Received: *June 15, 2001*, Accepted: *August 10, 2001*
Revised: *July 23, 2001*

A general algorithm to generate unweighted events for next-to-leading order calculations in electron-positron annihilation

Stefan Weinzierl

Dipartimento di Fisica, Università di Parma
INFN Gruppo Collegato di Parma
43100 Parma, Italy
E-mail: stefanw@fis.unipr.it

Abstract: Given a next-to-leading order calculation, we show how to set up a computer program, which generates a sequence of unweighted momentum configurations, each configuration containing either n or $n+1$ four-vectors, such that for any infrared safe observable the average over these configurations coincides with the NLO calculation up to errors of order y_{res}. The core of the algorithm is a method to combine real emission and virtual corrections on a point-by-point basis in hard phase space. The algorithm can be implemented on top of existing NLO calculations.

Keywords: QCD, NLO Computations, Jets.

JHEP08(2001)028

Contents

1. Introduction

As high energy experiments move towards a new generation of colliders, accurate simulation of collision events becomes increasingly important. Among the main tools used today are next-to-leading order (NLO) calculations and event generators. Up to now these two tools have a complementary regions of applicability. A NLO-calculation for an n-jet observable calculates this observable as a sum of two terms, one term living on the n-parton phase space, the other on the $(n+1)$-parton phase space. These two contributions correspond roughly to the virtual corrections and the real emissions, where infrared divergences have been taken care of by a method like phase space slicing or the subtraction method. Usually the two contributions to an observable are only combined in the end. As a side-effect, one deals at intermediate steps with variable and/or negative weights.

Event generators like HERWIG [1] and PYTHIA [2] on the other hand produce unweighted events through a three-stage process: from an initial hard process parton showers are generated. A hadronization model converts at a low scale the partons into hadrons. The hard process is usually based on Born matrix elements.

The two approaches have complementary strenght: parton showers describe quite reasonably multiple collinear emission, whereas a NLO-calculation is by far more accurate when n partons are well separated in phase space.

It is a topic of current interest to merge the benefits of the two approaches into an "event generator accurate at NLO". Friberg and Sjöstrand have suggested a possible road in [3]. Similar approaches have been considered by Collins [4], Pötter [5] and Dobbs and Lefebvre [6] and applied to deep-inelastic scattering ([7, 8]) and the production of electroweak gauge bosons in hadron-hadron collisions ([9, 10]). Mrenna has considered the case how to combine resummed calculations with parton showers [11]. Despite this progress, there is still room for improvement: the algorithm of Collins may generate events with negative weight. The algorithm of Pötter uses phase space slicing and introduces a unresolved region, such that the sum of the Born term, the virtual corrections and the real emission part is exactly zero [12]. It is not guaranteed that this region is always small enough such that the approximations of phase space slicing are still valid.

In combining parton showers with NLO-calculations two problems can immediately be identified: negative weights and double counting. In this paper we would like to address the first problem. As already stated before, a specific momentum configuration within a NLO calculation might come with a negative weight. There is nothing wrong with this fact by itself and one might think of carrying this weight along the further simulation of the event. However, there are some concerns raised against this approach:

a) Some hadronization models have problems with negative weights [3].

b) Experimentalists are unwilling to accept negative weights. Event reconstruction and detector simulation are quite expensive in terms of computer time and an event with a negative weight would "annihilate" parts of the CPU-time already spent.

Therefore it is desirable to avoid negative weights from the very beginning. In view of point b) this is more a question of efficiency than of principle.

In this paper we consider an "NLO partonic event generator". Given a next-to-leading order calculation, we show how to set up a computer program, which generates a sequence of unweighted momentum configurations, each configuration containing either n or $n+1$ four-vectors, such that for any infrared safe observable the average over these configurations coincides with the NLO calculation up to errors of a resolution variable y_{res}. The basic idea is simple: consider a $(n+1)$-parton configuration. If all partons are separated by at least y_{res}, we return this $(n+1)$-parton configuration. If on the other hand a parton is not resolved at a scale y_{res}, we average over the real emissions up to the scale y_{res}, combine the result with the virtual corrections and return a n-parton configuration. In this paper we show that this can be done in a general and effective way: our procedure does not depend on the specific hard process and the integration over the real emission part involves only a three-dimensional integration. These two points are important: today many NLO-calculations already exist and we would like to be able to use them without major modifications. Our algorithm can take as an input subroutines, which correspond to the matrix elements for real emission and virtual corrections without modifications. Only the subtraction terms within the dipole formalism have to be implemented. Secondly, if a parton is not resolved we have to integrate out this contribution. An important ingredient is a factorization of the phase space into a "hard" n-parton configuration and the insertion of a "soft" particle into this "hard" configuration. This allows us to integrate out the unresolved region on a point-by-point bases in hard phase space. We discuss explicitly the case of massless QCD processes in electron-positron annihilation, although the extension to initial-state partons and/or massive partons is in principle feasible.

In this paper we do not consider how to attach parton showers to the events nor do we address the problem of double counting. This is a separate problem and addressed for example in [3, 4].

This paper is organized as follows: the next section gives a more detailed outline of the algorithm. Section 3 reviews the structure of infrared divergences and the dipole formalism. Section 4 deals with basic Monte Carlo techniques and the generation of phase space. In section 5 we present an algorithm to generate unweighted events according to leading-order matrix elements. In section 6 this algorithm is generalized to unweighted events according to next-to-leading order matrix elements. Finally, section 7 contains the conclusions.

2. Preliminaries

We assume that we have at our disposal a NLO-calculation, consisting of the virtual corrections, symbolically denoted by

$$d\sigma^V = \left\langle \mathcal{M}^{(n)}_{\text{Born}} | \mathcal{M}^{(n)}_{1-\text{loop}} \right\rangle + \left\langle \mathcal{M}^{(n)}_{1-\text{loop}} | \mathcal{M}^{(n)}_{\text{Born}} \right\rangle, \tag{2.1}$$

the Born amplitude for the emission of an additional parton

$$d\sigma^R = \left\langle \mathcal{M}^{(n+1)}_{\text{Born}} | \mathcal{M}^{(n+1)}_{\text{Born}} \right\rangle, \tag{2.2}$$

as well as colour-ordered partial amplitudes for the Born process. The partial amplitudes are needed for the dipole formalism. We denote the Born matrix element by

$$d\sigma^B = \left\langle \mathcal{M}^{(n)}_{\text{Born}} | \mathcal{M}^{(n)}_{\text{Born}} \right\rangle. \tag{2.3}$$

We distinguish three scales:

- The scale y_{cut}: this scale enters directly the definition of a physical observable. An example would be the measurement of the 3-jet cross section in electron-positron annihilation with the Durham algorithm at $y_{\mathrm{cut}} = 0.01$. Results of physical observables may depend on the scale y_{cut}.

- The scale y_{res}: the resolution scale y_{res} gives the lower bound to which we might reasonably push y_{cut}. On the experimental side y_{res} is set by the finite resolution of the detector. We are here concerned with fixed-order perturbation theory. Consider first the situation where we would like to generate a sequence of n four-vectors according to the leading-order matrix element. This matrix element is divergent when one parton becomes either soft or collinear. We would therefore generate predominately events where one parton is soft or collinear. For any infrared-safe n-jet observable these events will be rejected since they do not pass the jet-defining cuts. This is inefficient. We therefore introduce a resolution function $\Theta_{y_{\mathrm{res}}}$ and generate only events which are at least resolved at the scale y_{res}. Let us now consider next-to-leading order calculations. With multiple scales in the problem, this will inevitably lead to logarithm of ratios of these scales. If the various scales are not of the same magnitude, these logarithm can become large and spoil the validity of a fixed-order perturbative expansion. Consider the following situation in the real emission part of an NLO-calculation: assume that we have n partons well separated in phase space with an additional parton close to one of the first n partons. We would like to replace this $(n+1)$-parton configuration with an n-parton configuration, where the unresolved parton has been integrated out. Integrating out the unresolved parton in the real emission part up to a resolution scale y_{res} will yield a positive, but divergent contribution. The divergent part is compensated by the virtual corrections, and the sum of the real emission part and the virtual corrections is finite, but not necesarrily positive. The sign will depend on the scale y_{res}. If y_{res} is chosen too small, we do not include enough contributions from the real emission amplitude and the combined result will be negative. This is unacceptable for a probabilistic interpretation. We therefore have too chose y_{res} large enough that for a given "hard" n-parton configuration we can integrate out an additional "soft" parton, which is within y_{res} of one of the "hard" partons and the combination of the obtained real emission part with the virtual corrections gives a positive contribution.

- The scale y_{min}: this is an internal technical parameter we introduce for integrating out efficiently unresolved contributions. In some place we will neglect contributions of order y_{min}, but since $y_{\mathrm{min}} \ll y_{\mathrm{res}}$ the introduced systematic error is negligible against the approximations made in the previous step.

Of course the technical problem is to integrate out efficiently the unresolved region for a given "hard" n-parton configuration. We show that this involves only a three-dimensional integration. We take special care that this can be done efficiently: our algorithm is based on the subtraction method. This is more efficient than the phase space slicing approach, since we do not have to reproduce numerically a logarithm of $y_{\min}$, which cancels in the end. Furthermore we use a specific remapping of the unresolved phase space, which tends to flatten the integrand.

Our method relies on the following ingredients:

- The structure of the infrared divergences is universal. We may therefore add and subtract an appropriately chosen term, such that the integral over the real emission part in the unresolved region and the virtual corrections become finite. Explicitly we are using the dipole formalism of Catani and Seymour [13] together with slight modifications given by Nagy and Trócsányi [14].

- An algorithm to generate the "hard" n-parton phase space from a set of random numbers distributed in $[0, 1]$. There are several algorithms on the market and we use the RAMBO algorithm by Kleiss, Stirling and Ellis [15] for this purpose.

- An algorithm to insert an additional "soft" particle into a given "hard" configuration. This algorithm is originally due to Kosower and has been implemented into the NLO 4-jet program MERCUTIO [16].

- An algorithm to generate a sequence of events according to a multi-dimensional probability density. We use the Metropolis algorithm for that [17].

In the next two sections we will briefly review these tools.

3. Review of the structure of infrared divergences

In this section we briefly review the dipole formalism and the factorization properties of QCD amplitudes in the soft and collinear limit. We follow the notation of Catani and Seymour [13].

The NLO cross section is given by

$$\sigma^{\mathrm{NLO}} = \int_n d\sigma^B + \int_n d\sigma^V + \int_{n+1} d\sigma^R . \tag{3.1}$$

$d\sigma^B$ represents the LO-contribution, $d\sigma^V$ the virtual corrections and $d\sigma^R$ the real emission part. Taken separately, both $d\sigma^V$ and $d\sigma^R$ are divergent, only their sum is finite. Within the subtraction method, one adds and subtracts a suitable chosen term. The NLO cross section is therefore rewritten as

$$\sigma^{\mathrm{NLO}} = \int_n d\sigma^B + \int_n \left(d\sigma^V + \int_1 d\sigma^A \right) + \int_{n+1} \left(d\sigma^R - d\sigma^A \right) . \tag{3.2}$$

The approximation $d\sigma^A$ has to fulfill the following requirements:

- $d\sigma^A$ must be a proper approximation of $d\sigma^R$ such as to have the same pointwise singular behaviour (in D dimensions) as $d\sigma^R$ itself. Thus, $d\sigma^A$ acts as a local counterterm for $d\sigma^R$ and one can safely perform the limit $\varepsilon \to 0$. This defines a cross-section contribution

$$\sigma^{\mathrm{NLO}}_{\{n+1\}} = \int_{n+1} \left(d\sigma^R\big|_{\varepsilon=0} - d\sigma^A\big|_{\varepsilon=0} \right). \tag{3.3}$$

- Analytic integrability (in D dimensions) over the one-parton subspace leading to soft and collinear divergences. This gives the contribution

$$\sigma^{\mathrm{NLO}}_{\{n\}} = \int_n \left(d\sigma^V + \int_1 d\sigma^A \right)_{\varepsilon=0}. \tag{3.4}$$

The final structure of an NLO calculation is then

$$\sigma^{\mathrm{NLO}} = \sigma^B + \sigma^{\mathrm{NLO}}_{\{n\}} + \sigma^{\mathrm{NLO}}_{\{n+1\}}. \tag{3.5}$$

All contributions on the r.h.s. of eq. (3.5) are now finite. However, it should be noted that $\sigma^{\mathrm{NLO}}_{\{n+1\}}$ is a difference of two terms and therefore not guaranteed to be positive-definite. Similar, $\sigma^{\mathrm{NLO}}_{\{n\}}$ contains the interference term between the one-loop amplitude and the Born amplitude and is therefore also not guaranteed to be positive-definite.

The $(n+1)$ matrix element is approximated by a sum of dipole terms

$$\sum_{\text{pairs } i,j} \sum_{k \neq i,j} \mathcal{D}_{ij,k} = \sum_{\text{pairs } i,j} \sum_{k \neq i,j} -\frac{1}{2p_i \cdot p_j} \left\langle 1, \ldots, (\tilde{ij}), \ldots, \tilde{k}, \ldots \right| \frac{\mathbf{T}_k \cdot \mathbf{T}_{ij}}{\mathbf{T}^2_{ij}} V_{ij,k} \times \\ \times \left| 1, \ldots, (\tilde{ij}), \ldots, \tilde{k}, \ldots \right\rangle,$$

where the emitter parton is denoted by $\tilde{ij}$ and the spectator by $\tilde{k}$. Here $\mathbf{T}_i$ denotes the colour charge operator for parton i. The colour charge operators $\mathbf{T}_i$ for a quark, gluon and antiquark in the final state are

$$\begin{aligned} \text{quark}: &\ \langle \ldots q_i \ldots | T^a_{ij} | \ldots q_j \ldots \rangle, \\ \text{gluon}: &\ \langle \ldots g^c \ldots | i f^{cab} | \ldots g^b \ldots \rangle, \\ \text{antiquark}: &\ \langle \ldots \bar{q}_i \ldots | \left(-T^a_{ji} \right) | \ldots \bar{q}_j \ldots \rangle. \end{aligned} \tag{3.6}$$

$V_{ij,k}$ is a matrix in the helicity space of the emitter with the correct soft and collinear behaviour. $|1, \ldots, (\tilde{ij}), \ldots, \tilde{k}, \ldots\rangle$ is a vector in colour- and helicity space. It is also convenient to introduce the variable

$$y_{ij,k} = \frac{2p_i p_j}{2p_i p_j + 2p_i p_k + 2p_j p_k} \tag{3.7}$$

and the ordered set $\mathcal{S}$ consisting of all relevant $y_{ij,k}$-variables:

$$\mathcal{S} = \{ y_{ij,k} | i \text{ emitter};\ j \text{ emitted particle};\ k \text{ spectator} \}. \tag{3.8}$$

We denote the number of elements of $\mathcal{S}$ by m. Explicit formulae for the dipole terms $\mathcal{D}_{ij,k}$ have been given by Catani and Seymour [13]. In the CDR-scheme they read[1]

$$\begin{aligned}
V_{q_i g_j,k} &= 8\pi\mu^{2\varepsilon}\alpha_s \delta_{ss'} C_F \left[\frac{2}{1-\tilde{z}_i(1-y_{ij,k})} - (1+\tilde{z}_i) - \varepsilon(1-\tilde{z}_i) \right], \\
V_{q_i \bar{q}_j,k} &= 8\pi\mu^{2\varepsilon}\alpha_s T_R \left[-g^{\mu\nu} - \frac{2}{p_i p_j}(\tilde{z}_i p_i^\mu - \tilde{z}_j p_j^\mu)(\tilde{z}_i p_i^\nu - \tilde{z}_j p_j^\nu) \right], \\
V_{g_i g_j,k} &= 8\pi\mu^{2\varepsilon}\alpha_s 2C_A \left[-g^{\mu\nu} \left(\frac{1}{1-\tilde{z}_i(1-y_{ij,k})} + \frac{1}{1-\tilde{z}_j(1-y_{ij,k})} - 2 \right) + \right. \\
&\qquad \left. + (1-\varepsilon)\frac{1}{p_i p_j}(\tilde{z}_i p_i^\mu - \tilde{z}_j p_j^\mu)(\tilde{z}_i p_i^\nu - \tilde{z}_j p_j^\nu) \right], \qquad (3.9)
\end{aligned}$$

where s, s' are the spin indices of the fermion $\tilde{ij}$ and μ, ν are the spin indices of the gluon $\tilde{ij}$. The momenta of the emitter and the spectator are related to the original $(n+1)$-parton configuration by

$$\begin{aligned}
\tilde{p}_k^\mu &= \frac{1}{1-y_{ij,k}} p_k^\mu, \\
\tilde{p}_{ij}^\mu &= p_i^\mu + p_j^\mu - \frac{y_{ij,k}}{1-y_{ij,k}} p_k^\mu. \qquad (3.10)
\end{aligned}$$

The variables $\tilde{z}_i$ and $\tilde{z}_j$ are given by

$$\tilde{z}_i = \frac{p_i p_k}{p_j p_k + p_i p_k}, \qquad \tilde{z}_j = 1 - \tilde{z}_i. \qquad (3.11)$$

We subtract the approximation only for $y_{ij,k} < y_{\mathrm{res}}$. Therefore

$$\begin{aligned}
\int d\sigma^R - d\sigma^A = &\int d\phi_{n+1} \left|\mathcal{M}(p_1,\ldots,p_{n+1})\right|^2 \Theta(\text{all } y_{ij,k} > y_{\mathrm{res}}) + \\
&+ \int d\phi_{n+1} \left[\left|\mathcal{M}(p_1,\ldots,p_{n+1})\right|^2 (1 - \Theta(\text{all } y_{ij,k} > y_{\mathrm{res}})) \times \right. \qquad (3.12) \\
&\quad \times \Theta^{\mathrm{cut}}_{n+1}(p_1,\ldots,p_{n+1}) - \sum_{\text{pairs } i,j} \sum_{k \neq i,j} \mathcal{D}_{ij,k}(p_1,\ldots,p_{n+1}) \times \\
&\quad \left. \times \Theta(y_{ij,k} < y_{\mathrm{res}}) \Theta^{\mathrm{cut}}_n(p_1,\ldots,\tilde{p}_{ij},\ldots,\tilde{p}_k,\ldots,p_{n+1}) \right].
\end{aligned}$$

Here we split the contribution from $d\sigma^R$ into a "resolved" region (first term) and an "unresolved" region (second term). The first term is infrared finite and positive definite. The sum of the second and third term is by construction infrared finite, but not necessarily positive definite. Both $d\sigma^R$ and $d\sigma^A$ are integrated over the same

[1] Corresponding formulae also exist for the four-dimensional scheme as well as conversion rules between the two schemes [18].

$(n+1)$ parton phase space, but it should be noted that $d\sigma^R$ is proportional to $\Theta^{\text{cut}}_{n+1}$, whereas $d\sigma^A$ is proportional to Θ^{cut}_{n}. Here Θ^{cut}_{n} denotes the jet-defining function for n-partons.

The subtraction term can be integrated over the one-parton phase space to yield the term

$$\mathbf{I} \otimes d\sigma^B = \int_1 d\sigma^A \Theta(y_{ij,k} < y_{\text{res}}) = \sum_{\text{pairs } i,j} \sum_{k \neq i,j} \int d\phi_{dipole} \mathcal{D}_{ij,k} \Theta(y_{ij,k} < y_{\text{res}}) \,. \tag{3.13}$$

The universal factor $\mathbf{I}$ still contains colour correlations, but does not depend on the unresolved parton j. The explicit results corresponding to the restricted region of integration $y_{ij,k} < y_{\text{res}}$ have been given by Nagy and Trócsányi [14]:

$$\begin{aligned} \int d\phi_{\text{dipole}} \mathcal{D}_{ij,k} \Theta(y_{ij,k} < y_{\text{res}}) = -\frac{\alpha_s}{2\pi} \frac{1}{\Gamma(1-\varepsilon)} \left(\frac{4\pi\mu^2}{2\tilde{p}_{ij}\tilde{p}_k} \right)^{\varepsilon} \mathcal{V}_{ij}(\varepsilon, y_{\text{res}}) \times \\ \times \left\langle \mathcal{M}^{(n)}_{\text{Born}} \left| \frac{\mathbf{T}_{ij} \cdot \mathbf{T}_k}{\mathbf{T}^2_{(ij)k}} \right| \mathcal{M}^{(n)}_{\text{Born}} \right\rangle \end{aligned} \tag{3.14}$$

with

$$\begin{aligned} \mathcal{V}_{qg}(\varepsilon, y_{\text{res}}) &= C_F \left\{ \left(\frac{1}{\varepsilon^2} - \ln^2 y_{\text{res}} \right) + \frac{3}{2} \left(\frac{1}{\varepsilon} - 1 + y_{\text{res}} - \ln y_{\text{res}} \right) + 5 - \frac{\pi^2}{2} + O(\varepsilon) \right\}, \\ \mathcal{V}_{gg}(\varepsilon, y_{\text{res}}) &= 2C_A \left\{ \left(\frac{1}{\varepsilon^2} - \ln^2 y_{\text{res}} \right) + \frac{11}{6} \left(\frac{1}{\varepsilon} - 1 + y_{\text{res}} - \ln y_{\text{res}} \right) + \frac{50}{9} - \frac{\pi^2}{2} + \right. \\ &\qquad \left. + O(\varepsilon) \right\}, \\ \mathcal{V}_{q\bar{q}}(\varepsilon, y_{\text{res}}) &= T_R \left\{ -\frac{2}{3} \left(\frac{1}{\varepsilon} - 1 + y_{\text{res}} - \ln y_{\text{res}} \right) - \frac{16}{9} + O(\varepsilon) \right\}. \end{aligned} \tag{3.15}$$

The term $\mathbf{I} \otimes d\sigma^B$ lives on the phase space of the n-parton configuration and has the appropriate singularity structure to cancel the infrared divergences coming from the one-loop amplitude. Therefore

$$d\sigma^V + \mathbf{I} \otimes d\sigma^B \tag{3.16}$$

is infrared finite.

We now introduce a function $\Theta_{ab,c}$ with $\Theta_{ab,c} = 1$ if $y_{ab,c}$ is the smallest element in the set $\mathcal{S}$, and $\Theta_{ab,c} = 0$ otherwise. If two invariants are equal (which is for example the case for $y_{ab,c}$ and $y_{ba,c}$) we consider the invariant $y_{ab,c}$ with the smaller value of

$$\frac{2p_b p_c}{2p_a p_b + 2p_a p_c + 2p_b p_c} \tag{3.17}$$

to be smaller. Since one of the elements must be the smallest, we have

$$1 = \sum_{\mathcal{S}} \Theta_{ab,c} \,. \tag{3.18}$$

The sum is over all elements in the set $\mathcal{S}$. We further define a parameter $s_{\min} = y_{\min} Q^2$ and slice the region where $y_{as,b}$ is the smallest element in the set into two parts as follows: the first region is defined as the region where $s_{as} > s_{\min}$ and $s_{sb} > s_{\min}$. The second region is the complement of the first: $s_{as} < s_{\min}$ or $s_{sb} < s_{\min}$. We now rewrite eq. (3.12) as follows:

$$\int d\sigma^R - d\sigma^A = \int d\phi_{n+1} \sum_{\mathcal{S}} \Theta_{as,b} \left(A_1 + A_2 + A_3 \right) \tag{3.19}$$

with

$$\begin{aligned}
A_1 &= \left|\mathcal{M}(p_1,\ldots,p_{n+1})\right|^2 \Theta(\text{all } y_{ij,k} > y_{\rm res}) \Theta^{\rm cut}_{n+1}(p_1,\ldots,p_{n+1}) \,, \\
A_2 &= \Theta(s_{as} - s_{\min}) \Theta(s_{sb} - s_{\min}) \cdot \\
&\quad \cdot \Bigg[\left|\mathcal{M}(p_1,\ldots,p_{n+1})\right|^2 \left(1 - \Theta(\text{all } y_{ij,k} > y_{\rm res})\right) \Theta^{\rm cut}_{n+1}(p_1,\ldots,p_{n+1}) - \\
&\qquad - \sum_{\text{pairs } i,j} \sum_{k \neq i,j} \mathcal{D}_{ij,k}(p_1,\ldots,p_{n+1}) \Theta(y_{ij,k} < y_{\rm res}) \times \\
&\qquad\qquad \times \Theta^{\rm cut}_{n}(p_1,\ldots,\tilde{p}_{ij},\ldots,\tilde{p}_k,\ldots,p_{n+1}) \Bigg] \,, \\
A_3 &= \left[1 - \Theta(s_{as} - s_{\min}) \Theta(s_{sb} - s_{\min})\right] \cdot \\
&\quad \cdot \Bigg[\left|\mathcal{M}(p_1,\ldots,p_{n+1})\right|^2 \left(1 - \Theta(\text{all } y_{ij,k} > y_{\rm res})\right) \Theta^{\rm cut}_{n+1}(p_1,\ldots,p_{n+1}) - \\
&\qquad - \sum_{\text{pairs } i,j} \sum_{k \neq i,j} \mathcal{D}_{ij,k}(p_1,\ldots,p_{n+1}) \Theta(y_{ij,k} < y_{\rm res}) \times \\
&\qquad\qquad \times \Theta^{\rm cut}_{n}(p_1,\ldots,\tilde{p}_{ij},\ldots,\tilde{p}_k,\ldots,p_{n+1}) \Bigg] \,.
\end{aligned} \tag{3.20}$$

Choosing $y_{\min}$ small enough makes the contribution from the A_3-term negligible. We will therefore not consider this term further. From eqs. (3.19) and (3.20) the structure of our algorithm emerges. If all $y_{ij,k}$ are larger than $y_{\rm res}$, we are in the region corresponding to the A_1-term and we generate configurations according to the unsubtracted matrix element $|\mathcal{M}^{(n+1)}_{\rm Born}|^2$. If one $y_{ij,k}$ is smaller than $y_{\rm res}$, we evaluate the integral of the unresolved region corresponding to the A_2-term, combine it with the virtual corrections and generate a n-parton configuration according to this result.

4. Review of Monte Carlo tools

In this section we review some algorithms concerning the generation of phase space as well as the Metropolis algorithm for sampling a given distribution.

JHEP08(2001)028

4.1 The Metropolis algorithm

In practical applications one often wants to generate random variables according to some probability density $P(x_1, \ldots, x_d)$, which not necesarrily factorizes. In practice one often uses the Metropolis algorithm ([17, 19]) for this purpose. Let us call the vector $\phi = (x_1, \ldots, x_d)$ a state of the ensemble, which we want to generate. Within the Metropolis algorithm one starts from a state ϕ_0, and replaces iteratively an old state by a new one, in such a way, that the correct probability density distribution is obtained in the limit of a large number of such iterations. The equilibrium distribution is reached, regardless of the state one started with. Once the equilibrium distribution is reached, repeated application of the algorithm keeps one in the same ensemble. In short, the desired distribution is the unique fix point of the algorithm. Two important conditions have to be met for the Metropolis algorithm to work: ergodicity and detailed balance. Detailed balance states that the transition probabilities $W(\phi_1 \to \phi_2)$ and $W(\phi_2 \to \phi_1)$ obey

$$P(\phi_1)W(\phi_1 \to \phi_2) = P(\phi_2)W(\phi_2 \to \phi_1) \,. \tag{4.1}$$

Ergodicity requires that each state can be reached from any other state within a finite number of steps. Given a state ϕ_1, one iteration of the Metropolis algoritm consists of the following steps:

1. Generate (randomly) a new candidate ϕ'.
2. Calculate $\Delta S = -\ln(P(\phi')/P(\phi_1))$.
3. If $\Delta S < 0$ set the new state $\phi_2 = \phi'$.
4. If $\Delta S > 0$ accept the new candidate only with probability $P(\phi')/P(\phi)$, otherwise retain the old state $\phi_2 = \phi_1$.
5. Do the next iteration.

Step 3 and 4 can be summarized that the probability of accepting the candidate ϕ' is given by $W(\phi_1 \to \phi') = \min(1, e^{-\Delta S})$. It can be verified that this transition probabilty satisfies detailed balance. The way how a new candidate ϕ' suggested is arbitrary, restricted only by the condition that one has to be able to reach each state within a finite number of steps as well as the condition that the probability $A(\phi_1 \to \phi')$ (the probability of suggesting ϕ' given we are in state ϕ_1) has to be equal to the probability $A(\phi' \to \phi_1)$.

4.2 Rambo

The RAMBO-algorithm [15] maps a hypercube $[0,1]^{4n}$ of random numbers into n physical four-momenta with center-of-mass energy $\sqrt{P^2}$. Massless four-vectors can be generated with uniform weight.

Let $P = (P, 0, 0, 0)$ be a time-like four-vector. The phase space volume for a system of n massless particles with center-of-mass energy $\sqrt{P^2}$ is

$$\Phi_n = \int \prod_{i=1}^{n} \frac{d^4 p_i}{(2\pi)^3} \theta(p_i^0) \delta(p_i^2) (2\pi)^4 \delta^4 \left(P - \sum_{i=1}^{n} p_i \right). \tag{4.2}$$

To derive the RAMBO-algorithm one starts instead from the quantity

$$R_n = \int \prod_{i=1}^{n} \frac{d^4 q_i}{(2\pi)^3} \theta(q_i^0) \delta(q_i^2) (2\pi)^4 f(q_i^0) = (2\pi)^{4-2n} \left(\int_0^\infty x f(x) dx \right)^n. \tag{4.3}$$

The quantity R_n can be interpreted as describing a system of n massless four-momenta q_i^μ that are not constrained by momentum conservation but occur with some weight function f which keeps the total volume finite. The four-vectors q_i^μ are then related to the physical four-momenta p_i^μ by the following Lorentz and scaling transformations:

$$p_i^0 = x \left(\gamma q_i^0 + \vec{b} \cdot \vec{q}_i \right), \qquad \vec{p}_i = x \left(\vec{q}_i + \vec{b} q_i^0 + a \left(\vec{b} \cdot \vec{q}_i \right) \vec{b} \right), \tag{4.4}$$

where

$$\begin{aligned} Q^\mu &= \sum_{i=1}^{n} q_i^\mu, \qquad M = \sqrt{Q^2}, \qquad \vec{b} = -\frac{1}{M} \vec{Q}, \\ \gamma &= \frac{Q^0}{M} = \sqrt{1 + \vec{b}^2}, \qquad a = \frac{1}{1+\gamma}, \qquad x = \frac{\sqrt{P^2}}{M}. \end{aligned} \tag{4.5}$$

Denote this transformation and its inverse as follows

$$p_i^\mu = x H_{\vec{b}}^\mu (q_i), \qquad q_i^\mu = \frac{1}{x} H_{-\vec{b}}^\mu (p_i). \tag{4.6}$$

By a change of variables one can show

$$\begin{aligned} R_n = &\int \prod_{i=1}^{n} \left(\frac{d^4 p_i}{(2\pi)^3} \delta(p_i^2) \theta(p_i^0) \right) (2\pi)^4 \delta^4 \left(P - \sum_{i=1}^{n} p_i \right) \cdot \\ &\cdot \left(\prod_{i=1}^{n} f \left(\frac{1}{x} H_{-\vec{b}}^0 (p_i) \right) \right) \frac{(P^2)^2}{x^{2n+1} \gamma} d^3 b dx. \end{aligned} \tag{4.7}$$

With the choice $f(x) = e^{-x}$ the integrals over $\vec{b}$ and x may be performed and one obtains

$$R_n = \Phi_n \cdot S_n \tag{4.8}$$

with

$$S_n = 2\pi (P^2)^{2-n} \frac{\Gamma(3/2)\Gamma(n-1)\Gamma(2n)}{\Gamma\left(n + \frac{1}{2}\right)}. \tag{4.9}$$

This gives a Monte Carlo algorithm which generates massless four-momenta p_i^μ according to the phase-space measure (4.2). The algorithm consists of two steps:

1. Generate independently n massless four-momenta q_i^μ with isotropic angular distribution and energies q_i^0 distributed according to the density $q_i^0 e^{-q_i} dq_i^0$. Using $4n$ random numbers u_i uniformly distributed in $[0,1]$ this is done as follows:

$$\begin{aligned} c_i &= 2u_{i_1} - 1\,, \qquad \varphi_i = 2\pi u_{i_2}\,, \qquad q_i^0 = -\ln(u_{i_3} u_{i_4})\,, \\ q_i^x &= q_i^0 \sqrt{1-c_i^2}\cos\varphi_i\,, \qquad q_i^y = q_i^0\sqrt{1-c_i^2}\sin\varphi_i\,, \qquad q_i^z = q_i^0 c_i\,. \end{aligned} \tag{4.10}$$

2. The four-vectors q_i^μ are then transformed into the four-vectors p_i^μ, using the transformation (4.4).

Each event has the uniform weight

$$w_0 = (2\pi)^{4-3n}\left(\frac{\pi}{2}\right)^{n-1}\frac{(P^2)^{n-2}}{\Gamma(n)\Gamma(n-1)}\,. \tag{4.11}$$

In summary, the RAMBO algorithm provides a mapping from a $4n$-dimensional hypercube into n physical four-momenta, each event being generated with weight w. By picking points in the hypercube at random, we sweep out the complete phase space. Therefore ergodicity is satisfied and we can combine the RAMBO algorithm with the Metropolis algorithm.

4.3 Generating configurations close to soft or collinear regions

In this section we review an algorithm on how to obtain a $(n+1)$-parton configuration from a given n-parton configuration. An additional four-momentum p_s is inserted between the legs p_a and p_b. For efficiency we generate the events such that $s_{as}, s_{sb} > s_{\min}$ with $s_{\min} = y_{\min} Q^2$. We assume that the complementary region $s_{as} < s_{\min}$ or $s_{sb} < s_{\min}$ gives only a contribution of order $y_{\min}$. We first relate a given $(n+1)$-configuration to a n-parton configuration and invert then this relation to obtain an algorithm to generate soft or collinear insertions [16].

Let k_a', k_s and k_b' be the momenta of the $(n+1)$-parton configuration such that $s_{as} = (k_a' + k_s)^2$, $s_{sb} = (k_b' + k_s)^2$ and $s_{ab} = (k_a' + k_s + k_b')^2$. We want to relate this $(n+1)$ particle configuration to a nearby "hard" n-particle configuration with $(k_a + k_b)^2 = (k_a' + k_s + k_b')^2$, where k_a and k_b are the corresponding "hard" momenta. Using the factorization of the phase space, we have

$$d\Phi_{n+1} = d\Phi_{n-1}\frac{dK^2}{2\pi} d\Phi_3(K, k_a', k_s, k_b')\,. \tag{4.12}$$

The three-particle phase space is given by

$$\begin{aligned} d\Phi_3(K, k_a', k_s, k_b') &= \frac{1}{32(2\pi)^5 s_{ab}} ds_{as} ds_{sb} d\Omega_b' d\phi_s \\ &= \frac{1}{4(2\pi)^3 s_{ab}} ds_{as} ds_{sb} d\phi_s d\Phi_2(K, k_a, k_b) \end{aligned} \tag{4.13}$$

and therefore

$$d\Phi_{n+1} = d\Phi_n \frac{ds_{as}ds_{sb}d\phi_s}{4(2\pi)^3 s_{ab}} \,. \tag{4.14}$$

The region of integration for s_{as} and s_{sb} is $s_{as} > s_{\min}$, $s_{sb} > s_{\min}$ (since we want to restrict the integration to the region where the invariants are larger than $s_{\min}$) and $s_{as} + s_{sb} < s_{ab}$ (Dalitz plot for massless particles). It is desirable to absorb poles in s_{as} and s_{sb} into the measure. A naive numerical integration of these poles without any remapping results in a poor accuracy. This is done by changing the variables according to

$$s_{as} = s_{ab}\left(\frac{s_{\min}}{s_{ab}}\right)^{u_1}, \qquad s_{sb} = s_{ab}\left(\frac{s_{\min}}{s_{ab}}\right)^{u_2}, \tag{4.15}$$

where $0 \le u_1, u_2 \le 1$. Note that $u_1, u_2 > 0$ enforces $s_{as}, s_{sb} > s_{\min}$. Therefore this transformation of variables may only be applied to invariants s_{ij} where the region $0 < s_{ij} < s_{\min}$ is cut out. The phase space measure becomes

$$d\Phi_{n+1} = d\Phi_n \frac{1}{4(2\pi)^3}\frac{s_{as}s_{sb}}{s_{ab}} \ln^2\left(\frac{s_{\min}}{s_{ab}}\right) \Theta(s_{as}+s_{sb}<s_{ab}) du_1 du_2 d\phi_s \,. \tag{4.16}$$

This give the following algorithm for generating a $(n+1)$-parton configuration:

1. Take a "hard" n-parton configuration and pick out two momenta k_a and k_b. Use three uniformly distributed random number v_1, v_2, v_3 and set

$$\begin{aligned} s_{ab} &= (k_a+k_b)^2 \,, \\ s_{as} &= s_{ab}\left(\frac{s_{\min}}{s_{ab}}\right)^{v_1}, \\ s_{sb} &= s_{ab}\left(\frac{s_{\min}}{s_{ab}}\right)^{v_2}, \\ \phi_s &= 2\pi v_3 \,. \end{aligned} \tag{4.17}$$

2. If $(s_{as}+s_{sb}) > s_{ab}$, reject the event.

3. If not, solve for k'_a, k'_b and k_s. If $s_{as} < s_{sb}$ we want to have $k'_b \to k_b$ as $s_{as} \to 0$. Define

$$\begin{aligned} &E_a = \frac{s_{ab}-s_{sb}}{2\sqrt{s_{ab}}}, \qquad E_s = \frac{s_{as}+s_{sb}}{2\sqrt{s_{ab}}}, \qquad E_b = \frac{s_{ab}-s_{as}}{2\sqrt{s_{ab}}}, \\ &\theta_{ab} = \arccos\left(1 - \frac{s_{ab}-s_{as}-s_{sb}}{2E_aE_b}\right), \qquad \theta_{sb} = \arccos\left(1-\frac{s_{sb}}{2E_sE_b}\right). \end{aligned} \tag{4.18}$$

   It is convenient to work in a coordinate system which is obtained by a Lorentz transformation to the center of mass of $k_a + k_b$ and a rotation such that k'_b is along the positive z-axis. In that coordinate system

$$\begin{aligned} p'_a &= E_a(1, \sin\theta_{ab}\cos(\phi_s+\pi), \sin\theta_{ab}\sin(\phi_s+\pi), \cos\theta_{ab}) \,, \\ p_s &= E_s(1, \sin\theta_{sb}\cos\phi_s, \sin\theta_{sb}\sin\phi_s, \cos\theta_{sb}) \,, \\ p'_b &= E_b(1,0,0,1) \,. \end{aligned} \tag{4.19}$$

The momenta p'_a, p_s and p'_b are related to the momenta k'_a, k_s and k'_b by a sequence of Lorentz transformations back to the original frame

$$k'_a = \Lambda_{\mathrm{boost}} \Lambda_{xy}(\phi) \Lambda_{xz}(\theta) p'_a \tag{4.20}$$

and analogous for the other two momenta. The explicit formulae for the Lorentz transformations are obtained as follows:

Denote by $K = \sqrt{(k_a + k_b)^2}$ and by p_b the coordinates of the hard momentum k_b in the center of mass system of $k_a + k_b$. p_b is given by

$$p_b = \left(\frac{Q^0}{K} k_b^0 - \frac{\vec{k}_b \cdot \vec{Q}}{K}, \vec{k}_b + \left(\frac{\vec{k}_b \cdot \vec{Q}}{K(Q^0 + K)} - \frac{k_b^0}{K} \right) \vec{Q} \right) \tag{4.21}$$

with $Q = k_a + k_b$. The angles are then given by

$$\begin{aligned} \theta &= \arccos \left(1 - \frac{p_b \cdot p'_b}{2 p_b^t p_b^{t'}} \right), \\ \phi &= \arctan \left(\frac{p_b^y}{p_b^x} \right). \end{aligned} \tag{4.22}$$

The explicit form of the rotations is

$$\begin{aligned} \Lambda_{xz}(\theta) &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta & 0 & \sin\theta \\ 0 & 0 & 1 & 0 \\ 0 & -\sin\theta & 0 & \cos\theta \end{pmatrix}, \\ \Lambda_{xy}(\phi) &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi & 0 \\ 0 & \sin\phi & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \end{aligned} \tag{4.23}$$

The boost $k' = \Lambda_{\mathrm{boost}} q$ is given by

$$k' = \left(\frac{Q^0}{K} q^0 + \frac{\vec{q} \cdot \vec{Q}}{K}, \vec{q} + \left(\frac{\vec{q} \cdot \vec{Q}}{K(Q^0 + K)} + \frac{q^0}{K} \right) \vec{Q} \right) \tag{4.24}$$

with $Q = k_a + k_b$ and $K = \sqrt{(k_a + k_b)^2}$.

4. If $s_{as} > s_{sb}$, exchange a and b in the formulae above.

5. The "soft" event has then the weight

$$w_{n+1} = \frac{\pi}{2} \frac{1}{(2\pi)^3} \frac{s_{as} s_{sb}}{s_{ab}} \ln^2 \left(\frac{s_{\min}}{s_{ab}} \right) w_n, \tag{4.25}$$

where w_n is the weight of the original "hard" event.

Again by picking v_1, v_2 and v_3 randomly, we sample the complete phase space except the region $s_{as} < s_{\min}$ or $s_{sb} < s_{\min}$.

5. Generating unweighted events according to leading-order matrix elements

In this section we give an algorithm how to generate a sequence of unweighted events according to leading-order matrix elements. This section serves as a warm-up exercise to the next section, where we extend the algorithm to next-to-leading order matrix elements. Since for leading-order calcultions the amplitude squared is always positive, negative weights do not occur at this stage. However we would like to keep the discussion quite general and process-independent. We therefore do not make any assumptions on the form of the matrix element squared. In particular we do not assume that the amplitude squared, viewed as a multi-dimensional probability density factorizes nor do we assume that we have at our disposal a simple function which is everywhere larger than the amplitude squared. Therefore the acceptance-rejection method is not at our disposal and we use the Metropolis algorithm to generate the desired distribution.

Observables in high-energy collider experiments are often of the following form

$$\langle O \rangle = \int d\Phi_n(p_a + p_b, p_1, \dots, p_n) \, \frac{|\mathcal{M}|^2}{8K(s)} \, J(O, p_1, \dots, p_n) \,, \tag{5.1}$$

where p_a and p_b are the momenta of the incoming particles, the outgoing particles are denoted by the labels 1,...,n. The Lorentz-invariant phase space is denoted by $d\Phi_n$, $1/(8K(s))$ is a kinematical factor with $s = (p_a + p_b)^2$ which includes the averaging over initial spins (we assumed two spin states for each initial particle), $|\mathcal{M}|^2$ is the relevant matrix element squared and $J(O, p_1, \dots, p_n)$ is a function which defines the observable and includes all experimental cuts. The situation described above corresponds to electron-positron annihilation. If hadrons appear in the initial state there are slight modifications. Using the RAMBO algorithm eq. (5.1) may be rewritten as

$$\langle O \rangle = \int d^{4n}u \; w_0(u) \frac{|\mathcal{M}(p_i(u))|^2}{8K(s)} \; J(O, p_1, \dots, p_n) \,, \tag{5.2}$$

where the weight $w_0(u)$ is given by eq. (4.11) and the final-state four-momenta p_i are obtained from the u_j's through the RAMBO algorithm. Suppose now that we have at our disposal a jet-defining function $\Theta^n_{y_{\rm res}}(p_1, \dots, p_n)$ such that the support for $J(O, p_1, \dots, p_n)$ is contained in the support for $\Theta^n_{y_{\rm res}}(p_1, \dots, p_n)$. The function $\Theta^n_{y_{\rm res}}(p_1, \dots, p_n)$ returns 1 if the n partons are resolved at the scale $y_{\rm res}$ and 0 otherwise. We may then replace $J(O, p_1, \dots, p_n)$ by

$$\Theta^n_{y_{\rm res}}(p_1, \dots, p_n) J(O, p_1, \dots, p_n) \tag{5.3}$$

and generate random numbers u_j distributed according to the probability density function

$$\frac{1}{\sigma_0} \frac{1}{8K(s)} w_0(u) \, |\mathcal{M}(p_i(u)|^2 \, \Theta^n_{y_{\rm res}}(p_1, \dots, p_n) \,, \tag{5.4}$$

where σ_0 is the total LO-cross section, which normalizes the probability density function to unity:

$$\sigma_0 = \frac{1}{8K(s)} \int d^{4n}u \; w_0(u) \left|\mathcal{M}(p_i(u)\right|^2 \Theta^n_{y_{\rm res}}(p_1, \dots, p_n) \,. \tag{5.5}$$

Note that the quantity in eq. (5.4) is positive-definite, as required for a probability density function. We can apply the Metropolis algorithm to generate the desired distribution.[2]

1. To suggest a new candidate we randomly pick a point $(u'_1, \dots, u'_{4n})$ in the $4n$-dimensional hypercube and use the RAMBO algorithm to obtain the corresponding n-parton configuration of four-vectors $p'_1, \dots, p'_n$.

2. The probability density $P(u')$ for this candidate is given by eq. (5.4).

3. If $P(u') = 0$ (which can occur due the presence of the jet-defining function $\Theta^n_{y_{\rm res}}(p'_1, \dots, p'_n)$) we reject the candidate and go back to step 1.

4. Calculate $\Delta S = -\ln(P(u')/P(u))$, where $P(u)$ is the probability density of the previous state.

5. If $\Delta S < 0$ accept the new candidate and return the n-parton configuration $p'_1, \dots, p'_n$.

6. If $\Delta S > 0$ accept the new candidate only with probability $P(u')/P(u)$, otherwise retain the old state and return the old n-parton configuration $p_1, \dots, p_n$.

7. Do the next iteration.

A technical comment on step 3 is in order: step 3 immediately rejects candidates which do not pass the cuts $\Theta^n_{y_{\rm res}}$. This does not violate detailed balance due to the specific way new candidates are suggested in step 1: a new state $(u'_1, \dots, u'_{4n})$ is suggested independently of the present state $(u_1, \dots, u_{4n})$. If one uses a different procedure in step 1, for example by changing the k-th coordinate $u_k \to u_k + \delta$ by a small, but random value δ, candidates suggested from states close to the boundary of $\Theta^n_{y_{\rm res}}$ are more likely to fall outside the boundary then candidates suggested from states inside the region. To ensure detailed balance, one has to discard step 3 and treat candidates with zero probability in step 6 as candidates which are never accepted. This inserts the old state once again into the event record.

[2]The Metropolis algorithm has also been considered by Kharraziha and S. Moretti [20] for the generation of phase space. These authors use a different method to suggest a new candidate and do not combine the Metropolis algorithm with a "standard" algorithm to generate the phase space.

6. Generating unweighted events according to next-to-leading-order matrix elements

In this section we generalize the algorithm from the previous section to next-to-leading order. For the real emission part we start from eqs. (3.19) and (3.20) and first describe a method, how to pick two legs a and b corresponding to the emitter and the spectator. We then try to insert a soft particle in between these two. We divide the interval $[0,1]$ into m subintervals (m is the number of elements in the set $\mathcal{S}$) and denote the size of the l-th subinterval by P_l. We further introduce functions $\Theta_l(x)$ which return 1 if x is in the l-th subinterval and zero otherwise. We may therefore rewrite eq. (3.19) as

$$\int d\sigma^R - d\sigma^A = \int dx \sum_l \frac{1}{P_l}\Theta_l(x) \int d\phi_{n+1}\Theta_{as,b}\left(A_1 + A_2\right). \tag{6.1}$$

Now the value of x decides where the soft leg gets inserted. In more detail, we may rearrange the above equation as

$$\int d\sigma^R - d\sigma^A = \int d^{4n}u \; w_0(u) \int dx \sum_l \frac{1}{P_l}\Theta_l(x) \int dv_1 \; dv_2 \; dv_3 \; w_{\mathrm{soft}}\Theta_{as,b} \times \\ \times \left(A_1 + A_2\right), \tag{6.2}$$

with

$$w_{\mathrm{soft}} = \frac{\pi}{2}\frac{1}{(2\pi)^3}\frac{s_{as}s_{sb}}{s_{ab}}\ln^2\left(\frac{s_{\min}}{s_{ab}}\right). \tag{6.3}$$

We may therefore use $4n+1+3$ random numbers $u_1, \dots, u_{4n}$, x, v_1, v_2 and v_3, all uniformly distributed in $[0,1]$ and generate the phase space as follows: we use the random numbers $u_1, \dots, u_{4n}$ to generate a hard configuration of n four-momenta using the RAMBO algorithm. The value of the random number x decides where we try to insert a soft leg. We check into which of the m subintervals x falls and pick out the corresponding element of the set $\mathcal{S}$. This tells us which parton is going to be the emitter and which parton is going to be the spectator. We now use the random numbers v_1, v_2 and v_3 and the algorithm for soft insertions and generate a $(n+1)$-parton configuration, where a soft particle has been inserted between the emitter and the spectator. This suggests the following Metropolis algorithm:

1. To suggest a new candidate we randomly pick a point $(u_1', \dots, u_{4n}', x', v_1', v_2', v_3')$ in the $4n+1+3$-dimensional hypercube. Using the first $4n$ random numbers $u_1, \dots, u_{4n}$ and the RAMBO algorithm we obtain the corresponding "hard" n-parton configuration of four-vectors $\phi_{\mathrm{hard}}' = (q_1', \dots, q_n')$. From this hard configuration and the values of x and v_1, v_2 and v_3 we obtain a related "soft" $(n+1)$-parton configuration $\phi_{\mathrm{soft}}' = (p_1', \dots, p_n', p_{n+1}')$ together with the information where the soft particle s has been inserted. Therefore the emitter a and the spectator b are known.

2. Check if $y_{as,b}$ is the smallest value in the set $\mathcal{S}$. If not, reject the candidate and go back to step 1.

3. Check if all $y_{ij,k}$ are larger then $y_{\rm res}$.

   If this is the case, the probabilty density of the new candidate is given by

$$P' = \frac{1}{\sigma_{\rm NLO}} \frac{1}{8K(s)} \frac{w_0(u) w_{\rm soft}}{P_l} \left| \mathcal{M}^{(n+1)}_{\rm Born}(p'_1, \dots, p'_{n+1}) \right|^2 \times \\ \times \Theta^{n+1}_{y_{\rm res}}(p'_1, \dots, p'_{n+1}) \,. \tag{6.4}$$

   and we set a flag $f = 1$. Here $\sigma_{\rm NLO}$ is given by

$$\sigma_{\rm NLO} = \frac{1}{8K(s)} \cdot \tag{6.5}$$
$$\cdot \left\{ \int d\phi_n \left[\left| \mathcal{M}^{(n)}_{\rm Born} \right|^2 + 2\,{\rm Re}\, \mathcal{M}^{(n)}_{\rm Born}{}^* \mathcal{M}^{(n)}_{\rm 1-loop} + \left\langle \mathcal{M}^{(n)}_{\rm Born} | \mathbf{I} | \mathcal{M}^{(n)}_{\rm Born} \right\rangle \right] \Theta^n_{y_{\rm res}} + \right.$$
$$\left. + \int d\phi_{n+1} \left[\left| \mathcal{M}^{(n+1)}_{\rm Born} \right|^2 \Theta^{n+1}_{y_{\rm res}} - \sum_{\text{pairs } i,j} \sum_{k \neq i,j} \mathcal{D}_{ij,k} \Theta(y_{ij,k} < y_{\rm res}) \Theta^n_{y_{\rm res}} \right] \right\} .$$

   $\Theta^{n+1}_{y_{\rm res}}$ returns 1 if at least n partons are resolved at the scale $y_{\rm res}$.

   If $P' = 0$ (which can occur due the presence of the jet-defining function $\Theta^{n+1}_{y_{\rm res}}(p'_1, \dots, p'_{n+1})$) we reject the candidate and go back to step 1.

   If at least one $y_{ij,k}$ in the set $\mathcal{S}$ is smaller then $y_{\rm res}$ the probability density is given by

$$P' = \frac{1}{\sigma_{\rm NLO}} \frac{1}{8K(s)} \frac{w_0}{V} \left\{ \left[\left| \mathcal{M}^{(n)}_{\rm Born} \right|^2 + 2\,{\rm Re}\, \mathcal{M}^{(n)}_{\rm Born}{}^* \mathcal{M}^{(n)}_{\rm 1-loop} + \right.\right.$$
$$\left.\left. + \left\langle \mathcal{M}^{(n)}_{\rm Born} | \mathbf{I} | \mathcal{M}^{(n)}_{\rm Born} \right\rangle \right] \Theta^n_{y_{\rm res}}(q'_1, \dots, q'_n) + R \right\}, \tag{6.6}$$

   where the integral over the unresolved region is given by

$$R = \sum_{\mathcal{S}} \int dv_1\, dv_2\, dv_3\, w_{\rm soft} \Theta_{as,b} \left[\left| \mathcal{M}^{(n+1)}_{\rm Born} \right|^2 (1 - \Theta(\text{all } y_{ij,k} > y_{\rm res}))\, \Theta^{n+1}_{y_{\rm res}} - \right.$$
$$\left. - \sum_{\text{pairs } i,j} \sum_{k \neq i,j} \mathcal{D}_{ij,k} \Theta(y_{ij,k} < y_{\rm res}) \Theta^n_{y_{\rm res}} \right] \tag{6.7}$$

   and the unresolved volume is given by

$$V = \sum_{\mathcal{S}} \int dv_1\, dv_2\, dv_3\, \Theta_{as,b}\, (1 - \Theta(\text{all } y_{ij,k} > y_{\rm res})) \,. \tag{6.8}$$

   In addition set the flag $f = 0$.

   If $P' < 0$ (which can occur if $y_{\rm res}$ is chosen too small) throw an exception.

4. Calculate $\Delta S = -\ln(P'/P)$, where P is the probability density of the previous state.

5. If $\Delta S < 0$ accept the new candidate. If the flag $f = 1$ return the "soft" $(n+1)$-parton configuration $p'_1, \ldots, p'_n$ otherwise if $f = 0$ return the "hard" n-parton configuration $q'_1, \ldots, q'_n$.

6. If $\Delta S > 0$ accept the new candidate only with probability P'/P, otherwise retain the old state and return the old parton configuration.

7. Do the next iteration.

If the value of y_{res} is chosen too small, it can occur that P' in eq. (6.6) turns out negative. However, this is not an artefact of our algorithm, but a failure of perturbation theory: P' corresponds to the NLO cross section for observing n jets with (fixed) four-vectors $q'_1, \ldots, q'_n$ resolved at a scale y_{res}. If y_{res} is too small the NLO prediction might become negative. This is of course unphysical. The algorihtm throws an exception in this case and one should rerun the program with a larger value of y_{res}.

Given a generated sequence of momentum configurations $\phi_1, \ldots, \phi_N$, where each momentum configuration contains either n or $(n+1)$ four-vectors, the Monte Carlo estimate for an infrared safe observable is then given by

$$\langle O \rangle = \sigma_{\mathrm{NLO}} \frac{1}{N} \sum_{i=1}^{N} J(O, \phi_i) \, . \tag{6.9}$$

Here $J(O, \phi_i)$ is a function which defines the observable and includes all experimental cuts. The approximations involved are:

- If within an $(n+1)$-parton configuration some value of $y_{ij,k}$ is smaller then y_{res}, this configuration is replaced with an n-parton configuration, which corresponds to the average over the unresolved region combined with the virtual corrections and the Born term.

- Due to the presence of the jet-defining functions $\Theta^{n}_{y_{\mathrm{res}}}$ and $\Theta^{n+1}_{y_{\mathrm{res}}}$ configurations which would classify as $(n-1)$-parton events according to these functions are cut out. One has to ensure that the support for the actual observable O is contained in the support for $\Theta^{n}_{y_{\mathrm{res}}}$ and $\Theta^{n+1}_{y_{\mathrm{res}}}$.

- We neglected the term A_3 in eq. (3.19). This introduces a (negligible) systematic error of $y_{\min}$. This is done for efficiency.

JHEP08(2001)028

7. Conclusions

In this paper we gave an algorithm to generate unweighted events of n or $(n+1)$ four-vectors according to next-to-leading order matrix elements. We considered massless QCD in electron-positron annihilation. The extension of our methods to initial-state partons or massive partons should be possible.

References

[1] G. Corcella et al., *Herwig 6: an event generator for hadron emission reactions with interfering gluons (including supersymmetric processes)*, *J. High Energy Phys.* **01** (2001) 010 [hep-ph/0011363].

[2] T. Sjöstrand, *High-energy physics event generation with PYTHIA 5.7 and JETSET 7.4. Comput. Phys. Commun.* **82** (1994) 74.

[3] C. Friberg and T. Sjöstrand, *Some thoughts on how to match leading log parton showers with nlo matrix elements*, hep-ph/9906316.

[4] J.C. Collins, *Subtraction method for nlo corrections in Monte Carlo event generators for leptoproduction*, *J. High Energy Phys.* **05** (2000) 004 [hep-ph/0001040].

[5] B. Pötter, *Combining QCD matrix elements at next-to-leading order with parton showers*, *Phys. Rev.* **D 63** (2001) 114017 [hep-ph/0007172].

[6] M. Dobbs and M. Lefebvre, *Unweighted event generation in hadronic WZ production at order (α_s)*, *Phys. Rev.* **D 63** (2001) 053011 [hep-ph/0011206].

[7] J.C. Collins and F. Hautmann, *Soft gluons and gauge-invariant subtractions in NLO parton-shower Monte Carlo event generators*, *J. High Energy Phys.* **03** (2001) 016 [hep-ph/0009286].

[8] B. Pötter and T. Schorner, *Combining parton showers with next-to-leading order QCD matrix elements in deep-inelastic eP scattering*, *Phys. Lett.* **B 517** (2001) 86 [hep-ph/0104261].

[9] M. Dobbs, *Incorporating next-to-leading order matrix elements for hadronic diboson production in showering event generators*, *Phys. Rev.* **D 64** (2001) 034016 [hep-ph/0103174].

[10] Y. Chen, J.C. Collins and N. Tkachuk, *Subtraction method for NLO corrections in Monte-Carlo event generators for z boson production*, *J. High Energy Phys.* **06** (2001) 015 [hep-ph/0105291].

[11] S. Mrenna, *Higher order corrections to parton showering from resummation calculations*, hep-ph/9902471.

[12] H. Baer and M. H. Reno, *A complete $O(\alpha_s)$ event generator for $p\bar{p} \to w^+x \to e^+nx$ with parton showering*, *Phys. Rev.* **D 44** (1991) 3375; *W and Z production at $p\bar{p}$ colliders: parton showers merged with $O(\alpha_s)$ Monte Carlo approach*, *Phys. Rev.* **D 45** (1992) 1503.

[13] S. Catani and M.H. Seymour, *A general algorithm for calculating jet cross sections in NLO QCD*, *Nucl. Phys.* **B 485** (1997) 291 [hep-ph/9605323].

[14] Z. Nagy and Z. Trócsányi, *Next-to-leading order calculation of four-jet observables in electron positron annihilation*, *Phys. Rev.* **D 59** (1999) 014020 [hep-ph/9806317].

[15] R. Kleiss, W. J. Stirling, and S. D. Ellis, *A new Monte Carlo treatment of multiparticle phase space at high-energies*, *Comput. Phys. Commun.* **40** (1986) 359.

[16] S. Weinzierl and D.A. Kosower, *QCD corrections to four-jet production and three-jet structure in e^+e^- annihilation*, *Phys. Rev.* **D 60** (1999) 054028 [hep-ph/9901277].

[17] N. Metropolis et al., *Equation of state calculations by fast computing machines*, *J. Chem. Phys* **21** (1953) 1087.

[18] S. Catani, M.H. Seymour and Z. Trocsanyi, *Regularization scheme independence and unitarity in QCD cross sections*, *Phys. Rev.* **D 55** (1997) 6819 [hep-ph/9610553].

[19] G. Bhanot, *The metropolis algorithm*, *Rept. Prog. Phys.* **51** (1988) 429.

[20] H. Kharraziha and S. Moretti, *The metropolis algorithm for on-shell four-momentum phase space*, *Erratum-ibid.* **134** (2001) 136–138 [hep-ph/9909313].