Introduction to Monte Carlo event generators

part I

Korinna Zapp

LIP (Lisbon), CERN

Münster 19. 10.–2. 11. 2016

Graduiertenkolleg 2149
Research Training Group

Outline

Part I

- Monte Carlo techniques
- MC event generators
- Hard process: matrix elements and phase space

Part II

- Parton Shower
- Hadronisation
- Multi-parton interactions

Part III

- Higher order corrections: matching and merging
- Additional tools and software

Outline - part I

Outline

Born cross section at hadron colliders

$$\sigma_{ab\to N} = \int_0^1 dx_a\, dx_b\, f_a(x_a,\mu_F) f_b(x_b,\mu_F) \int_{\text{cuts}} d\Phi_N \frac{1}{2\hat{s}} |\mathcal{M}(\Phi_N,\mu_F,\mu_R)|^2$$

- parton distribution functions $f(x,\mu_F)$
- N-particle phase space $d\Phi_N$
- incoming current $1/2\hat{s} = 1/2x_1x_2s$
- squared matrix element $|\mathcal{M}|^2$

  summed/averaged over polarisations
- complexity demands numerical methods for large N
- in addition: $\sigma_{\text{final state}} = \sigma_{\text{hard process}} \mathcal{P}_{\text{tot, hard process}\to\text{final state}}$
  where $\mathcal{P}_{\text{tot}} = \mathcal{P}_{\text{QCD rad.}} \mathcal{P}_{\text{hadronisation}} \mathcal{P}_{\text{decays}} \mathcal{P}_{\text{QED rad.}} \mathcal{P}_{\text{MPI}}$

Outline

Reminder: expectation and variance

expectation of a function $f(x)$:

$$E(f) = \langle f \rangle = \frac{1}{b-a} \int_b^a \mathrm{d}x\, f(x)$$

expectation is linear:

$$E(af + g) = aE(f) + E(g)$$

variance:

$$\begin{aligned} V(f) = \sigma^2 = E\left[(f - E(f))^2\right] &= E(f^2) - 2E[fE(f)] + E(f)^2 \\ &= E(f^2) - 2E(f)E(f) + E(f)^2 = E(f^2) - E(f)^2 = \langle f^2 \rangle - \langle f \rangle^2 \end{aligned}$$

for uncorrelated variables

$$V(x + y) = V(x) + V(y)$$

Monte Carlo integration

- want to calculate $I = \int_a^b \mathrm{d}x\, f(x) = (b-a)\langle f \rangle$

  f should be well-behaved, i.e. have finite $E(f)$ and $V(f)$
- pick N random numbers x_i distributed uniformly in $[a, b]$
- law of large numbers: $I = \lim_{N\to\infty} (b-a)\frac{1}{N}\sum_{i=1}^{N} f(x_i) = I_N$
- Monte Carlo estimate I_N is unbiased, i.e. $E(I_N) = I$
- central limit theorem: for large N I_N is normally distributed
- standard deviation of Monte Carlo estimate (for large N):
  $\sigma = \sqrt{V(f)/N}$

Comparison to other numerical integration methods

Convergence

- error on d-dimensional integral scales like

Monte Carlo	$\propto 1/N^{1/2}$
Trapezium rule	$\propto 1/N^{2/d}$
Simpson's rule	$\propto 1/N^{4/d}$

- MC wins for large d

Other Advantages of MC integration

- can handle arbitrarily complex integration regions
- can get first estimate with few points
- every additional point increases accuracy
- easy to estimate and monitor error

Equivalence of integration and sampling

- rewrite integral as

$$I = \int_a^b \mathrm{d}x\, f(x) = \int_a^b \mathrm{d}F(x) \qquad \text{where} \quad F(x) = \frac{\mathrm{d}f(x)}{\mathrm{d}x}$$

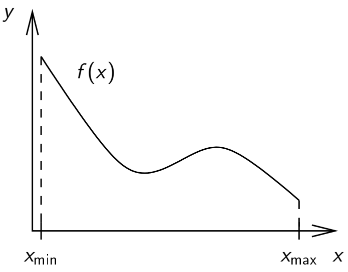
- Monte Carlo estimate of integral:

$$I_N = (b-a)\frac{1}{N}\sum_{i=1}^{N} f(x_i) = (b-a)\frac{1}{N}\sum_{i=1}^{N} f_i$$

- uniformly distributed random variables x_i with weights $f(x_i)$
- equivalent to random variables f_i distributed according to $f(x)$

Integration/Sampling in practice

- want to integrate/sample from function $f(x)$ in $[x_{\min}, x_{\max}]$
- assume $f(x)$ is positive on $[x_{\min}, x_{\max}]$ and piecewise continuous
- remember: $V(I_N) \propto \sqrt{V(f)/N}$



- for strongly varying functions convergence can be very slow

matrix elements typically of this type

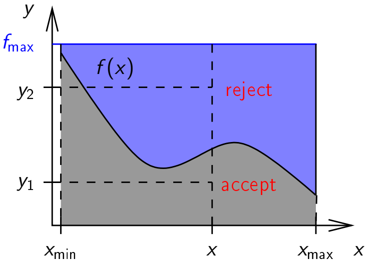
- employ variance reducing techniques

Hit-or-Miss

- need overestimate $f_{\max} \geq f(x)\ \forall\ x \in [x_{\min}, x_{\max}]$

Algorithm

1. pick random variable x from uniform distribution in $[x_{\min}, x_{\max}]$
2. pick random variable $y = R \cdot f_{\max}$
3. if $y > f(x)$ reject x and return to 1, else accept x



- $\int_{x_{\min}}^{x_{\max}} \mathrm{d}x\, f(x) = f_{\max}(x_{\max} - x_{\min}) N_{acc}/N_{tot} = A_{tot} N_{acc}/N_{tot}$
- failsafe method that always works
- slow convergence/low sampling efficiency when $I/A_{tot} \ll 1$

Direct sampling

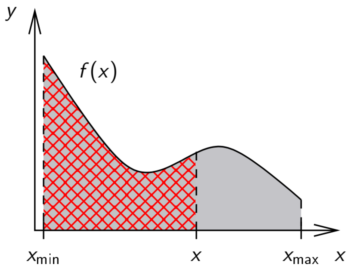
$$\int \cdots \int \mathrm{d}x_1 \ldots \mathrm{d}x_n \, f(x_1, \ldots, x_n) = \int \cdots \int \int_0^{f(x_1,\ldots,x_n)} \mathrm{d}x_1 \ldots \mathrm{d}x_n \mathrm{d}x_{n+1}$$

- an integral in n dimensions is a volume in $n+1$ dimensions
- $n = 1$: integral over $f(x)$ is the area under the curve

Direct sampling

- $n = 1$: integral over $f(x)$ is the area under the curve
- sampling of $f(x)$ corresponds to uniform distribution of area

$$\int_{x_{min}}^{x} dx'\, f(x') = R \int_{x_{min}}^{x_{max}} dx' f(x')$$



- if $f(x)$ is friendly, i.e. has an invertible primitive function:

$$F(x) - F(x_{min}) = R\left[F(x_{max}) - F(x_{min})\right]$$

$$\Rightarrow x = F^{-1}\left\{F(x_{min}) + R\left[F(x_{max}) - F(x_{min})\right]\right\}$$

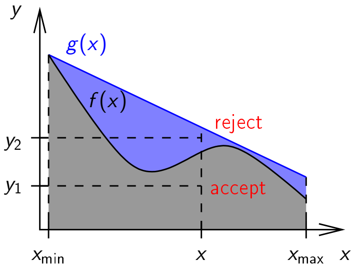
Importance sampling

- find friendly function $g(x) \geq f(x) \; \forall \; x \in [x_{\text{min}}, x_{\text{max}}]$

$$\int \mathrm{d}x \, f(x) = \int \mathrm{d}x \, g(x) \frac{f(x)}{g(x)} = \int \mathrm{d}G(x) \frac{f(x)}{g(x)}$$

Algorithm

1. pick random variable x from $g(x)$
2. pick random variable $y = R \cdot g(x)$
3. if $y > f(x)$ reject x and return to 1, else accept x

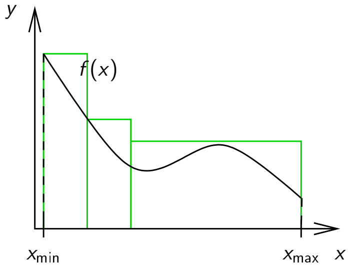


Stratified sampling

- divide integration region in subintervals $[x_i, x_{i+1}]$
- variance is linear: $V(f) = \sum_{i=1}^{m} \frac{x_{i+1} - x_i}{N_i} V_i(f)$
- can lead to reduction of variance when done smartly
- division in equally sized intervals does not increase variance

Algorithm

1. chose subintervals, e.g. such that V_i are similar
2. sample in each subinterval
3. add results with weights $(x_{i+1} - x_i)/N_i$



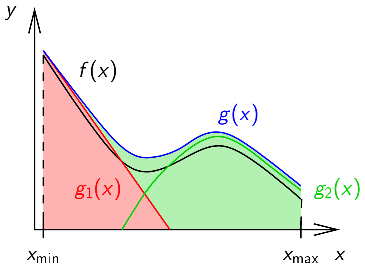
Multichannel sampling

- construct $g(x) \geq f(x)$ as sum of friendly functions

$$g(x) = \sum_i g_i(x) \quad \text{with} \quad A_i = \int_{x_{\min}}^{x_{\max}} \mathrm{d}x \, g_i(x)$$

Algorithm

1. pick $g_i(x)$ with probability A_i/A_{tot}
2. pick random variable x from $g_i(x)$
3. pick $y = R \cdot g(x)$
4. if $y > f(x)$ reject x and return to 1, else accept x



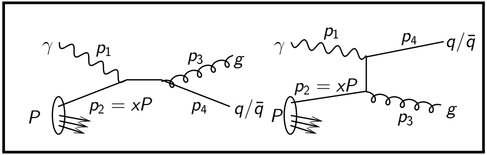
Multidimensional integration

- if boundaries too complicated $\rightarrow$ sample in larger hyper-rectangle and reject points outside integration region
- hit-or-miss and stratified sampling always work
- importance sampling: factorised ansatz
  - $g(\mathbf{x}) = g(x_1, x_2, \ldots, x_n) = g^{(1)}(x_1) g^{(2)}(x_2) \ldots g^{(n)}(x_n)$
    each $g^{(i)}(x_i)$ can again be a sum of $g_j^{(i)}(x_i)$
  - select the x_i independently from $g^{(i)}(x_i)$
  - reject with $f(\mathbf{x})/g(\mathbf{x})$
- importance sampling: nested ansatz
  - if range of x_1 known, that of x_2 depends only on x_1, that of x_3 only on x_1 and x_2 etc., construct $g(x)$ such that
  - x_1 distributed according to $g(x_1) = \int \mathrm{d}x_2 \ldots \mathrm{d}x_n g(\mathbf{x})$
  - x_2 distributed according to $g(x_2; x_1) = \int \mathrm{d}x_3 \ldots \mathrm{d}x_n g(\mathbf{x})$ etc.
  - x_n distributed according to $g(x_n; x_1, \ldots, n_{n-1})$
  - reject with $f(\mathbf{x})/g(\mathbf{x})$

An example: Photoproduction of jets

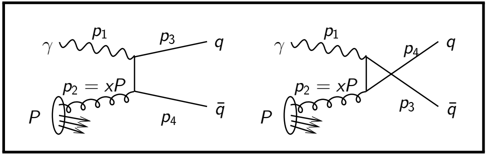
QCD Compton scattering:

$$\frac{\mathrm{d}\hat{\sigma}^{(1)}}{\mathrm{d}\hat{t}} = \frac{8}{3}\frac{\pi\alpha\alpha_s e_q^2}{\hat{s}^2}\left(\frac{-\hat{u}}{\hat{s}} + \frac{\hat{s}}{-\hat{u}}\right)$$



photon-gluon fusion:

$$\frac{\mathrm{d}\hat{\sigma}^{(2)}}{\mathrm{d}\hat{t}} = \frac{\pi\alpha\alpha_s e_q^2}{\hat{s}^2}\left(\frac{\hat{u}}{\hat{t}} + \frac{\hat{t}}{\hat{u}}\right)$$



$$\sigma = \int_{x_{\min}}^{1} \mathrm{d}x \int_{\hat{t}_{\min}(x)}^{\hat{t}_{\max}(x)} \mathrm{d}\hat{t} \sum_{i,k} f_i(x, Q^2)\frac{\mathrm{d}\hat{\sigma}^{(k)}}{\mathrm{d}\hat{t}} = \int_{x_{\min}}^{1} \mathrm{d}x \int_{\hat{t}_{\min}(x)}^{\hat{t}_{\max}(x)} \mathrm{d}\hat{t}\, \mathcal{F}(x, \hat{t})$$

An example: Photoproduction of jets

Construct $g_x(x)$ and $g_t(\hat{t})$ to account for the dominant x and $\hat{t}$ behaviour in $\mathcal{F}(x,\hat{t})$:

$$g_x(x) = \frac{A_x}{\ln 1/x_{\text{min}}}\frac{1}{x} + \frac{B_x}{1-x_{\text{min}}}\frac{1}{x^2} + \frac{C_x 2x_{\text{min}}^2}{1-x_{\text{min}}^2}\frac{1}{x^3}$$

$$\Rightarrow \int_{x_{\text{min}}}^{1} \mathrm{d}x\, g_x(x) = A_x + B_x + C_x$$

$$g_t(\hat{t}) = \frac{A_t}{\hat{t}_{\text{max}} - \hat{t}_{\text{min}}} + \frac{B_t}{\ln \hat{t}_{\text{max}}/\hat{t}_{\text{min}}}\frac{1}{\hat{t}} + \frac{C_t}{\ln \hat{t}_{\text{max}}/\hat{t}_{\text{min}}}\frac{1}{\hat{u}}$$

An example: Photoproduction of jets

Procedure

1. select channel in g_x with relative probability A_x, B_x, C_x
2. generate x from this channel by direct sampling
3. compute $t_{\text{min}}(x)$ and $t_{\text{max}}(x)$
4. select $\hat{t}$ from g_t in the same way

$$\sigma = \int_{x_{\text{min}}}^{1} \mathrm{d}x \int_{\hat{t}_{\text{min}}(x)}^{\hat{t}_{\text{max}}(x)} \mathrm{d}\hat{t}\, \mathcal{F}(x,\hat{t}) = \int_{x_{\text{min}}}^{1} \mathrm{d}x\, g_x(x) \int_{\hat{t}_{\text{min}}(x)}^{\hat{t}_{\text{max}}(x)} \mathrm{d}\hat{t}\, g_t(\hat{t}) \frac{\mathcal{F}(x,\hat{t})}{g_x(x) g_t(\hat{t})}$$

$$= (A_x + B_x + C_x)(A_t + B_t + C_t) \left\langle \frac{\mathcal{F}(x,\hat{t})}{g_x(x) g_t(\hat{t})} \right\rangle$$

Sampling from Poisson distribution - 1

Consider radioactive decay

- nucleus can decay only once → Poisson distribution
- $\Delta(t)$: probability that nucleus has not decayed by time t
- $P(t) = -\mathrm{d}\Delta(t)/\mathrm{d}t$: probability for decay at time t
- must take survival probability into account: $P(t) = c\Delta(t)$
- survival probability $\Delta(t) = ce^{-ct}$
- for radioactive decay c a is constant, now generalise to $P(t) = f(t)\Delta(t)$
- assume $f(t)$ is a friendly function

Sampling from Poisson distribution - 2

$$P(t) = -\frac{\mathrm{d}\Delta(t)}{\mathrm{d}t} = f(t)\Delta(t) \qquad \text{with} \quad f(t) \geq 0$$

Standard solution:

$$\frac{1}{\Delta(t)}\frac{\mathrm{d}\Delta(t)}{\mathrm{d}t} = \frac{\mathrm{d}(\ln\Delta(t))}{\mathrm{d}t} = -f(t)$$

$$\ln\Delta(t) - \ln\Delta(0) = -\int_0^t \mathrm{d}t'\, f(t') \quad \Rightarrow \quad \Delta(t) = \exp\left(-\int_0^t \mathrm{d}t'\, f(t')\right)$$

$$\Delta(t) = \exp\left[-(F(t) - F(0))\right]$$

Since $\Delta(0) = 1$ and $\Delta(\infty) = 0$: $\Delta(t) = R$

$$t = F^{-1}\left[F(0) - \ln R\right]$$

The veto algorithm

Problem: $f(t)$ may not be a friendly function

Solution: find friendly $g(t) \geq f(t)$ and use veto algorithm

Veto algorithm

1. set $i = 0$ and $t_0 = 0$
2. set $i = i + 1$
3. pick $t_i = G^{-1}\left[G(t_{i-1}) - \ln R\right]$, i.e. $t_i > t_{i-1}$
4. pick $y = R g(t_i)$
5. if $y > f(t_i)$ reject and return to 2, else accept $t = t_i$

The veto algorithm: proof - 1

Now: $\Delta(t_a, t_b) = \exp\left(-\int_{t_a}^{t_b} \mathrm{d}t'\, g(t')\right)$

$P_n(t)$: probability to accept t after n rejections

$$P_0(t) = P(t = t_1) = g(t)\Delta(0,t)\frac{f(t)}{g(t)} = f(t)\Delta(0,t)$$

The veto algorithm: proof - 1

Now: $\Delta(t_a, t_b) = \exp\left(-\int_{t_a}^{t_b} \mathrm{d}t'\, g(t')\right)$

$P_n(t)$: probability to accept t after n rejections

$$P_0(t) = P(t = t_1) = g(t)\Delta(0,t)\frac{f(t)}{g(t)} = f(t)\Delta(0,t)$$

$$P_1(t) = P(t = t_2) = \int_0^t \mathrm{d}t_1\, g(t_1)\Delta(0,t_1)\left(1 - \frac{f(t)}{g(t)}\right)g(t)\Delta(t_1,t)\frac{f(t)}{g(t)}$$

$$= f(t)\Delta(0,t)\int_0^t \mathrm{d}t_1\, [g(t_1) - f(t_1)] = P_0(t) I_{g-f}$$

The veto algorithm: proof - 1

Now: $\Delta(t_a, t_b) = \exp\left(-\int_{t_a}^{t_b} \mathrm{d}t'\, g(t')\right)$

$P_n(t)$: probability to accept t after n rejections

$$P_0(t) = P(t = t_1) = g(t)\Delta(0,t)\frac{f(t)}{g(t)} = f(t)\Delta(0,t)$$

$$P_1(t) = f(t)\Delta(0,t)\int_0^t \mathrm{d}t_1\ [g(t_1) - f(t_1)] = P_0(t) I_{g-f}$$

$$P_2(t) = P_0(t)\int_0^t \mathrm{d}t_1\ [g(t_1) - f(t_1)]\int_{t_1}^t \mathrm{d}t_2\ [g(t_2) - f(t_2)]$$

$$= P_0(t)\int_0^t \mathrm{d}t_1\ [g(t_1) - f(t_1)]\int_0^t \mathrm{d}t_2\ [g(t_2) - f(t_2)]\,\theta(t_2 - t_1)$$

$$= P_0(t)\frac{1}{2}\left(\int_0^t \mathrm{d}t_1\ [g(t_1) - f(t_1)]\right)^2 = P_0\frac{1}{2}I_{g-f}^2$$

The veto algorithm: proof - 1

Now: $\Delta(t_a, t_b) = \exp\left(-\int_{t_a}^{t_b} \mathrm{d}t'\, g(t')\right)$

$P_n(t)$: probability to accept t after n rejections

$$P_0(t) = P(t = t_1) = g(t)\Delta(0,t)\frac{f(t)}{g(t)} = f(t)\Delta(0,t)$$

$$P_1(t) = f(t)\Delta(0,t)\int_0^t \mathrm{d}t_1\,\left[g(t_1) - f(t_1)\right] = P_0(t) I_{g-f}$$

$$P_2(t) = P_0(t)\frac{1}{2}\left(\int_0^t \mathrm{d}t_1\,\left[g(t_1) - f(t_1)\right]\right)^2 = P_0 \frac{1}{2} I_{g-f}^2$$

$$P_n(t) = P_0 \frac{1}{n!} I_{g-f}^n$$

The veto algorithm: proof - 2

$$P(t) = \sum_{n=0}^{\infty} P_n(t) = P_0(t) \sum_{n=0}^{\infty} \frac{1}{n!} I_{g-f}^n = P_0(t) \exp(I_{g-f})$$

$$= f(t) \exp\left(-\int_0^t \mathrm{d}t'\, g(t')\right) \exp\left(\int_0^t \mathrm{d}t'\, [g(t') - f(t')]\right)$$

$$= f(t) \exp\left(-\int_0^t \mathrm{d}t'\, f(t')\right)$$

True random numbers

- (true) random numbers are uncorrelated and unpredictable
- can only be obtained from observing a physical process

  radioactive decay, electronic noise, . . .
- hard to construct a device that is accurate, unbiased and fast
- reading in stored random numbers not feasible

  MC computations tend to use too many random numbers
- debugging MC code with true random numbers very difficult

Pseudo-random numbers

- pseudo-random numbers are generated by algorithm
- predictable, but uncorrelated(?)
- generation is fast
- can reduce variance of difference between runs by using same (pseudo-)random number sequence
- finite period: when a number is generated for the second time, the sequence will repeat itself
- good mathematical understanding of some generators
- multiplicative congruential generators: $r_i = (ar_{i-1}) \mod m$
- mixed congruential generators: $r_i = (ar_{i-1} + b) \mod m$
- obvious choice: $m = 2^t$ where t number of bits in representation of integer

Pseudo-random numbers: Marsaglia effect

- successive d-tuples of pseudo-random numbers fall on finite number of parallel hyperplanes in d-dimensional space

G. Marsaglia, *Proc. Nat. Acad. Sci.* 61 (1968) 25-8

$\Rightarrow$ pseudo-randoms are always correlated

- irrelevant in practice, if number of hyperplanes large enough
- number of hyperplanes: $(d!2^t)^{1/d}$
- compound multiplicative congruential generator:
  $r_i = (ar_{i-1} + br_{i-2}) \mod m$
  increases number of hyperplanes by factor $2^{t/d}$

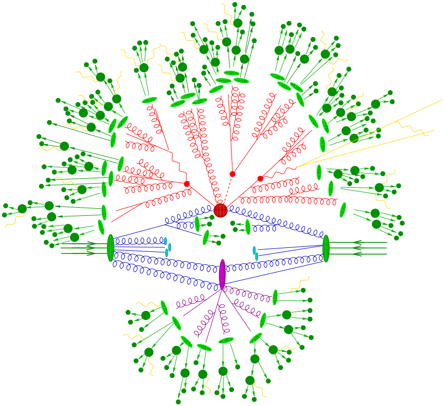
for clever choice of a and b

H. J. Ahrends and U. Dieter (1979)

Outline

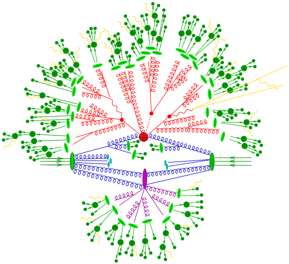
Overview: Multi purpose event generators

$$d\sigma_{\text{final state}} = d\sigma_{\text{hard process}} \mathcal{P}_{\text{QCD rad.}} \mathcal{P}_{\text{hadronisation}} \mathcal{P}_{\text{decays}} \mathcal{P}_{\text{QED rad.}} \mathcal{P}_{\text{MPI}}$$



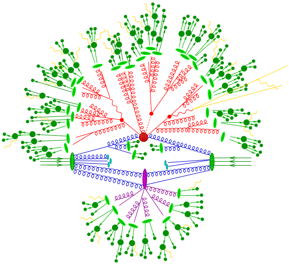
- integrates cross section
- generates events: sets of particles distributed according to $d\sigma_{\text{final state}}$
- can calculate any observable
  no new calculation for new observable
- relies on separation of scales
- multi-purpose generators: HERWIG , PYTHIA, SHERPA

Hard process



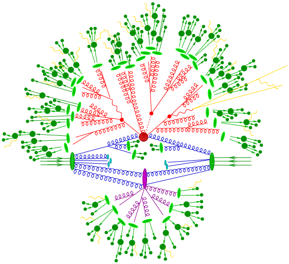
- hard scattering matrix elements
- calculated at fixed order in perturbation theory
- low multiplicity MEs can be hard-coded
- for intermediate and high multiplicity automatic ME generators are needed
- in event generators LO and NLO MEs available
- multi-purpose event generators interface MEs from dedicated generators

Parton showers



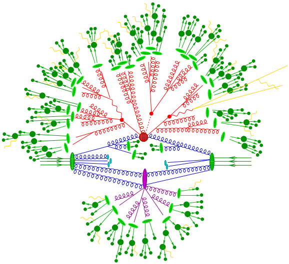
- radiative corrections in QCD $\rightarrow$ QCD bremsstrahlung
- initial and final state parton shower
- explicit DGLAP evolution
- resummation of collinear logs in QCD
- perturbative calculation, but not fixed order
- leading log (LL) accuracy with some sub-leading (NLL) pieces

Hadronisation



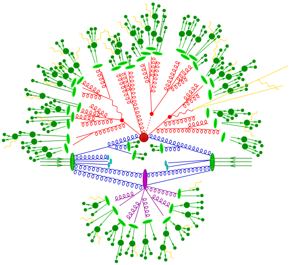
- conversion of partons into hadrons
- non-perturbative long-distance physics
- phenomenological models
- have to be tuned to data
- process independent by factorisation arguments

Decays



- hadron decays
- τ decays
- EM, weak & strong decays
- weak neutral meson mixing
- many-body decays
- polarisation, angular correlations
- tables of decay channels

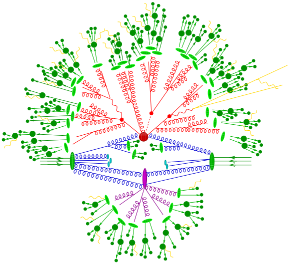
QED radiation



- collinear resummation $\rightarrow$ DGLAP
- resummation of soft photons à la YFS
  - resummation of soft-photon logs in massive Abelian gauge theories
  - collinear logs can be added order by order, but not resummed
  - no ordering of emissions
  - coherent radiation off charged multipole
- simultaneous QCD & QED DGLAP evolution
- cannot combine QCD DGLAP & YFS
- apply YFS to non-QCD final state

no photon radiation off quarks

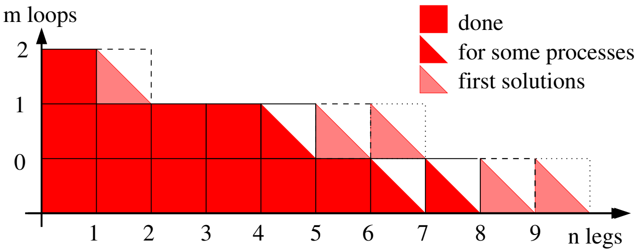
Multiple parton interactions



- more than one parton-parton interaction per proton-proton collision
- gives rise to additional activity
  $\rightarrow$ underlying event
- beyond factorisation theorems
  $\rightarrow$ need to model
- related to minimum bias physics, i.e. reactions without hard scattering

Outline

Availability of matrix elements



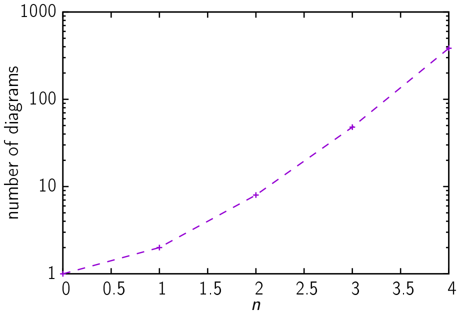
- tree level matrix elements highly automated
- one-loop case is getting there
- for now stick to tree level

Leading order matrix elements: a fundamental problem

Factorial growth of number of diagrams:

simple process: $e^+ + e^- \to q + \bar{q} + ng$

n	# diagrams
0	1
1	2
2	8
3	48
4	384



Leading order matrix elements: naive approach

Textbook method:

- calculate amplitudes using Feynman diagrams
- square using completeness relations
- sum/average over external states (helicity and colour)
- proliferation of interference terms
- computational effort grows quadratically with # diagrams

Improvement:

- remember: amplitudes are complex numbers
- add them before squaring
- computational effort grows linearly with number of diagrams

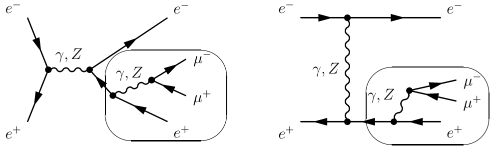
Leading order matrix elements: helicity amplitudes

Idea:

- introduce helicity spinors
- write everything as spinor products, e.g. $\bar{u}(p_1, h_1)u(p_2, h_2) \in \mathbb{C}$
- translate Feynman diagram into helicity amplitudes: complex valued functions of momenta and helicities

Improvement – taming the factorial growth:

- many Feynman diagrams share sub-graphs

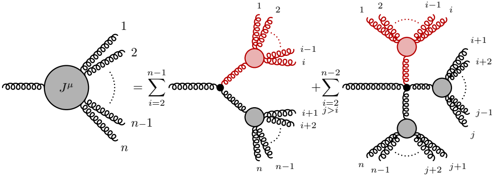


$\Rightarrow$ book-keep sub-amplitudes and reuse

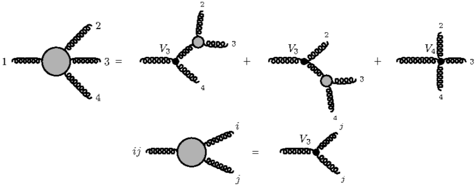
Leading order matrix elements: recursion relations

Berends-Giele recursion relations: Berends, Giele NPB306(1988)759

- construct amplitude recursively:



- a simple example:

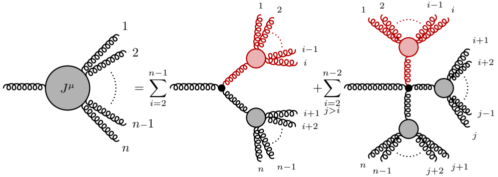


Leading order matrix elements: recursion relations

Berends-Giele recursion relations:

Berends, Giele NPB306(1988)759

- construct amplitude recursively:



Further improvements:

- sampling over colours — amplitudes can be stripped of colour factors

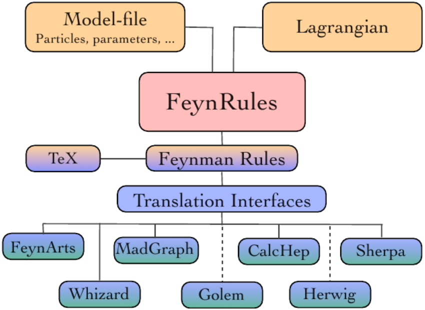
  Maltoni, Paul, Stelzer, Willenbrock, Phys. Rev. D **67** (2003) 014026

- colour dressing — evaluate colour dynamically at each vertex

  Duhr, Hoeche, Maltoni, JHEP **0608** (2006) 062

New physics models – FeynRules

- most ME generators suited for any physics model, but implementing Feynman rules tedious and error-prone
- automated by FeynRules
- extracts vertices from Lagrangian based on minimal information about particle content



Christensen, Duhr, Comput. Phys. Commun. 180 (2009) 1614

Phase space

$$d\Phi_N = \left[\prod_{i=1}^{N} \frac{dp_i^4}{(2\pi)^4}\delta(p_i^2 - m_i^2)\theta(E_i)\right](2\pi)^4\delta^{(4)}\left(p_a + p_b - \sum_{i=1}^{N} p_i\right)$$
$$= \left[\prod_{i=1}^{N} \frac{d^3p_i}{(2\pi)^3 2E_i}\right](2\pi)^4\delta^{(4)}\left(p_a + p_b - \sum_{i=1}^{N} p_i\right)$$

1-particle phase space:

$$d\Phi_1 = \frac{d^3p}{(2\pi)^3 2E} = \frac{p\,dE\,d\Omega}{16\pi^3} = \frac{d^2p_\perp dy}{16\pi^3}$$

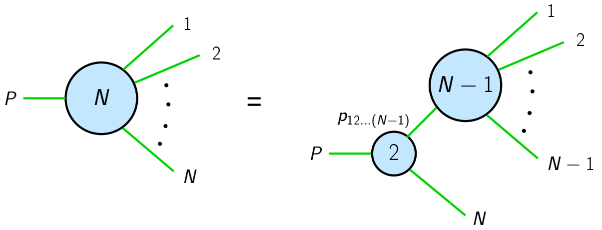
2-particle phase space (evaluated in rest frame):

$$d\Phi_2 = \frac{1}{8\pi}\frac{2p}{E_{cm}}\frac{d\Omega}{4\pi}$$

Recursive phase space

Introduce intermediate state $p_{12\ldots(N-1)} = \sum_{i=1}^{N-1} p_i$

$$\mathrm{d}\Phi_N(P; p_1, \ldots, p_N) = \mathrm{d}m^2_{12\ldots(N-1)} \mathrm{d}\Phi_2(P; p_{12\ldots(N-1)}, p_N) \times \mathrm{d}\Phi_{N-1}(p_{12\ldots(N-1)}; p_1 \ldots, p_{N-1})$$



Uniform phase space (RAMBO/MAMBO)

Kleiss, Stirling, Ellis, Comput. Phys. Commun. 40 (1986) 359

- consider N massless particles without 4-momentum conservation

$$R_N = \int \prod_{i=1}^{N} \mathrm{d}^4 q_i \delta(q_i^2)\theta(q_i^0) f(q_i^0) = \left[2\pi \int_0^\infty \mathrm{d}q_i^0 \, q_i^0 f(q_i^0) \right]^N$$

$f(q_i^0)$: weight function to keep phase space volume finite

- boost to overall rest frame
- recsale by common factor to reach desired mass
- for $f(q_i^0) = e^{-q_i^0}$ uniform phase space distribution

Advanced methods

- follow QCD antenna pattern (HAAG/Sarge)

  van Hameren, Papadopoulos, Eur. Phys. J. C 25 (2002) 563
- multi-channeling: each Feynman diagram related to a phase space mapping ("channel"), optimise relative weights

  Kleiss, Pittau, Comput. Phys. Commun. 83 (1994) 141
- improve by building channels recursively
- for best efficiency integrate phase space generation with matrix element generator

Outline

Summary - 1

Monte Carlo integration

- Monte-Carlo method of choice for multidimensional integration
- side-product: events with a statistical interpretation that can be projected onto arbitrary observables
- discussed variance reducing techniques
- for temporal problems: sampling from Poisson distribution

Summary - 2

Monte Carlo event generators

- Monte Carlo event generators provide theoretical description of high energy scattering as close to nature as possible
- multi-component description relying on separation of scales
- some parts faithful representation of perturbation theory

  matrix elements, parton showers
- some parts phenomenological models of non-perturbative aspects

  e.g. hadronisation

Summary - 3

Matrix elements and phase space

- tree level matrix elements fully automatised
- fight factorial growth with diagrammatic or recursive techniques
- build phase space recursively
- multi-channeling: one channel per diagram