C}, \ldots$

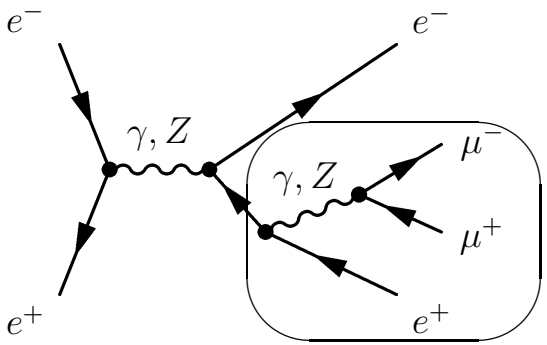
$\Rightarrow$ helicity amplitudes, explicit spin information

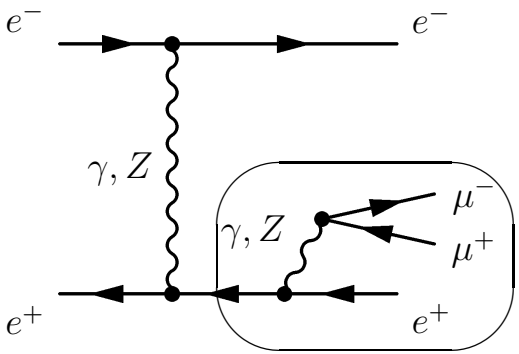
Leading order matrix elements – book-keeping

LO calculation Born term: $\mathrm{B} =$ 

Improvements:

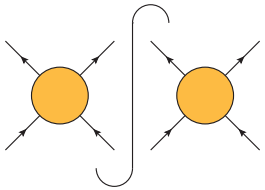
- recurring subgraphs
  many Feynman diagrams share same subgraphs





$\Rightarrow$ book-keep subamplitudes and reuse

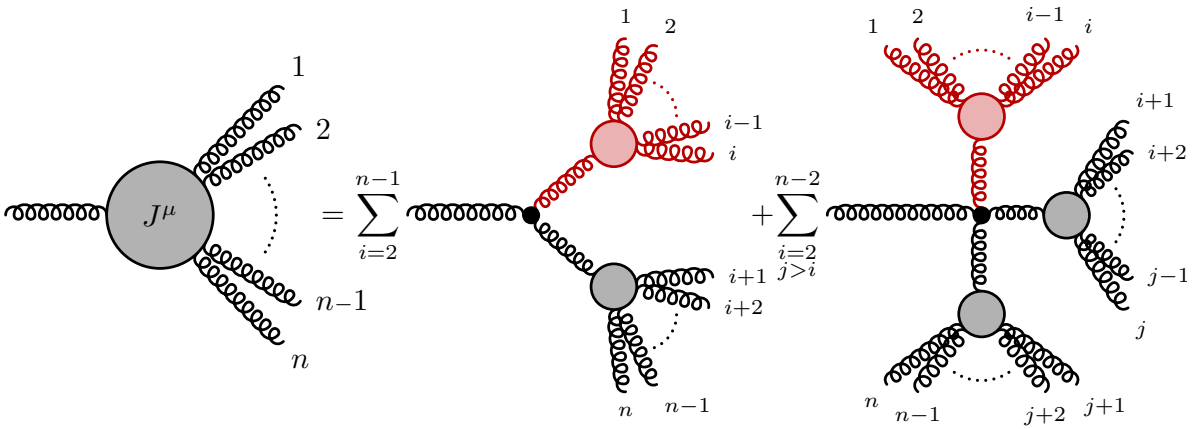
Leading order matrix elements – recurrence relations

LO calculation Born term: $\mathrm{B} =$ 

Improvements:

- complexity of Feynman diagram grows factorial with number of external legs

Berends, Giele NPB306(1988)759
Cachazo, Svrček, Witten JHEP09(2004)006
Britto, Cachazo, Feng NPB715(2005)499



$\Rightarrow$ recurrence relations for processes with many external legs

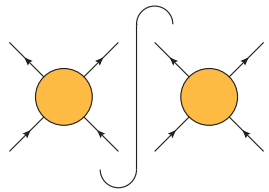
Leading order matrix elements – recurrence relations

Duhr, Höche, Maltoni JHEP08(2006)062

Computation time (s) of the $2 \to n$ gluon amplitudes for 10^4 phase space points for one random helicity and colour configuration

Final State	BG		BCF		CSW	
	CO	CD	CO	CD	CO	CD
$2g$	0.24	0.28	0.28	0.33	0.31	0.26
$3g$	0.45	0.48	0.42	0.51	0.57	0.55
$4g$	1.20	1.04	0.84	1.32	1.63	1.75
$5g$	3.78	2.69	2.59	7.26	5.95	5.96
$6g$	14.2	7.19	11.9	59.1	27.8	30.6
$7g$	58.5	23.7	73.6	646	146	195
$8g$	276	82.1	597	8690	919	1890
$9g$	1450	270	5900	127000	6310	29700
$10g$	7960	864	64000	–	48900	–

Leading order matrix elements – helicity & colour sums

LO calculation Born term: $\mathrm{B} =$ 

Improvements:

- not all helicity configurations contribute equally
  not applicable when using traces/completeness relations

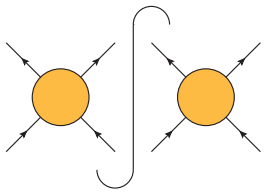
$\Rightarrow$ calculate amplitudes for only one helicity configuration, include helicity dof. in phase space integral

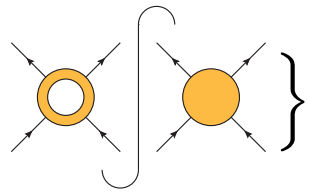
- not all colour configurations contribute equally
  not applicable when colour matrix has been separated from Lorentz-structure of matrix elements, e.g. when using helicity methods

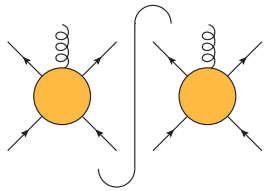
$\Rightarrow$ calculate amplitudes for only one colour configuration, include colour dof. in phase space integral

Next-to-leading order matrix elements

NLO calculation

Born term: $\mathrm{B} =$ 

Virtual terms: $\mathrm{V} = 2\,\mathrm{Re}\{$  $\}$

Real terms: $\mathrm{R} =$ 

- UV divergences in V removed by renormalization procedure
- V and R both still infrared divergent
- IR divergences cancel between V and R (KLN theorem)
  $\rightarrow$ finite result for IR safe observables

Next-to-leading order matrix elements

NLO calculation

$$\langle O \rangle^{\text{NLO}} = \int \mathrm{d}\Phi_B \Big[\mathrm{B}(\Phi_B) + \mathrm{V}(\Phi_B) \Big] O(\Phi_B) + \int \mathrm{d}\Phi_R \, \mathrm{R}(\Phi_R) \, O(\Phi_R)$$

- IR divergences cancel between V and R (KLN theorem), but live in different phase spaces
  $\rightarrow$ IR divergences in V arise from integral over loop momentum
  $\rightarrow$ IR divergences in R arise from integral over soft-collinear external momentum
- **subtraction method:** construct universal integrable terms that reproduce R in the soft-collinear limit

Subtraction method

NLO calculation

$$\langle O \rangle^{\text{NLO}} = \int \mathrm{d}\Phi_B \left[\mathrm{B}(\Phi_B) + \mathrm{V}(\Phi_B) + \int \mathrm{d}\Phi_1 \mathrm{D}(\Phi_B \cdot \Phi_1) \right] O(\Phi_B)$$
$$+ \int \mathrm{d}\Phi_R \left[\mathrm{R}(\Phi_R)\, O(\Phi_R) - \mathrm{D}(\Phi_B \cdot \Phi_1)\, O(\Phi_B) \right]$$

- **subtraction method:** construct universal integrable terms D that reproduce R in the soft-collinear limit
- holds for infrared-safe observables, i.e. $O(\Phi_R) \to O(\Phi_B)$ in IR limit
- need to add $\int \mathrm{d}\Phi_1 \mathrm{D}$ back
  $\to$ cancels divergences in V (KLN)

$\Rightarrow$ **integrands of both phase space integrals separately finite**

Subtraction method

NLO calculation

$$\langle O \rangle^{\text{NLO}} = \int \mathrm{d}\Phi_B \left[\mathrm{B}(\Phi_B) + \mathrm{V}(\Phi_B) + \int \mathrm{d}\Phi_1 \mathrm{D}(\Phi_B \cdot \Phi_1) \right] O(\Phi_B)$$
$$+ \int \mathrm{d}\Phi_R \left[\mathrm{R}(\Phi_R)\, O(\Phi_R) - \mathrm{D}(\Phi_B \cdot \Phi_1)\, O(\Phi_B) \right]$$

- **subtraction method:** construct universal integrable terms D that reproduce R in the soft-collinear limit
- holds for infrared-safe observables, i.e. $O(\Phi_R) \to O(\Phi_B)$ in IR limit
- need to add $\int \mathrm{d}\Phi_1 \mathrm{D}$ back
  $\to$ cancels divergences in V (KLN)

$\Rightarrow$ **integrands of both phase space integrals separately finite**

Subtraction method

NLO calculation

$$\langle O \rangle^{\text{NLO}} = \int \mathrm{d}\Phi_B \left[\mathrm{B}(\Phi_B) + \mathrm{V}(\Phi_B) + \int \mathrm{d}\Phi_1 \mathrm{D}(\Phi_B \cdot \Phi_1) \right] O(\Phi_B)$$
$$+ \int \mathrm{d}\Phi_R \left[\mathrm{R}(\Phi_R)\, O(\Phi_R) - \mathrm{D}(\Phi_B \cdot \Phi_1)\, O(\Phi_B) \right]$$

- **subtraction method:** construct universal integrable terms D that reproduce R in the soft-collinear limit
- holds for infrared-safe observables, i.e. $O(\Phi_R) \to O(\Phi_B)$ in IR limit
- need to add $\int \mathrm{d}\Phi_1 \mathrm{D}$ back
  $\to$ cancels divergences in V (KLN)

$\Rightarrow$ integrands of both phase space integrals separately finite

Subtraction method

NLO calculation

$$\langle O \rangle^{\text{NLO}} = \int \mathrm{d}\Phi_B \left[\mathrm{B}(\Phi_B) + \mathrm{V}(\Phi_B) + \int \mathrm{d}\Phi_1 \mathrm{D}(\Phi_B \cdot \Phi_1) \right] O(\Phi_B)$$
$$+ \int \mathrm{d}\Phi_R \left[\mathrm{R}(\Phi_R)\, O(\Phi_R) - \mathrm{D}(\Phi_B \cdot \Phi_1)\, O(\Phi_B) \right]$$

- **subtraction method:** construct universal integrable terms D that reproduce R in the soft-collinear limit
- holds for infrared-safe observables, i.e. $O(\Phi_R) \to O(\Phi_B)$ in IR limit
- need to add $\int \mathrm{d}\Phi_1 \mathrm{D}$ back
  $\to$ cancels divergences in V (KLN)

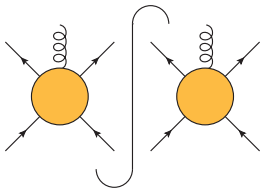
$\Rightarrow$ **integrands of both phase space integrals separately finite**

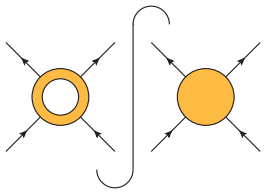
Subtraction method

Existing subtraction methods

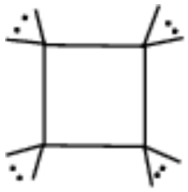
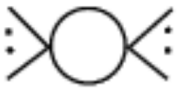
- Catani-Seymour dipole subtraction implemented in AMEGIC++, COMIX, MADDIPOLE, ... Catani, Seymour NPB485(1997)291
- Frixione-Kunszt-Signer subtraction implemented in MADFKS Frixione, Kunszt, Signer NPB467(1996)
- Antenna subtraction Kosower PRD57(1998)5410
- Nagy-Soper subtraction implemented in HELAC Nagy, Soper JHEP09(2007)114

Real and virtual correction

Real correction: $\mathrm{R} =$  $\Rightarrow$ tree-level, same techologies as for B

Virtual correction: $\mathrm{V} =$ 

- reduce 1-loop matrix element to master integrals

$\mathcal{M}^{\text{1-loop}} = D$  $+ C$  $+ B$  $+ A$  $+ R$

- compute coefficients either with tensor reduction or unitarity cuts
- problem: numerical stability, may need quad-precision

Commonly used matrix element generators (selection)

- many automated tools with different technologies

	Models	$2 \to n$	MEs	Integration	NLO
ALPHA (ALPGEN)	SM	$n = 8$	recursive	Multichannel	–
AMEGIC++ (SHERPA)	SM,BSM	$n = 6$	hel. amps.	Multichannel	Sub.
COMIX (SHERPA)	SM	$n = 8$	recursive	Multichannel	Sub.
COMPHEP	SM,BSM	$n = 4$	trace	1Channel	–
HELAC	SM	$n = 8$	recursive	Multichannel	Sub.+Loop
MADGRAPH (MADEVENT)	SM,BSM	$n = 6$	hel. amps.	Multichannel	Sub.+Loop
OMEGA (WHIZARD)	SM,BSM	$n = 8$	recursive	Multichannel	Sub.

Limiting factors:

- factorial growth of number of colour configurations
  $\to$ solution: colour dressing Duhr, Höche, Maltoni JHEP08(2006)062
- efficient phase space integration
  (matrix elements on their own are not very useful if they cannot be used to calculate cross sections)
  $\to$ see 3rd part of this lecture for some answers

One-loop matrix element generators

- for a long time only process specific codes:
  - MCFM
  - VBFNLO
  - NLOJET++
  - NJET++
- recently: (semi-)automated codes
  - BLACKHAT
  - ROCKET
  - HELAC
  - MADLOOP
  - OPENLOOPS
- typically interfaced to leading order event generators that take care of leading order matrix elements & phase space integration

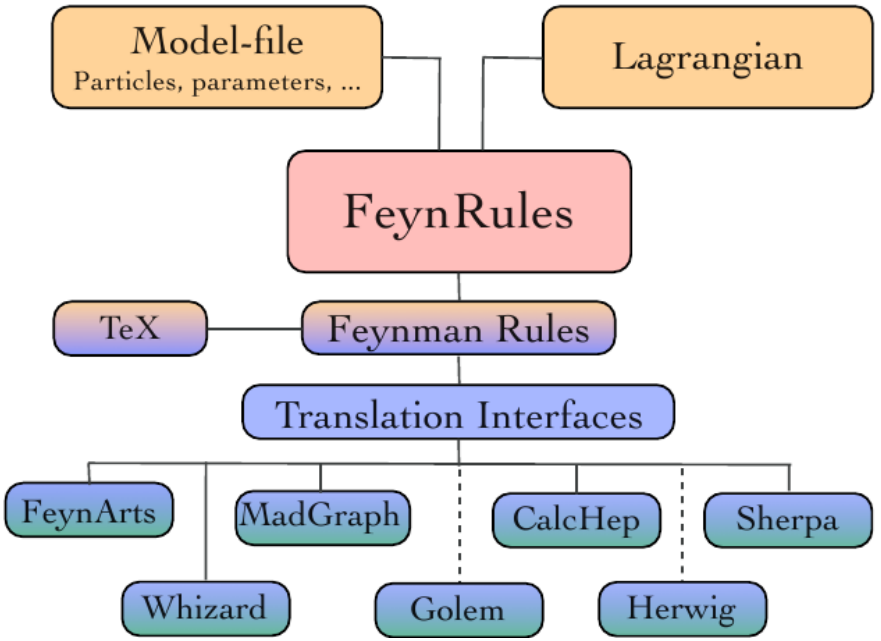
Limitations

- fixed-order implies fixed multiplicity
  higher-multiplicities are of higher order and therefore beyond approximation
- $pp \to e^+e^-$ calculation is always a $pp \to e^+e^- + X$ calculation (inclusive in X)
- breakdown of convergence of perturbation theory when phase space integration generates large logarithms in every order
  $\to$ need to rearrange pert. expansion to resum large logarithms
  $\to$ parton showers are one solution
  $\to$ see lecture by Stefan Prestel
- experimental definitions rely on observable hadrons
  $\to$ need hadron level generators

New physics models – FeynRules

Christensen, Duhr arXiv:0806.4194

- most ME generators suited for any physics model, but implementing Feynman rules tedious and error-prone
- **automated by FeynRules package**
- extracts vertices from Lagrangian
  based on minimal information about particle content
- writes generator-specific output permitting easy cross-checks



Monte Carle event generation: Introduction I

Aim

$$\langle O \rangle = \int \mathrm{d}\Phi \int_0^1 \mathrm{d}x_a \int_0^1 \mathrm{d}x_b \; f_a(x_a, \mu_F^2)\, f_b(x_b, \mu_F^2) \; \left|\mathcal{M}(ab \to X; \mu_R^2, \mu_F^2)\right|^2 O(\Phi)$$

So far, we have seen how to construct the integrand. But to compute observables we need to do the integral. As the phase space structure of the observable might be infinitely complex analytic integration is out of the question.

Turn to numerical methods:

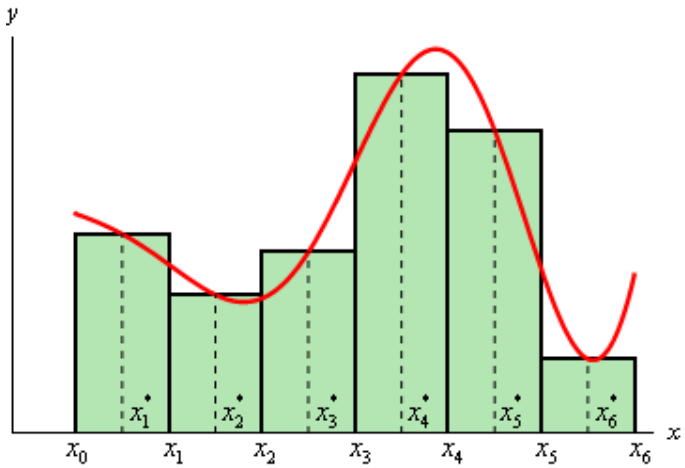
a) text book methods
b) Monte Carlo integration

Start with simple example

$$I_f^{(a,b)} = \int_a^b \mathrm{d}x \; f(x)$$

Integral of an arbitrary function on 1 dim. phase space with boundaries

Text book methods



Find solution to

$$I_f^{(a,b)} = \int_a^b \mathrm{d}x\, f(x)$$

Rectangle method

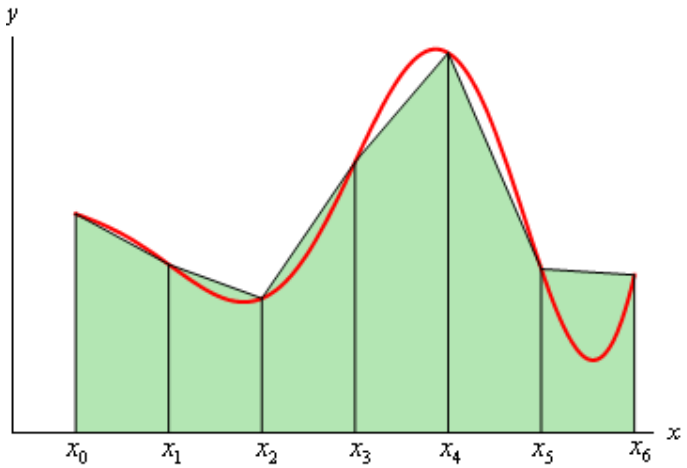
- divide interval $[a, b]$ in n subintervals of size $\Delta x = (b-a)/n$

$$I_f^{(a,b)} \approx \sum_{i=0}^{n-1} f(x_i)\Delta x \qquad \text{with} \qquad x_i \in \Big[a + i\Delta x, a + (i+1)\Delta x\Big]$$

  $\Rightarrow$ replace integration by sum over rectangles

- error scales with $\frac{1}{n}$ (n number of function evaluations)

Text book methods



Find solution to

$$I_f^{(a,b)} = \int_a^b \mathrm{d}x\, f(x)$$

Trapezoid method

- divide interval $[a, b]$ in n subintervals of size $\Delta x = (b - a)/n$

$$I_f^{(a,b)} \approx \sum_{i=0}^{n-1} \frac{f(x_i) + f(x_{i+1})}{2} \Delta x \qquad \text{with} \qquad x_i = a + i\Delta x$$

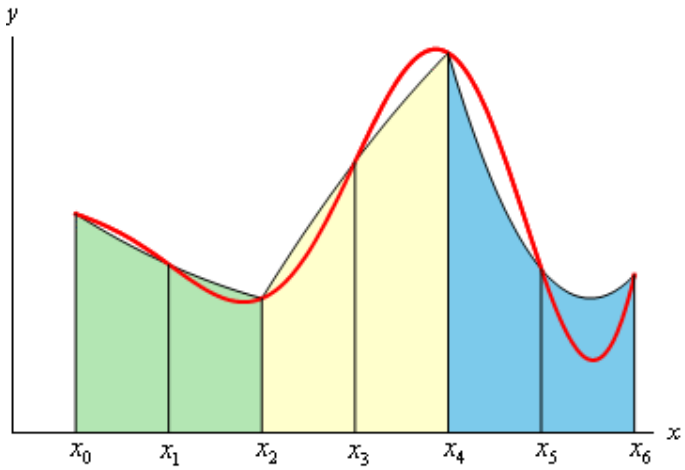
  $\Rightarrow$ replace integration by sum over trapezoids

- error scales with $\frac{1}{n^2}$ (n number of function evaluations)

Text book methods

Find solution to

$$I_f^{(a,b)} = \int_a^b \mathrm{d}x\, f(x)$$



Simpson's rule

- divide interval $[a, b]$ in $n/2$ subintervals of size $\Delta x = 2(b-a)/n$

$$I_f^{(a,b)} \approx \sum_{i=0}^{n-2} \frac{f(x_i) + 4f(x_{i+1}) + f(x_{i+2})}{6} \Delta x \qquad \text{with} \qquad x_i = a + i\frac{\Delta x}{2}$$

  $\Rightarrow$ replace integration by sum over areas under parabolas

- error scales with $\frac{1}{n^4}$ (n number of function evaluations)

Text book methods

Generalisation: Newton-Cotes method

- divide interval $[a, b]$ in n subintervals of size $\Delta x = (b - a)/n$, approximate $f(x)$ on interval by a polynomial P_p of degree p, supported at $p + 1$ equally spaced points, and weights $w_i = w_i(P_p)$

$$I_f^{(a,b)} \approx \sum_{i=0}^{n} w_i \, f(x_i) \, \Delta x \qquad \text{with} \qquad x_i = a + i\Delta x$$

- error scales with $\frac{1}{n^{2p}}$ (n number of function evaluations)

Issue:

- the above arguments work per dimension, but the N particle phase space has $d = 3N - 2$ dimensions
  $\rightarrow$ number of function calls for support grid grows as n^d

Text book methods

Generalisation: Newton-Cotes method

- divide interval $[a, b]$ in n subintervals of size $\Delta x = (b - a)/n$, approximate $f(x)$ on interval by a polynomial P_p of degree p, supported at $p + 1$ equally spaced points, and weights $w_i = w_i(P_p)$

$$I_f^{(a,b)} \approx \sum_{i=0}^{n} w_i \, f(x_i) \, \Delta x \qquad \text{with} \qquad x_i = a + i\Delta x$$

- error scales with $\frac{1}{n^{2p}}$ (n number of function evaluations)

Issue:

- the above arguments work per dimension, but the N particle phase space has $d = 3N - 2$ dimensions
  $\rightarrow$ number of function calls for support grid grows as n^d

Monte Carlo integration

$$I_f^{(a,b)} = (b-a)\langle f \rangle \qquad \text{with} \qquad \langle f \rangle \approx \frac{1}{n}\sum_{i=1}^{n} f(x_i) \qquad \text{with } x_i \in [a,b] \text{ equally distributed}$$

- evaluate integrand at random points
  $\rightarrow$ need to be equally distributed
  $\rightarrow$ (pseudo-)random number generators
- no smoothness criterion for integrand, only needs to be finite
- error scales with $\sqrt{n}$ independent of the dimension of the phase space

$$\sigma = \sqrt{\frac{\langle f^2 \rangle - \langle f \rangle^2}{N-1}}$$

$\Rightarrow$ simultaneously gives an error estimate

Monte Carlo integration

$$\langle O \rangle = \int \mathrm{d}\Phi \int_0^1 \mathrm{d}x_a \int_0^1 \mathrm{d}x_b \; f_a(x_a, \mu_F^2)\, f_b(x_b, \mu_F^2) \; \left|\mathcal{M}(ab \to N; \mu_R^2, \mu_F^2)\right|^2 O(\Phi)$$

- Monte Carlo integration now means sampling the N particle $(3N-2$ dimensional) phase space with random points Φ_N

$\Rightarrow$ produce weight $w_i = f_a(x_a, \mu_F^2)\, f_b(x_b, \mu_F^2) \; \left|\mathcal{M}(ab \to N; \mu_R^2, \mu_F^2)\right|^2$ for every $\Phi_{N,i}$

- inteprete as event of configuration $\Phi_{N,i}$ with weight w_i
- **Benefit:** simultanously project onto arbitrary many observables

$\Rightarrow$ can compute the expectation value for arbitrary many observables simultaneously

Importance sampling

- **realistic scenario**: peaked integrand

$\Rightarrow$ most of the equally distributed points in regions that contribute little
few points that carry the bulk of the integral
$\rightarrow$ large variance

- **solution:** variable transform

$$I = \int_a^b \mathrm{d}x\, g(x)\, \frac{f(x)}{g(x)} = \int_{G(a)}^{G(b)} \mathrm{d}G(x)\, w(x) \quad \text{with} \quad w(x) = \frac{f(x)}{g(x)}$$

approximate $f(x)$ with $g(x)$ with known integral $G(x)$

$\Rightarrow$ throw more points were there is a large contribution to integral

- Note: I is independent of g, σ is not

$$I = \Big[G(b) - G(a)\Big]\langle w\rangle \qquad \sigma = \Big[G(b) - G(a)\Big]\sqrt{\frac{\langle w^2\rangle - \langle w\rangle^2}{N-1}}$$

Selecting from known distribution

- Random Number Generators (RNGs) produce uniformly distributed pseudo random numbers in $[0, 1]$
- assume we want points following distribution $g(x)$ with known integral $G(x)$ and its inverse G^{-1} instead
  $\rightarrow$ probability to produce point in $[x, x + \mathrm{d}x]$ is $g(x)\mathrm{d}x$
- generate x according to

$$\int_a^x \mathrm{d}\bar{x}\, g(\bar{x}) \;=\; R \int_a^b \mathrm{d}\bar{x} g(\bar{x})$$

where R is a uniform random number in $[0, 1]$, then

$$x = G^{-1}\Big[G(a) + R\big(G(b) - G(a)\big)\Big]$$

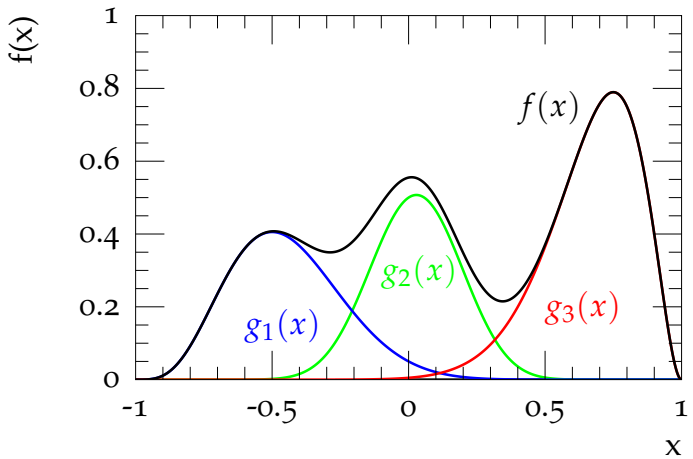
Stratified sampling

- **Problem:** Limited information on integrand
- decompose integral in M sub-integrals

$$\langle f \rangle = \sum_{i=1}^{M} \langle f_i \rangle \qquad \sigma^2 = \sum_{i=1}^{M} \sigma_i^2$$

- overall variance smallest, if variance in each contribution equal
  $\rightarrow$ sample most where fluctuations are largest
- algorithm:
  - divide integration region into bins (variable bin-size or weight)
  - adjust bins such that variance identical in all bins
- Example: VEGAS

Multichannel



- **Problem:** integrand with intricate peak structure
- combine importance sampling and stratified sampling
  - importance sample with "eigenfunctions" $g_i(x)$, each approximating one peak of the integrand, assign weight α_i

$$g(x) = \sum_{i=1}^{M} \alpha_i \, g_i(x)$$

  - optimise α_i such that variance for each $\langle f_i \rangle$ is equal

$\Rightarrow$ method of choice when integrating multileg matrix elements

Unweighting

- recall from importance sampling

$$\langle f \rangle = \frac{G(b) - G(a)}{b - a} \frac{1}{N} \sum_{i=1}^{N} w(x_i) \qquad \text{and} \qquad \sigma \propto \sqrt{\frac{\langle w^2 \rangle - \langle w \rangle^2}{N - 1}}$$

$\Rightarrow$ variance minimal if $w(x) = w = const.$, i.e. all x_i contribute the same to the integral, then

$$\langle f \rangle = \frac{G(b) - G(a)}{b - a} w \qquad \text{and} \qquad \sigma = 0$$

- however this implies $g(x) = w \cdot f(x)$ and $\langle f \rangle$ was known beforehand

Unweighting – Hit-and-miss method

- **realistically:** $g(x)$ approximates $f(x)$
  $\rightarrow$ there exists a constant c such that $c \cdot g(x) > f(x)$ for all x
- Hit-and-miss method
  - choose x_i according to $g(x)$, if necessary replace $g(x) = \sum_i \alpha_i \, g_i(x)$
  - accept x_i with probability $\frac{f(x)}{c \cdot g(x)}$, i.e. draw a uniform random number R on $[0, 1]$ and accept if $\frac{f(x)}{c \cdot g(x)} < R$
  - assign it the weight $\langle f \rangle$
- **problem:** c and $\langle f \rangle$ not known beforehand, two run strategy
  - integrate inclusive observable first, e.g. incl. cross section
  - book keep $\max\limits_{x_i} \frac{f(x)}{g(x)}$ to find minimal c and $\langle f \rangle$
  - in second run unweight and assign constant weight $\langle f \rangle$

$\Rightarrow$ optimally distributed phase space points of second run can be used to project onto observables, each x_i can be interpreted as event with same probability

- very costly if $g(x)$ not a good approximation to $f(x)$

Selecting from a Poisson distribution

- Example: Assume nuclear decay process described by $g(x)$
- nucleus can only decay if it has not yet decayed already
  must account for survival probability $\leftrightarrow$ Poisson distribution

$$\mathcal{G}(x) = g(x)\,\Delta(x,b) \qquad \text{with} \quad \Delta(x,b) = \exp\left\{-\int_x^b \mathrm{d}\bar{x}\, g(\bar{x}\right\}$$

- if $G(x)$ known, then integral of $\mathcal{G}(x)$ is

$$\int_x^b \mathrm{d}\bar{x}\,\mathcal{G}(x) = \int_x^b \mathrm{d}\bar{x}\,\frac{\mathrm{d}\Delta(\bar{x},b)}{\mathrm{d}\bar{x}} = 1-\Delta(x,b)$$

- generate events by requiring $1-\Delta(x,b) = 1-R$

$$x = G^{-1}\Big[G(b)+\log R\Big]$$

The veto algorithm

- want to sample Poisson distributed process $f(x)$, but F unknown
- Veto algorithm: Hit-or-miss method for Poisson distribution
  - choose $g(x) > f(x)$ with known G and G^{-1}
  - generate random number x according to $\mathcal{G}(x)$
  - accept with $w(x) = f(x)/g(x) < 1$
  - if rejected, continue starting from x
- probability for immediate acceptance of x is

$$\frac{f(x)}{g(x)}\, g(x)\, \exp\left\{-\int_x^b \mathrm{d}\bar{x} g(\bar{x})\right\}$$

- probability for acceptance of x after rejection at x' is

$$\frac{f(x)}{g(x)}\, g(x) \int \mathrm{d}x'\, \exp\left\{-\int_x^{x'} \mathrm{d}\bar{x} g(\bar{x})\right\} \left(1 - \frac{f(x')}{g(x')}\right) g(x')\, \exp\left\{-\int_{x'}^b \mathrm{d}\bar{x}\, g(\bar{x})\right\}$$

- for n intermediate rejections $\rightarrow$ n nested integrals $\int_x^b \int_{x'}^b \int_{x''}^b \ldots$
- disentangling yields $1/n!$ and summing over all possible rejections

$$f(x) \exp\left\{-\int_x^b \mathrm{d}\bar{x}\, g(\bar{x})\right\} \sum_{n=0}^{\infty} \frac{1}{n!} \left[\int_x^b \mathrm{d}\bar{x}\, [g(\bar{x}) - f(\bar{x})]\right]^n = f(x) \exp\left\{-\int_x^b \mathrm{d}\bar{x}\, f(\bar{x})\right\}$$

Summary: MC event generators I

Lecture:

- To calculate the probabilities of collider events they are structured/factorised into event stages.
- Fixed order matrix elements largely automated at LO and NLO level.
- Monte-Carlo integration is the method of choice for multidimensional integration. Side-product: events with a statistical interpretation that can be projected onto arbitrary observables.

Tutorial I:

- Get to know some Monte-Carlo event generators (PYTHIA8 & SHERPA).
- Use `MadGraph5_aMC@NLO` and SHERPA for fixed order event generation.

Thank you for your attention!